

# Développement d'un outil de scènes visuelles favorisant la communication des personnes en situation de handicap cognitif

par Yanis Girardin  
2023

Jury

M. Portet

Mme. Laget-Kamel

# Laboratoire de recherche



Recherche et Développement dans l'informatique

# Axes de recherche

## AXES DE RECHERCHE ET EQUIPES

### Génie des Logiciels et des Systèmes d'Information

CTRL-A\* : E. Rutten  
SIGMA : C. Labbé  
VASCO : Y. Ledru

### Méthodes Formelles, Modèles et Langages

CAPP : N. Peltier  
CONVECS\* : R. Mateescu  
SPADES\* : G. Goessler  
TYREX\* : P. Geneves

### Systèmes Interactifs et Cognitifs

IIHM : L. Nigay  
MARVIN : D. Pellier  
MeTAH : A. Chaachoua  
M-PSI : D. Vaufreydaz

### Systèmes Répartis, Calcul Parallèle et Réseaux

CORSE\* : F. Rastello  
DATAMOVE\* : B. Raffin  
DRAKKAR : A. Duda  
ERODS : D. Donsez  
POLARIS\* : A. Legrand

### Systèmes Intelligents pour les Données, les Connaissances et les Humains

AMA : M.-R. Amini  
GETALP : F. Portet  
MOEX\* : J. Euzenat  
MRIM : G. Quénot  
SLIDE : S. Amer-Yahia  
STEAMER : M. Villanova-Oliver

Tiré de l'organigramme du LIG, janvier 2021 : <https://www.liglab.fr/fr/presentation/organigramme>

# L'équipe

Systèmes Intelligents pour les Données,  
les Connaissances et les Humains

AMA : M.-R. Amini

GETALP : F. Portet

MOEX\* : J. Euzenat

MRIM : G. Quénot

SLIDE : S. Amer-Yahia

STEAMER : M. Villanova-Oliver

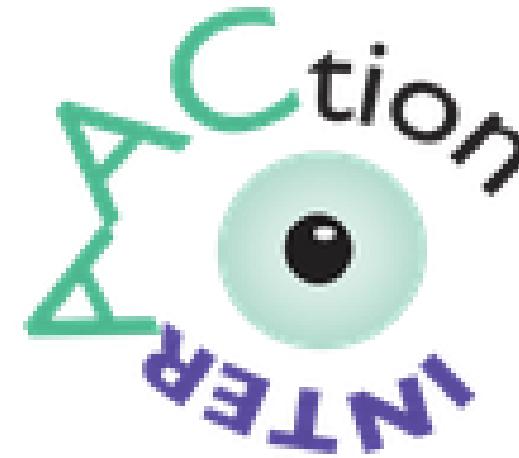


Communication

Traitement de l'information

Groupe d'Étude pour la Traduction Automatique et le  
Traitement Automatisé des Langues et de la Parole

# InterAACTION & la CAA



## Communication Alternative et Augmentée

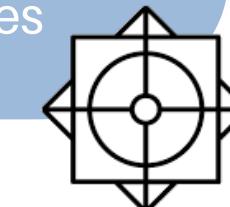
*Techniques de communication visant à remplacer la parole*

