## Pins documentation

Objet 3D interactif pointant sur des centres d’intérets dans la scène 3D de UD-Viz. Ces éléments interactifs sont representés par une bulle d’image ainsi qu’une épingle ; l’image permet de donner un aperçu sur le contenu qui sera afficher si l’utilisateur intéragit avec en cliquant sur celui-ci.

Chacun de ces éléments est un contenu d’un épisode et un épisode comprends plusieurs contenus. Ce contenu permet de détailler une zone spécifique sur une maquette en y ajoutant du texte ou une image.

### Technical details

Un pin est un contenu d’un épisode qui est matérialisé par la classe EpisodeContent.jset contient toutes les informations sur un contenu d’un épisode. La classe EpisodeVisualizer.js comprend la liste des contenus de l’épisode et donc une liste d’objet EpisodeContent qui apparaissent dans la scène 3D.

#### configEpisodes.json :

Le configEpisodes.json est le fichier JSON qui permet de configurer tout le contenu de votre épisode. C’est ici que vous allez modifier les sources d’images, sa position etc… et doit suivre le template suivant :

* lock : L’élément 3D peut être vérrouillé ou dévérouillé en fonction de sa valeur dans le fichier de config. Cette valeur dépend d’ou se trouve l’utilisateur dans le visionnage d’un épisode. Le contenu pourra être accessible que si celui-ci a une valeur à **false**
* position : la position georéférence dans votre scène avec son type de projection (EPSG:3946).
* imgUnLock : le chemin en direction votre image dévérouillé.
* imgLock : le chemin en direction votre image vérouillé.

Une fois dévérouillé, l’épingle est accessible et intéractive pour afficher le contenu de ce point d’intéret plus en détail. Une fenêtre html apparait et donne une description de la zone que pointe l’élément.

#### Code example

Exemple de code pour créer 3 contenu d’un épisode et l’afficher dans la scène.

let content\_1 = new EpisodeContent(configEpisode['episode-1-data']['content-1']);  
 let content\_2 = new EpisodeContent(configEpisode['episode-1-data']['content-2']);  
 let content\_3 = new EpisodeContent(configEpisode['episode-1-data']['content-3']);  
 let listContents = [content\_1,content\_2,content\_3];  
   
 const episode\_1 = new EpisodeVisualizer('episode\_1', view3D, listContents);   
 episode\_1.constructAllContent();