

# ***J-VOTING***

Par Chahid, Firas, Hugo, Lucas,  
Maxime, Nicolas et Sylvain



# ***Sommaire***

- **Projet J-Voting**
  - + Objets manipulés
  - + L'objectif
- **Évolutions apportées**
  - + ExportDot
  - + GUI
- **Difficultés rencontrées**



# ***D'une alternative à un profile***

---

Alternative

---

Préférence

---

Votant

---

Profile





M. B :



Mme A :



Madame A

Monsieur B

Madame C

Shrek

Iphone 11

Raclette







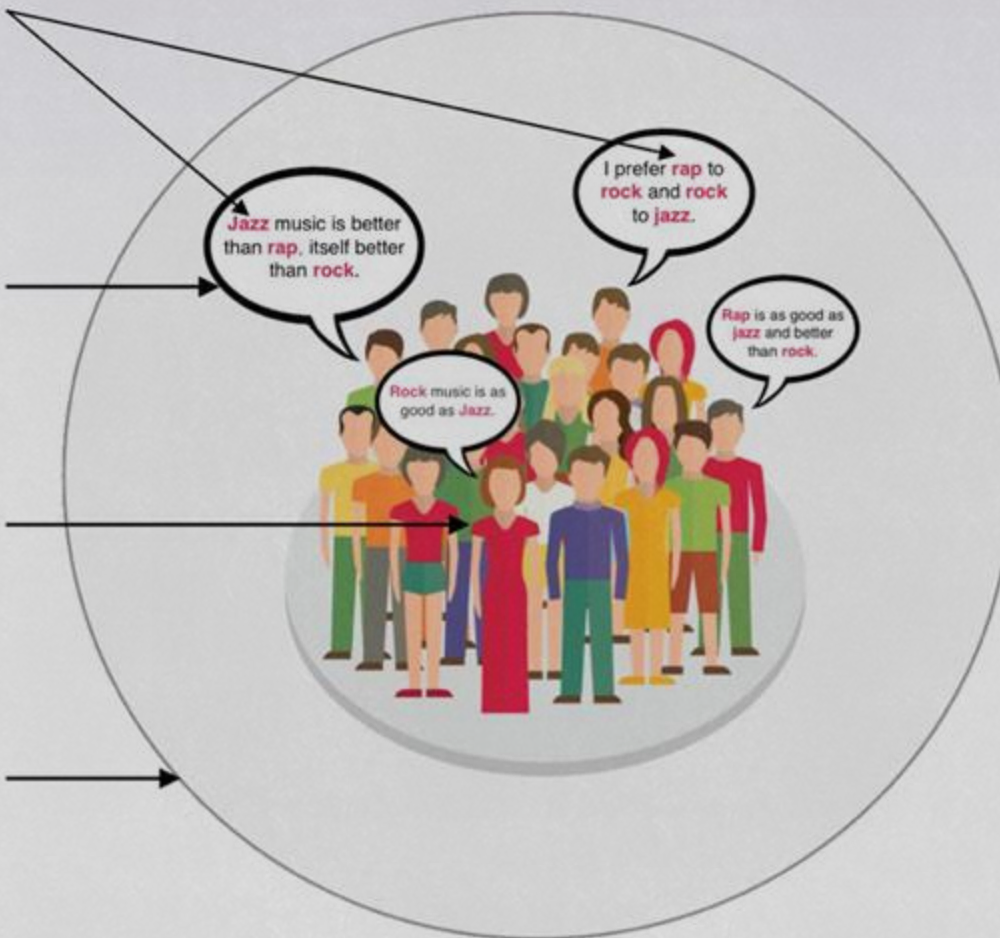
Il existe plusieurs types de préférences et donc de profils !

