



# Interaction multimodale & Handicap

---

## Objectif

L'objectif du TP est de vous familiariser avec le développement de ressources numériques accessibles.

---

## Documents de référence utilisés

Il existe de nombreuses ressources en ligne permettant de comprendre les enjeux de l'accessibilité et les différentes solutions proposées

- **Accessibility Cheatsheet** : <https://moritzgiessmann.de/accessibility-cheatsheet/>
- **Getting Started with Website Accessibility** : <https://medium.com/statuscode/getting-started-with-website-accessibility-5586c7febc92>
- **ARIA** (Accessible Rich Internet Application) : <https://www.w3.org/TR/using-aria/>

Si la plupart des solutions exposées semblent l'être pour des mal ou non-voyants, le handicap ne se limite pas à cette problématique tant les obstacles à l'accessibilité et les situations sont nombreux : pensez l'accessibilité, c'est aussi penser à des usages où chacun peut être en situation de handicap (mains, yeux occupés à une autre tâche, bruit ambiant, etc.).

Comprendre la situation, c'est déjà commencer à répondre à la problématique en utilisant le plus souvent des modalités alternatives.

---

## Les solutions techniques : l'exemple des lecteurs d'écran

Essayez un lecteur d'écran (utilisé par les non-voyants). Vous pouvez par exemple télécharger NVDA (Non Visual Desktop Access) ici → <https://www.nvda-fr.org> (pour une version portable, télécharger la version ici : <https://www.softpedia.com/get/PORTABLE-SOFTWARE/System/System-Enhancements/Portable-NVDA.shtml>)

Utilisez NVDA les yeux fermés et tentez de naviguer sur le site de l'université pour y trouver le lien d'accès à l'intranet (en n'utilisant que les **feedbacks vocaux**).

Toujours dans le même contexte, ouvrez ensuite un éditeur de texte et prenez des notes sur votre expérience vécue 😊.

Quels sont les problèmes que vous avez identifiés ? Comment pensez-vous y remédier en utilisant de solutions multimodales ?

---

## Une application à développer

Téléchargez l'archive zip **p5js.zip** à l'adresse

<https://github.com/truillet/upssitech/blob/master/SRI/3A/IHM/TP/Outils/p5js.zip>

**p5.js** (<https://p5js.org/get-started>) est la version « javascript » de Processing qui permet le prototypage rapide d'applications interactives pour le web.

- Dézippez l'archive et lancer **ReconnaissanceVocale\_Simple.html** dans un navigateur web Google Chrome (à cause du système de reconnaissance vocale et de synthèse utilisés), faites de même avec **ReconnaissanceVocale\_Continue.html**.  
Regardez le code et modifiez-le de manière à aider l'utilisateur sans ses actions.
- Testez et comprenez le code écrit dans **Accessibilite.html**

Développez enfin une mini-application interactive (cf. exemples proposés ci-dessous) adaptable à au moins un handicap (par exemple adaptée aux non et mal -voyants, personnes malentendantes, daltoniennes, dyslexiques, ... ) à produire avec le framework **p5.js** (<https://p5js.org>) et les modules **p5.accessibility**, **p5.speech** (voire d'autres disponibles ici) :

**p5.js**

- <https://p5js.org/libraries>
- ou là (TopCodes – Réalité Augmentée) : <https://github.com/sehmon/TIDAL-topcodes-demo>

#### Exemples d'applications à créer :

- un formulaire vocal de demande de renseignements permettant à la fois de naviguer et de remplir les items.
- un jeu de « dictée magique » ([https://fr.wikipedia.org/wiki/Dictée\\_magique](https://fr.wikipedia.org/wiki/Dictée_magique))
- un mini jeu vidéo type « snake »  
([https://fr.wikipedia.org/wiki/Snake\\_\(genre\\_de\\_jeu\\_vidéo\)](https://fr.wikipedia.org/wiki/Snake_(genre_de_jeu_vidéo)))
- un jeu musical type « Simon » ([https://fr.wikipedia.org/wiki/Simon\\_\(jeu\)](https://fr.wikipedia.org/wiki/Simon_(jeu)))...

---

**Le travail sera à rendre après cette séance pour être évalué dans le cadre de la partie « Mineure Interaction »**