

# Fame to blame

<https://gitlab-ovh-02.cloud.centralesupelec.fr/tom.bray/fame-to-blame>

Tom Bray, Siméon Boyer, Gaspard  
Debiais, Ludovic Mulat, Matthieu Neau  
et Mathilde Jacquotot

# Présentation du produit

**GENERE UN CLASSEMENT DES CELEBRITES LES PLUS INSULTEES SUR TWITTER PARMIS UN PANEL CHOISI PAR L'UTILISATEUR**

Utilise la détection des insultes à des fins de divertissement.

Sensibilisation des utilisateurs de Twitter.

En français ou en anglais.

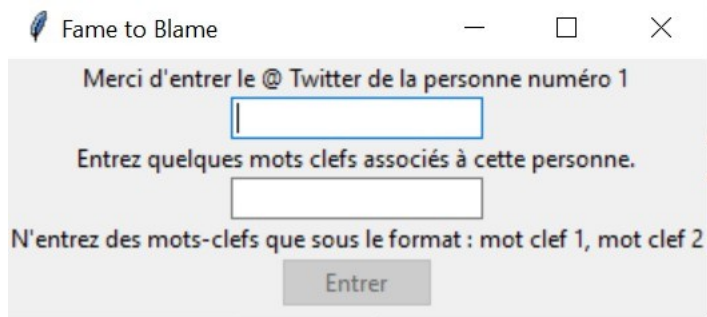
Possibilité d' utiliser le produit pour détecter le harcèlement.

# Présentation du produit

## FONCTIONNEMENT

Choix de la langue puis de plusieurs célébrités qui possèdent un compte Twitter + mots-clés.

Classement par pourcentage de tweets insultants



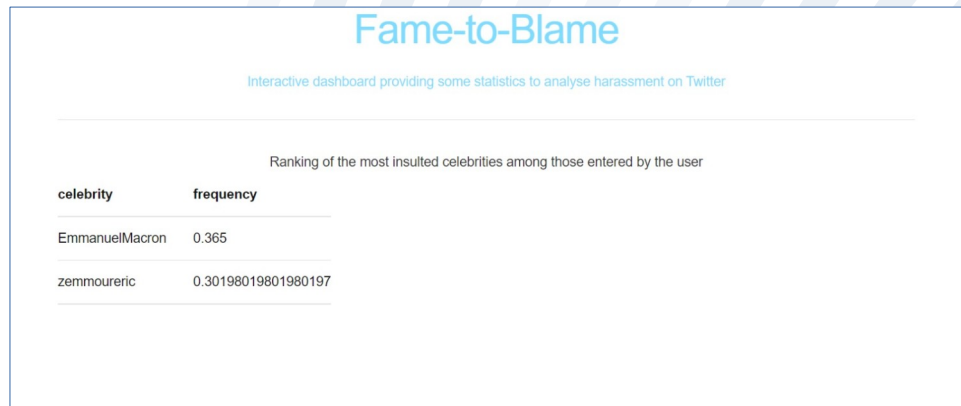
Fame to Blame

Merci d'entrer le @ Twitter de la personne numéro 1

Entrez quelques mots clefs associés à cette personne.

N'entrez des mots-clefs que sous le format : mot clef 1, mot clef 2

Entrer



**Fame-to-Blame**

Interactive dashboard providing some statistics to analyse harassment on Twitter

Ranking of the most insulted celebrities among those entered by the user

celebrity	frequency
EmmanuelMacron	0.365
zemmoureric	0.30198019801980197

# Etapes de conception - Sprints



# Répartition des tâches



**Tom Bray**

Sprint 7

Coordination de l'équipe



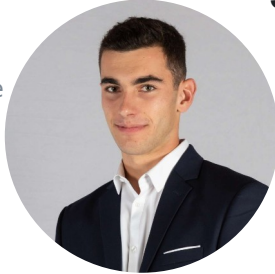
**Siméon Boyer**

Sprints 4 et 6



**Gaspard Debiais**

Sprints 1, 2 et 5



**Ludovic Mulat**

Sprints 5, 6 et 7  
Début du 8  
(Staff Forum)