### 3 Objectifs

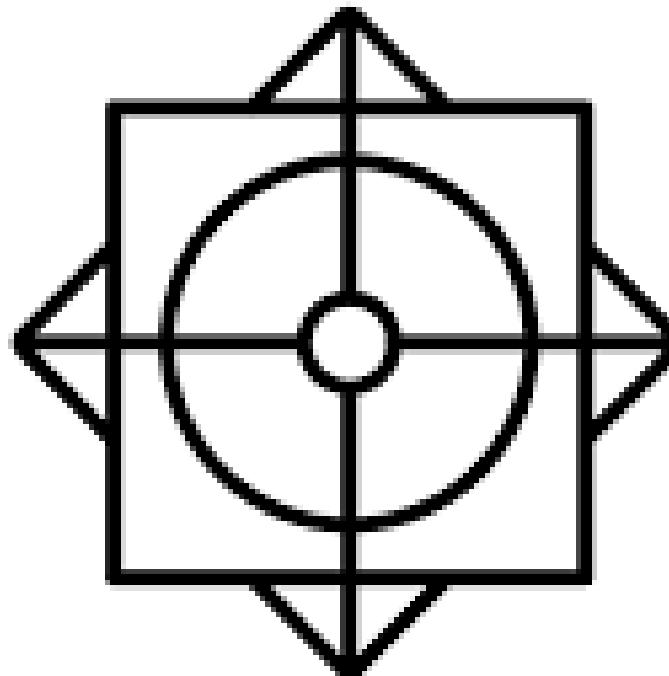
Diffuser les connaissances de la CAA

Avancer les connaissances et méthodes de la CAA

Développer des outils de CAA libres et accessibles



# InterAACtionBox



## InterAACtionBox



Polyhandicap

Open-Source

Développer la communication

Enfants et adultes