Those are Alternatives

This is a Preference

This is a Voter

This is a Profile



# ***Dans quel but?***

## Manipuler:

- Editer différentes formes de préférence

## Visualiser:

- Afficher les préférences sous différents formats

## Tester des méthodes de vote:

- La théorie du choix social



***Evolutions***

***apportées***



# ***Export DOT***

## Utilité :

- Exporter une **préférence** dans un format **.dot**

## Objectif :

- Créer la classe **ExportDot**





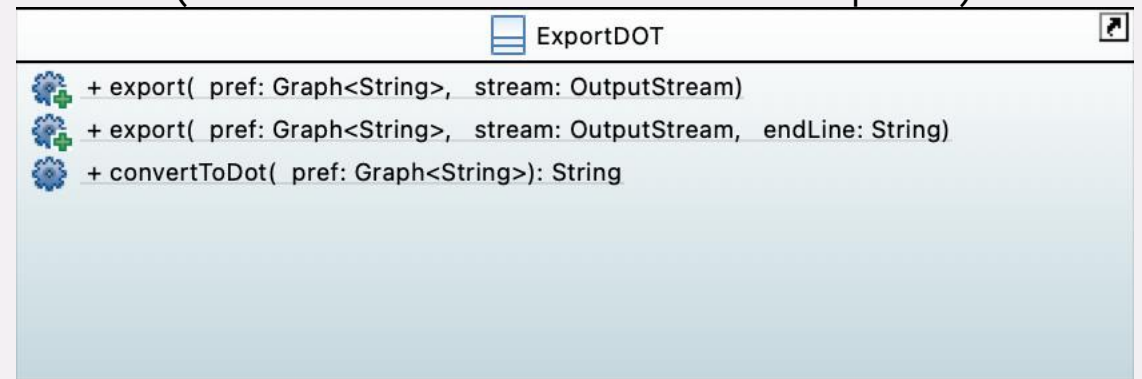
# **1<sup>ère</sup> mission : Documenter la classe ExportDot**

- Définition :

- Bilan :

- + 4 itérations :

- Documentation approuvée dans master
    - Ajout continu de documentation, problème de FOCUS. Donc approuvée dans master lors de la documentation de GraphViz (Cf. 3<sup>ème</sup> mission : Documenter GraphViz)



```
ExportDOT
+ export( pref: Graph<String>, stream: OutputStream)
+ export( pref: Graph<String>, stream: OutputStream, endLine: String)
+ convertToDot( pref: Graph<String>): String
```



# ***Format dot***

- Définition
- Exemple

```
digraph G {  
    a1;  
    a2;  
    a4;  
    a3;  
    a1 -> a2;  
    a1 -> a4;  
    a2 -> a3;  
    a2 -> a4;  
    a3 -> a4;  
}
```



# **2<sup>ème</sup> mission : Implémenter la classe ExportDot en Java**

- Enjeux :

- + Implémenter 2 méthodes

- convertToDot()
- Export()

- Bilan :