**Matthieu Neau**

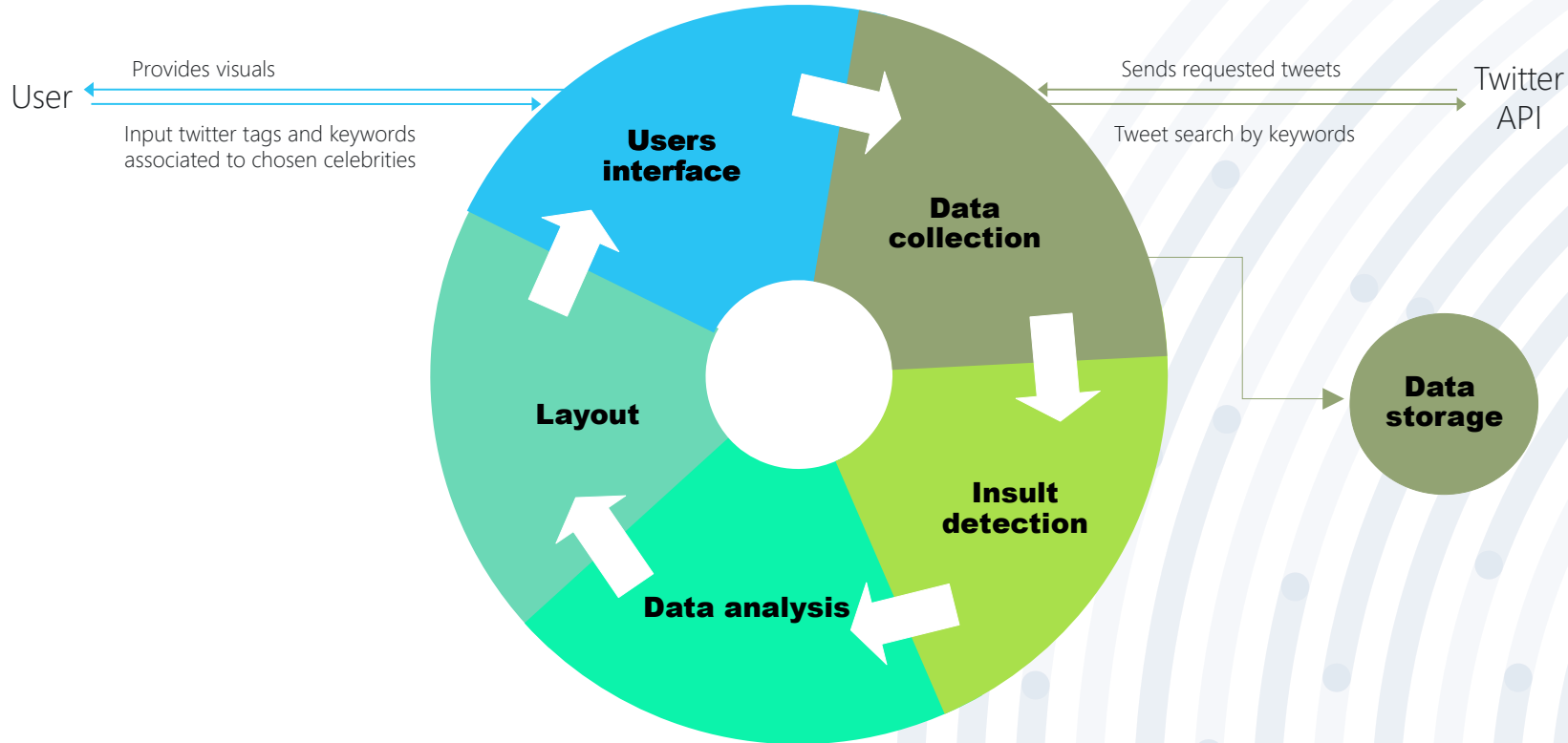
Sprints 1, 2 et 5  
(Staff Forum)