Gratuit

Applications de CAA

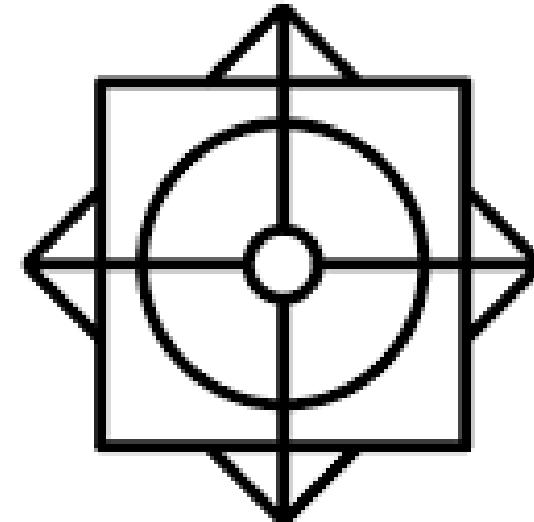
Interactions oculaires

# Occulomètre

**Tobi Eye Tracker 5**



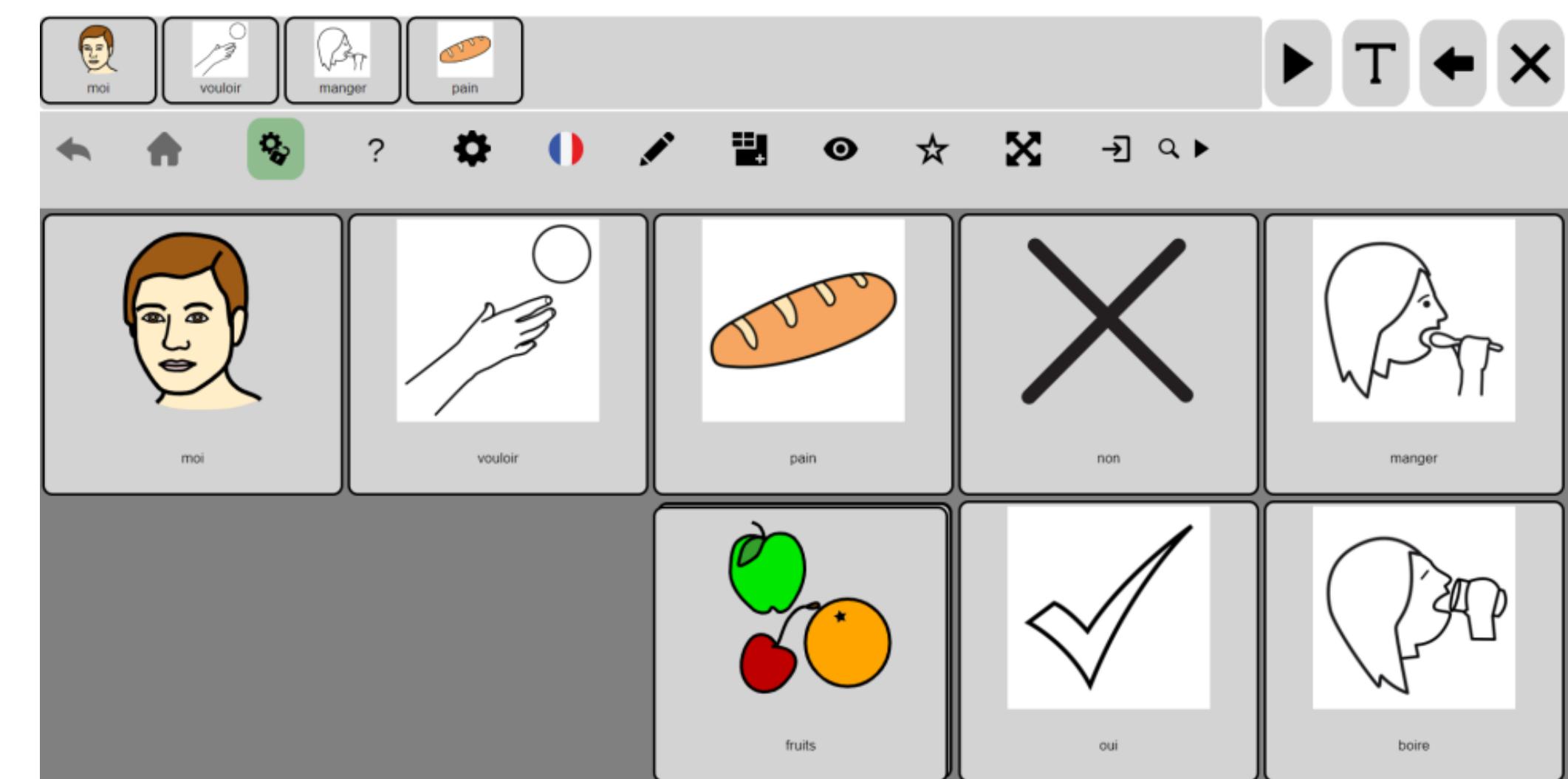
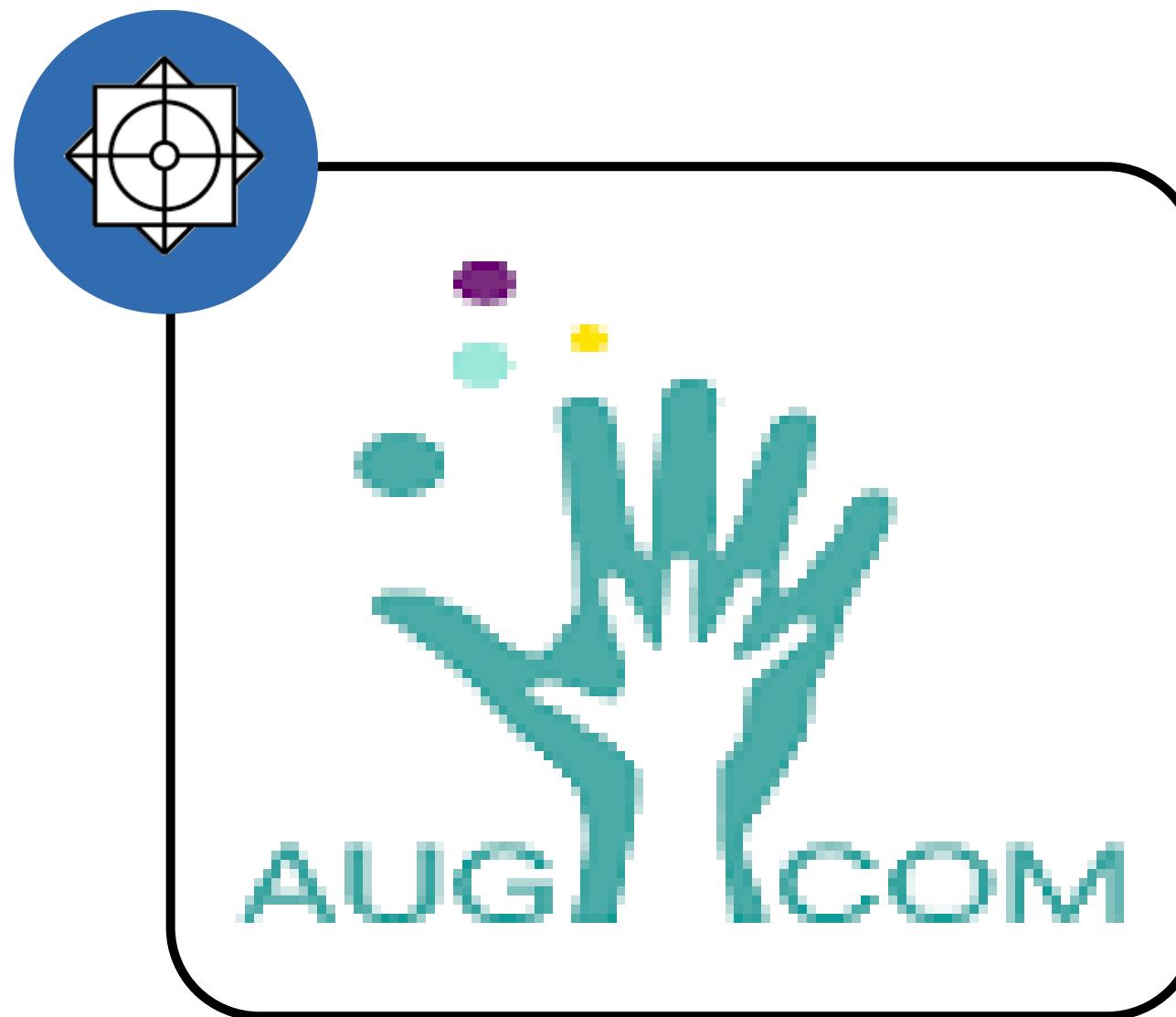
# Les applications de



InterAACtionBox   
**AFSR** [www.afsr.fr](http://www.afsr.fr)



