

# README

---

## RIA1\_EVAL\_PLAYLIST

---

RIA1 EVAL PLAYLIST est un projet permettant d'évaluer les compétences des technicien-ne-s en JavaScript.

### Evaluation

Les compétences évaluées sont les suivantes:

- Dessiner un diagramme de classe en partant d'un code existant.
- Coder en partant des tests.
- Appliquer les bonnes pratiques (OOP, git, TDD).

### Installation du projet

Pré-requis :

- npm 6.14.13 ou ultérieure
- node v14.17.3 ou ultérieure
- git version 2.34.1.windows 1 ou ultérieure

Note : configuration testée et validée sur l'IDE WebStorm : version 2021.3.1

Après avoir récupéré le référentiel:

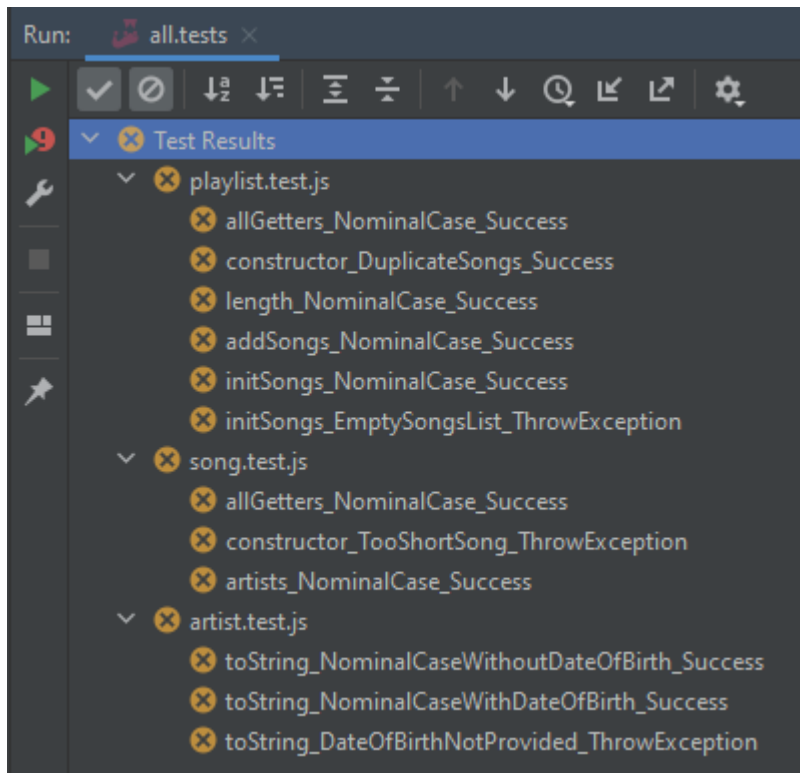
```
npm install
```

### Usage

Pour tester tout le projet :

```
npm run test
```

Le résultat a obtenir en début de projet :



Pour tester une seule classe :

```
npm run test [className.test.js]
```

## Prise en main du projet

Une fois le code récupéré, il s'agit de le *forker* sur un référentiel vous appartenant. Puis de traiter les classes de tests disponibles.

## Aides

Voici des sources qui vous aideront à vous lancer :

- [Les classes en Java Script](#)
  - [Les accesseurs](#)
  - [Les mutateurs](#)
  - [Exception](#)
  - [Les dates](#)
- [Premier pas avec Jest](#)
  - [Les comparateurs](#)
  - [Les exceptions](#)
- [Convention de nommage en Java Script](#)

## Contributing

Lors de la lecture en commun des tests, vos retours sont les bienvenus.

## Règles du jeu

- Vous disposez du code produit en classe tout comme d'un accès libre à internet.
- Le travail collaboratif n'est pas admis dans cette épreuve (chat sur des forums ou entre collègues par exemple)
- L'enseignant réalisera une vidéo de tous les écrans, à des fins de contrôle.

## Livraison

- Un zip respectant les critères suivants:
  - Nommage = RIA1\_EVAL01\_[IntranetFirstFamilyName].zip
  - Livré dans teams, via un message privé au scrum master (qui vous quittancera la bonne réception).
  - Les fichiers en lien avec git. permettant de voir la bonne utilisation du versioning.
- La branche qui sera prise en compte pour l'évaluation est la "develop"

## Evaluation

- Un point par test réussi. La note sera calculée ainsi  $(\text{nb test réussis} / \text{nb test tot}) * 5 + 1$
- Bonus:
  - Des commits réguliers et soignés.
  - Une approche TDD (coder, tester, coder, tester).
  - Du code qui ne fonctionne pas mais accompagné de votre réflexion/analyse.
  - La qualité et le soin du code.

L'examen vaut pour 30% de la partie pratique du module.

## License

Cet examen est la propriété du CPNV et de son auteur.

## Source

[Make a readme](#)