

# Wolves Tournament Master

Présenté par :

- Chambaz Alan
- Pelletier Colin
- Roso Sébastien

# Table des matières

- Présentation du projet
- Démonstration
- Présentation des technologies utilisées
- Aspects d'implémentation particuliers
- CI/CD
- Améliorations

# But de l'application

- Organisation de tournois de e-sportifs
  - Création et gestion d'équipes
  - Création de tournois
  - Création d'arbres de tournois automatique
  - Arbitrage de tournois

# Notifications

- Gestion des notifications
  - Invitation d'une équipe
  - Renvoi d'une équipe
  - Assignment d'arbitrage
  - Participations de l'équipe à un tournoi

# Notifications



## Information

[Team] You have been fired from DLM B



## Invitation

You were invited to join a team!

[ACCEPT](#)[DECLINE](#)

## Information

[Accepted] You were invited to join DLM B team!



## Information

[Referee] You've been assigned as referee for the tournament ddd of ddd the 2021-04-23



## Information

[Tournament] Your team DLM Alpha participates to the tournament CS GO League of CS GO the 2021-04-24.

# Tournois

DLM



Doc enjoyers VS DLM Alpha ^

Score : 0 - 16

Jambon fromage VS Macaroni ^

Score : 12 - 16

DLM Alpha VS Macaroni ^

Score : 16 - 9



# Gestion des membres d'équipe

Recruit a new member



## USERS

Username ↑

Email

Actions

Alan

alan@wtm.ch



Deltora

Deltora@gmail.com



George

george@wtm.ch



test

test@test.ch



Rows per page:

10 ▼

1-4 of 4



# Gestion des membres d'équipe



## Invitation

You were invited to join the big brains team!

ACCEPT

DECLINE



# Gestion des membres d'équipe

## Members



test

test@test.ch

EXCLUDE



seb

seb@gmail.com



Alan

alan@wtm.ch

EXCLUDE



Deltora

Deltora@gmail.com

EXCLUDE

# Gestion des membres d'équipe



## Information

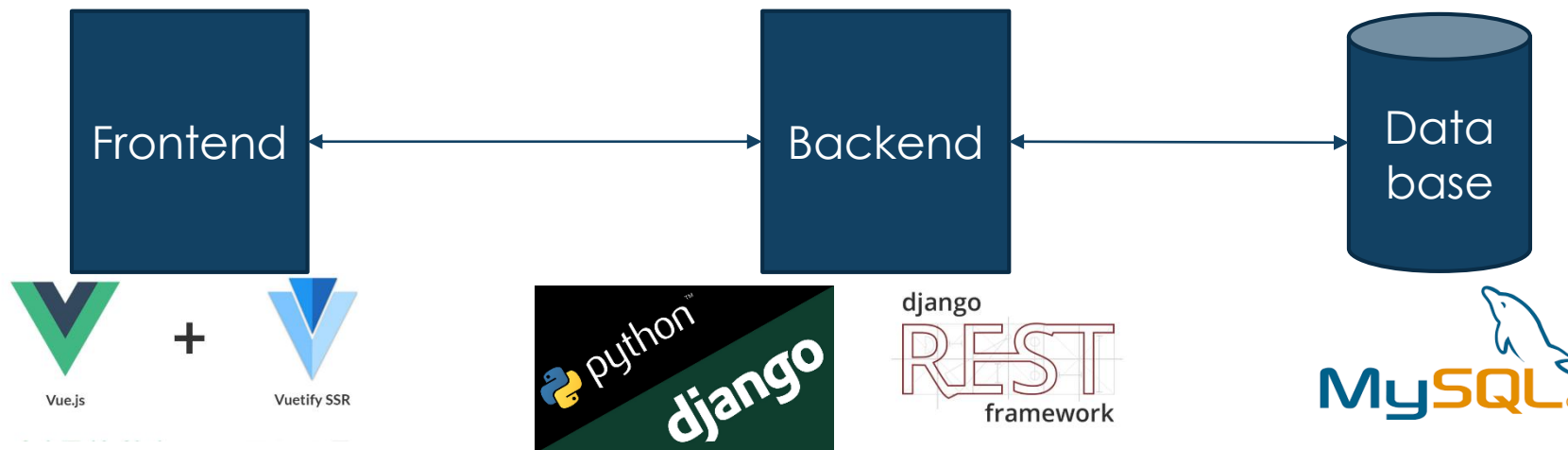
[Team] You have been fired from the big brains

# Démonstration



# Technologies utilisées

- Backend : Django REST Framework
- Frontend : VueJS, Vuetify, ...
- DB : MySQL

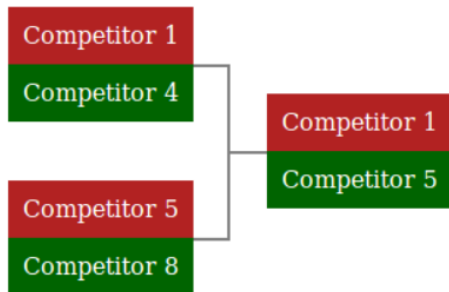


# Arbre de tournois

- Affichage
  - V1 : génération de code HTML récursive -> Musée des horreurs
  - V2 : prise en main et amélioration du composant NodeJS «vue-tournament-bracket »