# AugCom



# Gaze Play

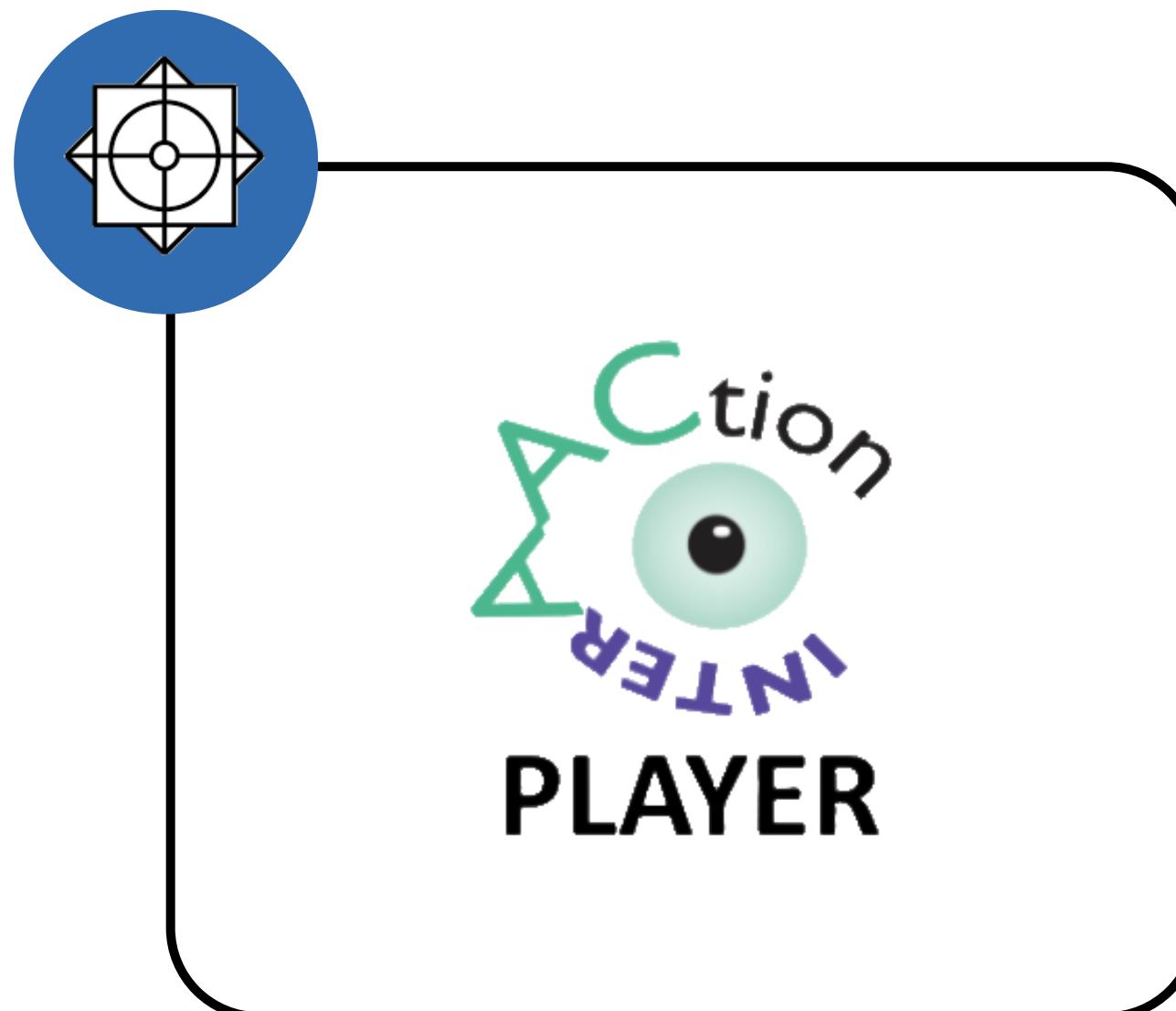


60 jeux éducatifs

Éveille et développe la  
communication des  
enfants

Interaction par le  
regard

# InterAACTION Player



# InterAActionScene



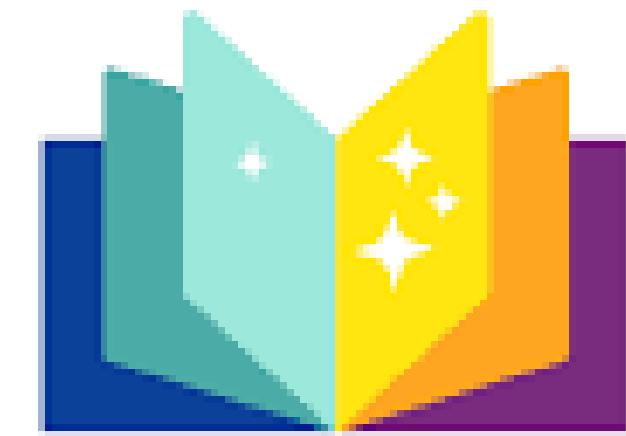
**Créer des scènes visuelles à partir  
d'images**

