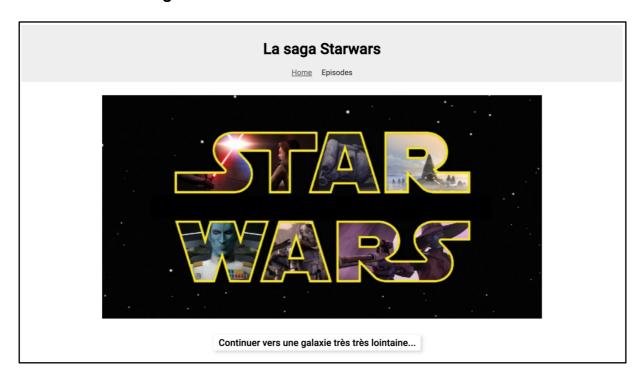
## TP 2 - Angular

## **Exercice 1: Services**

- 1. Créer un composant episode-list et modifier l'application pour qu'il soit le parent de tous les éléments EpisodeComponent.
- 2. Créer un dossier services puis un fichier episodes.service.ts
- 3. Créer un service episodeService contenant les méthodes suivantes :
  - a. getAllEpisodes : qui permet de récupérer tous les spots de surf
  - b. getEpisodeById : qui permet de récupérer un épisode par son ld
  - c. likeEpisodeById : qui permet d'ajouter ou de retirer un "Like" à un épisode
- 4. Modifier l'application en utilisant les méthodes lorsqu'elles sont pertinentes.

## **Exercice 2: Routing**



- 1. Créer un composant landing-page qui servira de page d'accueil à l'application.
- 2. Créer un module de routing en créant un fichier app-routing.module.ts dans le dossier app.
- 3. Associer LandingPageComponent à la route vide.

4. Créer deux liens dans le header pour permettre à l'utilisateur de changer de route.

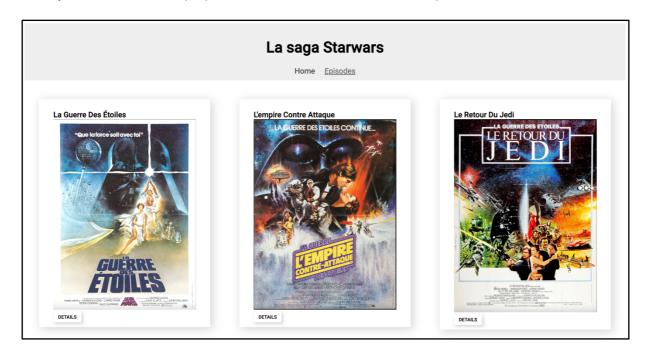




- 5. Mettre en place un style particulier pour indiquer la route active.
- 6. Créer un bouton sur la page d'accueil qui permet de rentrer dans l'application en appliquant la navigation programmatique.

## **Exercice 3: Routing (2)**

- 1. Créer un composant episode-detail et y copier tout depuis EpisodeComponent: template, styles et Typescript.
- 2. Créer la route qui appellera EpisodeDetailComponent en utilisant un paramètre dynamique qui contiendra l'Id de l'épisode.
- 3. Modifier EpisodeComponent pour ne plus afficher que le titre et l'affiche, puis ajouter un bouton qui permet d'accéder au détail de l'épisode.



4. Ajouter un bouton qui permet de retourner à la liste des épisodes.