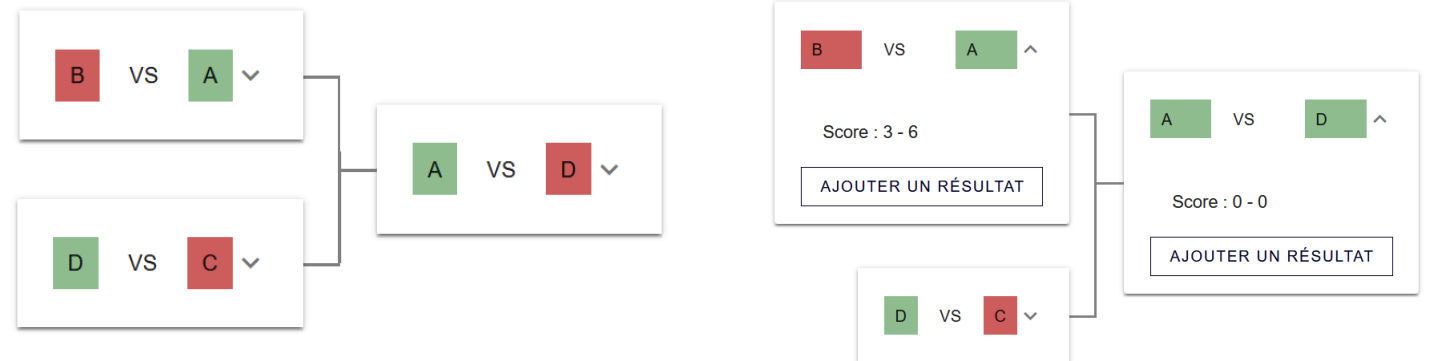
# Arbre de tournois

- Version originale



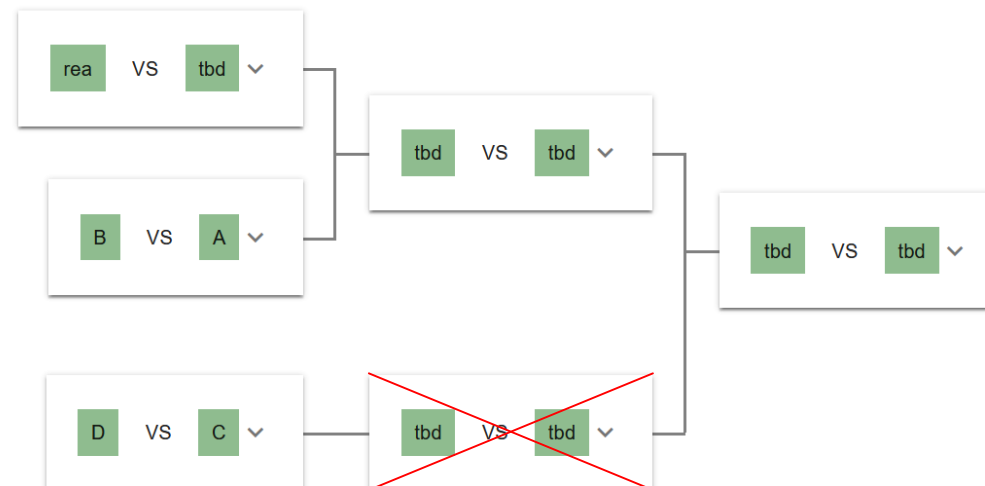
<https://raw.githubusercontent.com/kamilwylegala/vue-tournament-bracket/master/docs/thirdplaceplayoff.png>

- Version améliorée



# Arbre de tournois

- Algorithme de génération
  - Traité comme un arbre binaire
  - Problème : trop de cas spécifiques
  - Exemple avec 5 équipes:



# Requête générique

```
Request(method, url, data = null, header = null) {  
  return new Promise(resolve => {  
    axios({  
      method: method,  
      url: url,  
      data: data,  
      headers: header  
    })  
    .then(response => {  
      resolve({ isSuccess: true, result: response.data })  
    })  
    .catch(error => {  
      resolve({ isSuccess: false, result: error })  
    })  
  })  
},
```

Verbe HTTP

URL de la ressource

Données à envoyer

En-tête de la requête



# CI/CD

- 3 actions
  - Build Backend : push sur dev et main
  - Build Frontend : push sur dev et main
  - Deploy : push sur main

# Améliorations

- Actualisation des notifications
- Rendu de l'arbre de tournois
- Affichage des menus sur grand écran
- Page d'administration

# Sources

Description	Source
Documentation VueJS	<a href="https://vuejs.org/v2/guide/">https://vuejs.org/v2/guide/</a>
Documentation Rest Framework	<a href="https://www.django-rest-framework.org/topics/documenting-your-api/">https://www.django-rest-framework.org/topics/documenting-your-api/</a>
Vuetify	<a href="https://vuetifyjs.com/en/getting-started/installation/">https://vuetifyjs.com/en/getting-started/installation/</a>
Vue-tournament-bracket	<a href="https://www.npmjs.com/package/vue-tournament-bracket">https://www.npmjs.com/package/vue-tournament-bracket</a>