**Interagir avec les images**

**Apprentissage et Communication**

**Améliorer les interactions sociales**

# Sujet de Stage



## InterAActionScene

Nouvelles fonctionnalités

Tests



13

# Démonstration



# Tech. et outils utilisés

## Gestion de projet



Discord

Échanges informels au sein de l'équipe



GitHub

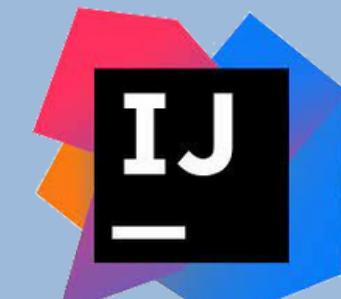
Gérer le développement du code du projet



Zoom

Réunions hebdomadaires

## Développement



IntelliJ

Environnement de développement



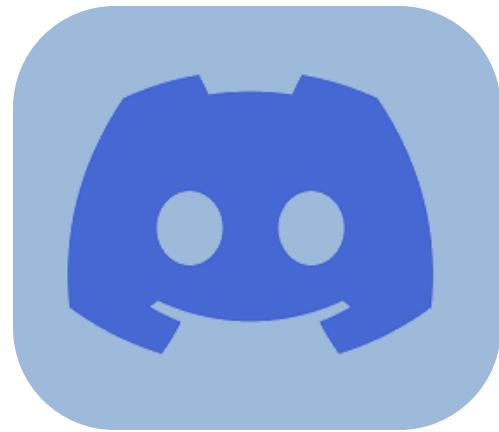
Angular CLI

Framework de développement Web

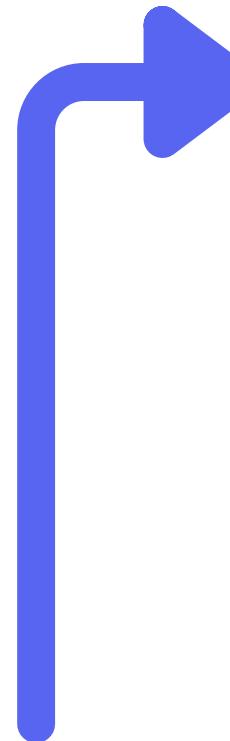


Framework de tests

# Gestion de projet



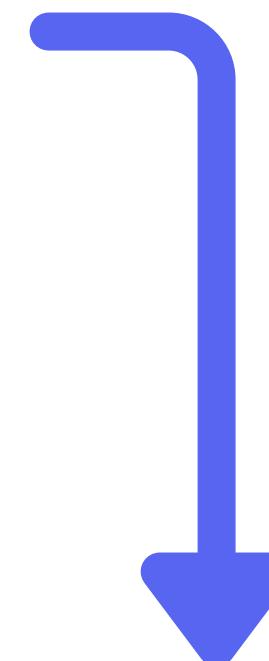
Attribution / Identification de tâches



Rapport de tâches



Réalisations des tâches



# Gestion de projet



Publication des réalisations

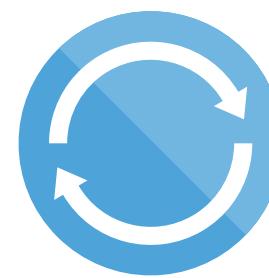
in Pull requests



ou



ou



# Angular et Karma/Jasmine



## Tour of Heroes

### Tour of Heroes

Dashboard   Heroes

#### Heroes List

12	Dr. Nice	X
13	Bombasto	X
14	Celeritas	X
15	Magneta	X
16	RubberMan	X
17	Dynama	X
18	Dr. IQ	X
19	Magma	X
20	Tornado	X

### Tour of Heroes

Dashboard   Heroes

#### CELERITAS Details

id: 14

Hero name: Celeritas

save

go back

Hero name:

Add hero

## Karma/Jasmine



Illustration by [Ekaterina Rogova](#) from [Ouch!](#)