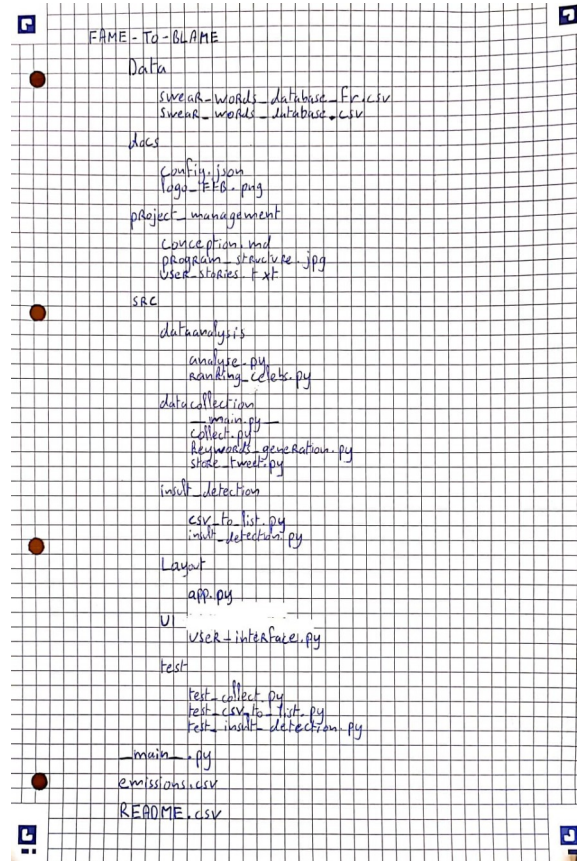
**Mathilde Jacquotot**

Sprint 3  
Slides de présentation

# La structure de notre code



# La structure de notre code - détail





# La structure de notre code – extrait du main

## PARAMETRAGE

Langue

Path vers le répertoire qui contient credentials

Nombre de célébrités à classer

Identifiant twitter et mots clés pour chaque célébrité

```
start()
from src.UI.user_interface import PATH,NUM_OF_CEL,PEOPLES,LANGUAGE
set_of_data=[]
names = []
flag=0
for celeb_number in range(NUM_OF_CEL):
    user_keywords=PEOPLES[celeb_number]
    keywords=converting_keywords(user_keywords)
    names.append(keywords[0])
    tweet_collection=collection(keywords,PATH,lang=LANGUAGE)
    if tweet_collection == 'This twitter user does not exist.':
        set_of_data.append([celeb_number,keywords[0],'This user does not exist.'])
        flag+=1
    else:
        tweet_collection.insert(0,celeb_number)
        set_of_data.append(tweet_collection)
if flag==0:
```

```
    if flag==0:
        detect_insults=detect_insults_tweets(set_of_data,lang=LANGUAGE)
        list_of_frequencies={'celebrity':[], 'frequency' : []}

        for celeb_number in range(NUM_OF_CEL):
            frequency = insult_frequency(detect_insults[str(celeb_number)])
            list_of_frequencies['celebrity'].append(names[celeb_number])
            list_of_frequencies['frequency'].append(frequency)
        cleanup(names)
        print(list_of_frequencies)
        print(detect_insults)
        print(pd.DataFrame.from_dict(list_of_frequencies))
        appli(pd.DataFrame.from_dict(list_of_frequencies))

    else :
        print(str(flag) + ' celebrities you entered did not have a valid twitter username')
        print(set_of_data)

main()
```



# Documentation du code

## Exemple de documentation de fonction

```
def detect_insult_tweet(tweet, lang='en'):
    """
    Returns a list of the insults contained in a tweet

    Parameters
    -----
    tweet : a dataframe panda with a key 'Texte'

    Returns
    -----
    insults : a list of strings
    """
```

## Elements abordés dans le README.md

- Consignes d' installation
- Consignes d' utilisation
- Présentation des développeurs
- Comment le projet pourrait être amélioré
- Consignes pour contribuer au projet
- Versions
- Lien vers le GitLab

## Qualité du code – tests

- Pylint : propreté du code  
Moyenne de 8,5
- Carbon AI : consommation de notre application  
0,15 mWh
- Pytest : couverture du code  
Les modules de test sont créés mais ne sont pas encore utilisables par pytest.



# **Merci de votre attention**

**Nous allons désormais vous faire une démonstration de l'utilisation du produit**