- + 4 itérations

- + Approuvé dans le master

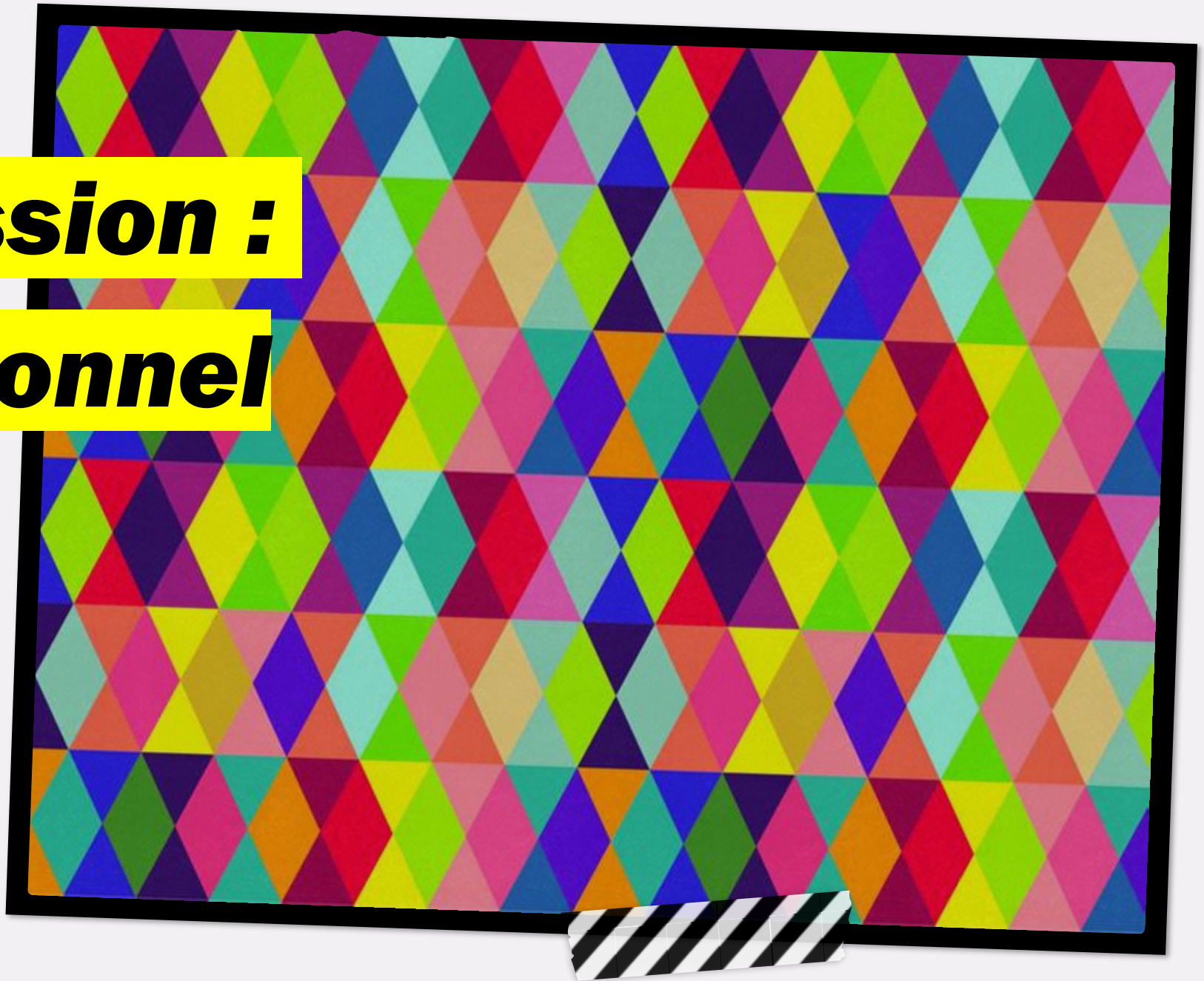


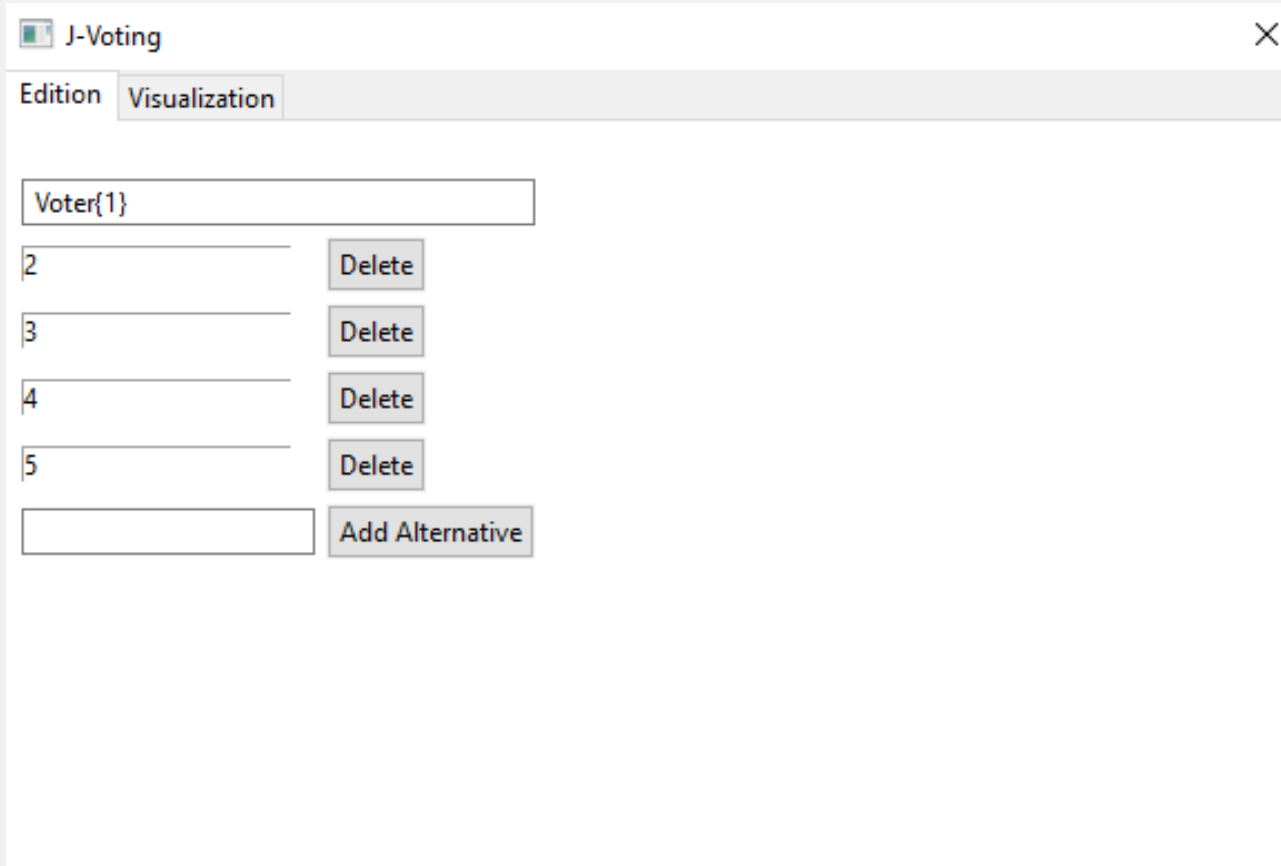
# **3<sup>ème</sup> mission : le plugin GraphViz**

- Définition & utilité pour Export Dot
- Problèmes rencontrés



# ***4 ème mission :*** ***GUI fonctionnel***





# ***Qu'est-ce qu'un GUI ?***

- Le GUI (Graphical User Interface) est l'interface qu'utilise l'utilisateur pour interagir avec le programme





3  
1, Shrek (Full-screen)  
2, The X-Files: Season 2  
3, The Punisher  
7, 7, 6  
1, 2, 1, 3  
1, 1, 2, 3  
1, 1, 3, 2  
1, 2, 3, 1  
1, 3, 1, 2  
2, 3, 2, 1

## **Qu'est-ce qu'un fichier SOC ?**

- Les fichiers au format SOC facilitent le stockage des votants et des alternatives pour lesquelles ils ont votées



J-voting client

Edition Visualization

Open / Save as

Enter a profile name

Voter1 name

1. Alternative 1

2. Alternative 2

3. Alternative 3

4. Alternative 4

5. New alternative

Voter2 name

1. Alternative 1

2. Alternative 2

3. Alternative 3

4. Alternative 4

5. New alternative

# Fonctionnement du GUI

- Implémentation d'un storyboard et de diagramme de séquences
- 3 scénarios d'utilisation définis
- Fonctionnement "basique du GUI"



# ***Difficultées rencontrées dans l'implémentation***

- Visualiser plusieurs votants dans le GUI à partir d'un fichier SOC :
  - Classe MutableLinearPreferenceImpl
  - Tableau de MutableLinearPreferenceImpl
  - Solutions complexes
- Implémentation d'une scrollbar :
  - Problème avec le mainComposite et le scrollComposite





# ***Difficultés rencontrées***

- Plugin GraphViz
- FOCUS
- Compréhension