*Réalisation n°1 et prise en main du projet*

# Supprimer toutes les scènes



# Supprimer toutes les scènes





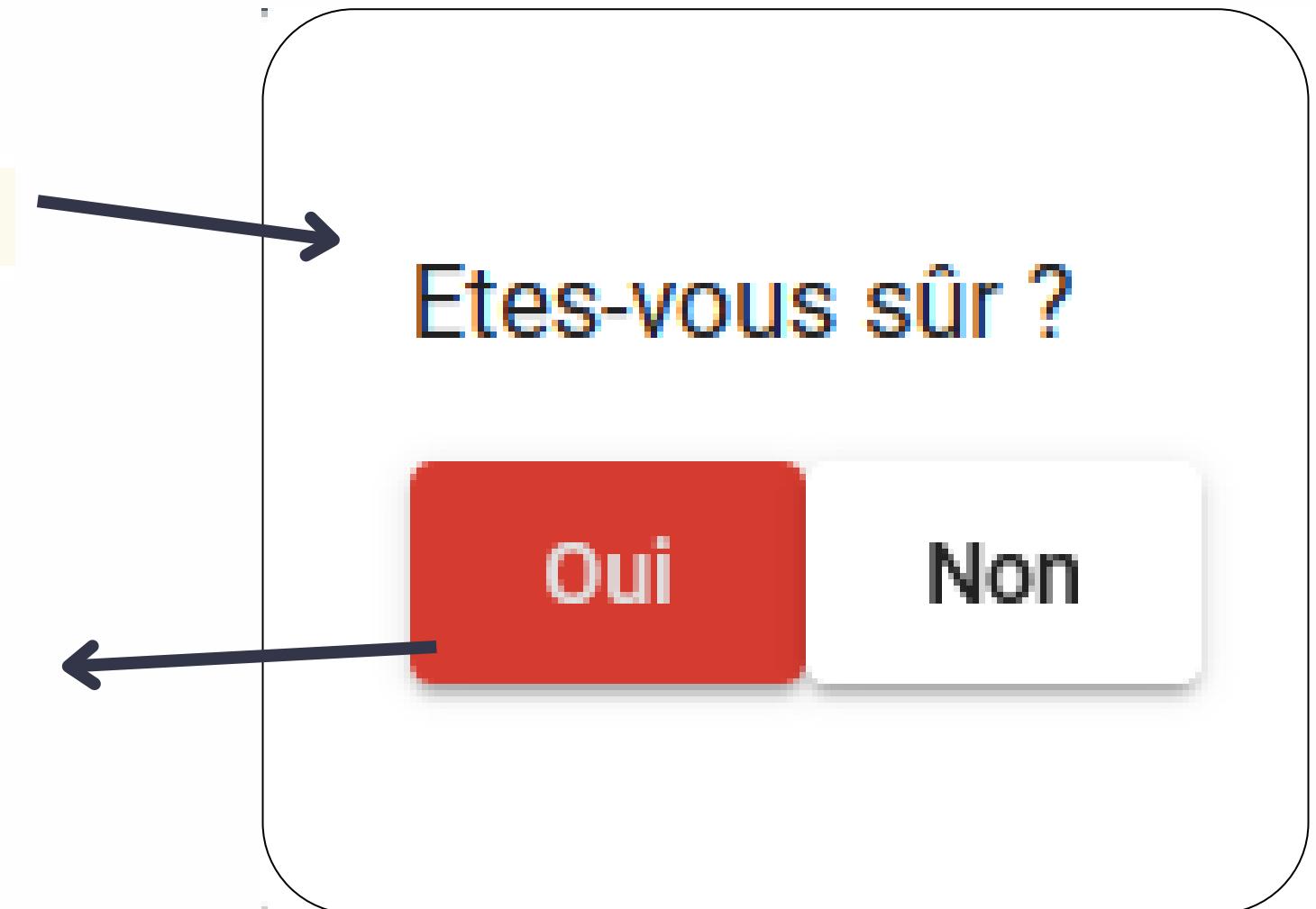
# Supprimer toutes les scènes



```
<button [disabled]!="!(scenesService.hasAtLeastOneScene())" (click)="openDialogReset()" class="button">
  
  <span>{{'Dashboard.Reset' | translate }}</span>
</button>
```



```
(click)="openDialogReset()" class="button">>
```



```
(click)="resetScenes()"
```



```
resetScenes(): void {  
    this.sceneService.removeAll();  
}
```



```
removeAll() {  
    this.SCENES = [];  
    this.update();  
}
```



# Supprimer toutes les scènes

The screenshot shows a user interface for managing scenes. At the top, there is a navigation bar with five items: Jouer (highlighted in blue), Hotspots, Scènes, Dessiner, and Recommander. Below the navigation bar is a grid of cards. The first two cards in the top row are labeled "Scène" and "Image", both featuring a large black plus sign inside a circle. The second card is highlighted with an orange border. The third row contains four cards: "villes" (with a thumbnail of a building), "vif" (with a thumbnail of a building), "dragon" (with a thumbnail of a dragon), and "varces" (with a thumbnail of a landscape). To the right of the grid is a large, light-blue rectangular area. In the bottom right corner of this area, there is a small, separate video player window displaying a video of a building with a blue play button.



*Réalisation n°2*

# Ajout de nouvelles zones d'intérêts



# *Navigation entre les images*

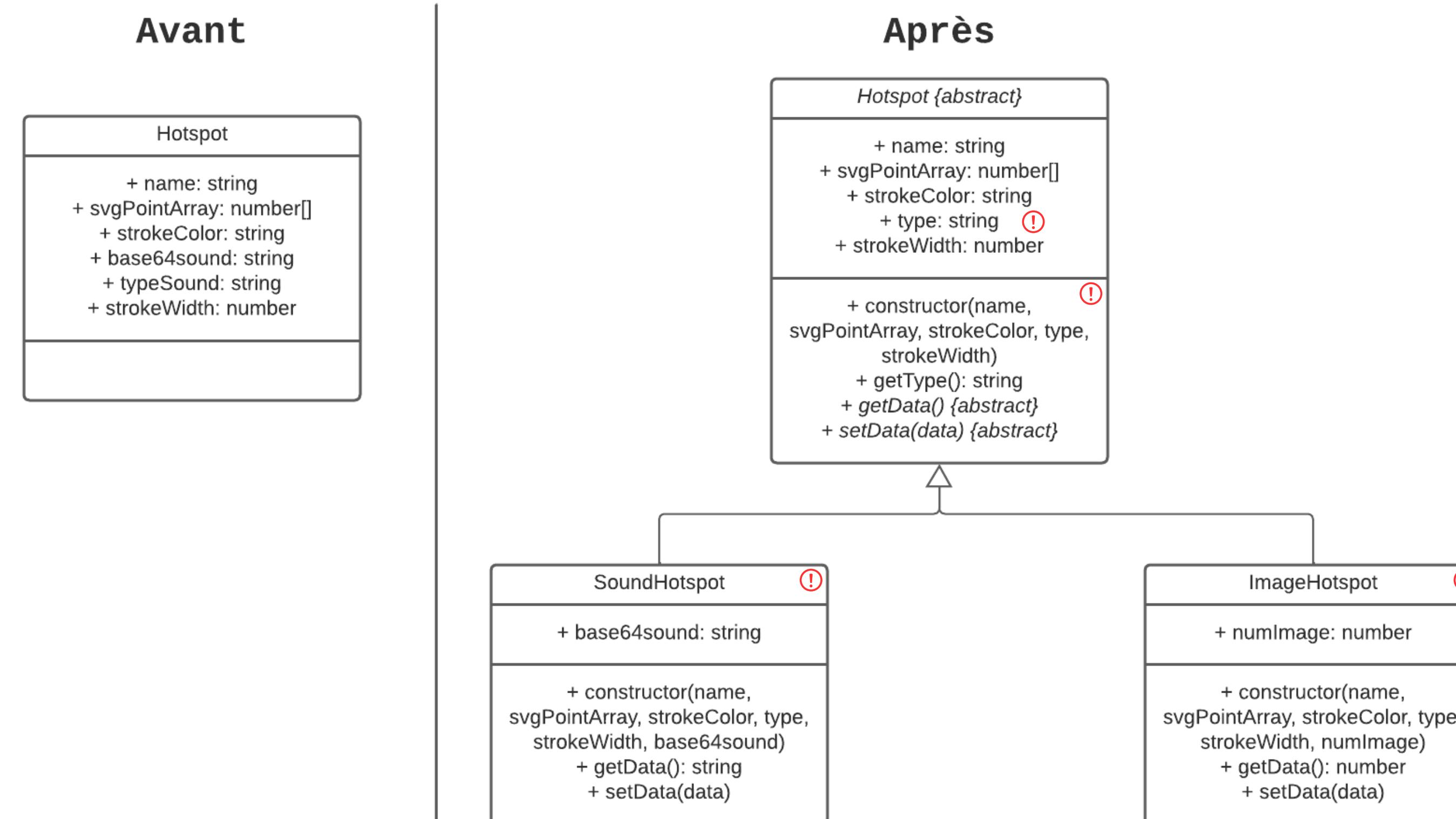
## Ajout de nouvelles zones d'intérêts



*Pour ce faire ...* **26**



# Ajout de nouvelles zones d'intérêts



*Création***Créer hotspot**

Ajouter un son

**Importer un son****Enregistrer un son****Écrire un son****Parcourir...** Aucun fichier sélectionné.

Selectionner la couleur

**Bleu**

Entrer la taille du trait

**2**

Entrer le nom du hotspot

**Importer****Annuler**



# Ajout de nouvelles zones d'intérêts

*Création*

Créer hotspot

New!

Ajouter un son

Ajouter une référence

Importer un son

Enregistrer un son

Écrire un son

Parcourir... Aucun fichier sélectionné.

Selectionner la couleur

Bleu

Entrer la taille du trait

2

Entrer le nom du hotspot

Importer

Annuler

New!

Créer hotspot

New!

Ajouter un son

Ajouter une référence

Choisir l'image

Plaine

Selectionner la couleur

Bleu

Entrer la taille du trait

2

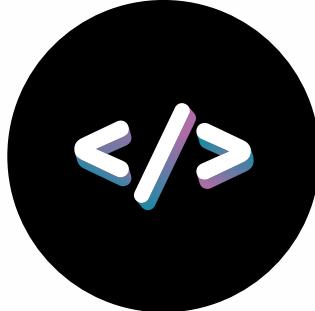
Entrer le nom du hotspot

Importer

Annuler

## Création

```
let hotspot = new ImageHotspot(hotspotName, svgPath, strokeColor, type, strokeWidth, numImage);
```



```
let hotspot = new SoundHotspot(hotspotName, svgPath, strokeColor, type, strokeWidth, base64sound);
```

**Modification****Modifier l'hotspot**

Ajouter un son

**Importer un son****Enregistrer un son****Écrire un son****Parcourir...** Aucun fichier sélectionné.

Selectionner la couleur

**Bleu**

Entrer la taille du trait

**2**

Entrer le nom du hotspot

**serg****redessiner****Modifier****Annuler**



# Ajout de nouvelles zones d'intérêts

**Modification**

## Modifier l'hotspot

[Ajouter un son](#) [Ajouter une référence](#)

[Importer un son](#)

[Enregistrer un son](#)

[Écrire un son](#)

[Parcourir...](#) Aucun fichier sélectionné.

Selectionner la couleur

Bleu

Entrer la taille du trait

2

Entrer le nom du hotspot

Maison

redessiner

**New!**

## Modifier l'hotspot

[Ajouter un son](#) [Ajouter une référence](#)

Choisir l'image

Maison

Selectionner la couleur

Bleu

Entrer la taille du trait

2

Entrer le nom du hotspot

Maison

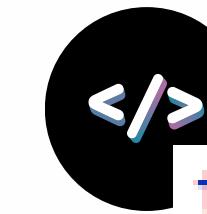
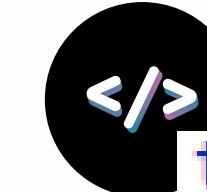
**New!**

[Modifier](#)

[Annuler](#)

[Modifier](#)

[Annuler](#)

**Modification****Changement de type**`this.hotspot = new ImageHotspot(...)``this.hotspot = new SoundHotspot(...)`**Identification du type initial**

```
</>  
if (this.hotspot instanceof SoundHotspot) {  
  
</>  
else if (this.hotspot instanceof ImageHotspot) {
```

**Pas de changement**`this.hotspot.setData`

## Exécution



```
PlayAudio(hotspot: Hotspot) {
    if (hotspot.typeSound == "soundAudio"){
        this.audioPlayer.src = hotspot.base64sound;
        this.audioPlayer.load();
        this.audioPlayer.play();
        this.audioPlayer.volume = this.settingsService.VOLUME;
    }else if (hotspot.typeSound == "writeAudio"){
        let speak = new SpeechSynthesisUtterance(hotspot.base64sound);
        speak.lang = this.languageService.activeSpeechSpeakerLanguage;
        window.speechSynthesis.speak(speak);
    }
}
```

## Avant



```
PlayHotspot(hotspot: Hotspot) {
    if (hotspot instanceof SoundHotspot) {
        if (hotspot.type === "soundAudio") {
            this.audioPlayer.src = hotspot.base64sound;
            this.audioPlayer.volume = this.settingsService.VOLUME;
            this.audioPlayer.load();
            this.audioPlayer.play();
        } else if (hotspot.type === "writeAudio") {
            let speak = new SpeechSynthesisUtterance(hotspot.base64sound);
            speak.lang = this.languageService.activeSpeechSpeakerLanguage;
            speak.volume = this.settingsService.VOLUME;
            window.speechSynthesis.speak(speak);
        }
    } else if (hotspot instanceof ImageHotspot) {
        this.sceneDisplayService.selectedImage = hotspot.numImage;
        this.scenesService.updateScenes();
        this.sceneDisplayService.UpdateDimensions();
    }
}
```

## Après

New!

**Plein d'autres fonctionnalités...**

*Ajout et modification du panel de dessin*

*Renommer les scènes*

*Zoomer et organiser les images*

*Supprimer tous les hotspots d'une image*

*Mise à jour du tutoriel*

*Écriture des tests*

**En cours**

*Remplir automatiquement les champs de nom*

**Séminaires**

*L'approche syntaxique dans la qualité logicielle*

par Xavier Blanc

*Projet d'améliorations de l'outil Whisper*  
par le groupe de lecture du groupement de recherche  
TAL

# Conclusion

Nouvelles technologies



Développement Web



# Conclusion



Université Grenoble Alpes

 **Département  
INFO**

**Alternance**

Merci de votre attention !