

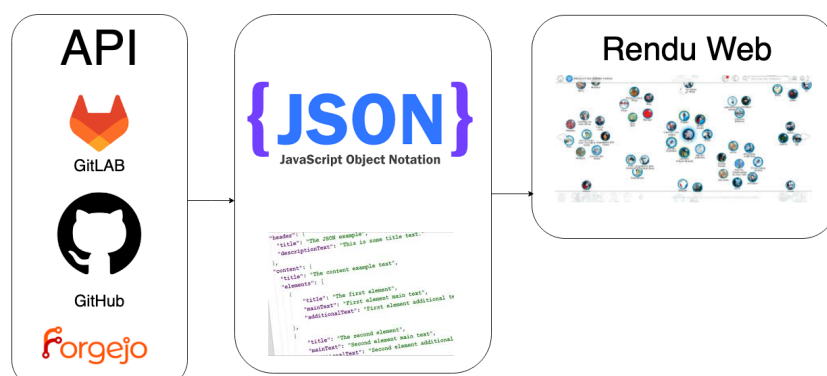
JSON2Pearl

JSON2Pearl : développer une application Angular permettant de représenter sous la forme de perles la sortie JSON d'une API de Forge Logicielle (GitHub). Ce travail fait partie d'un projet de recherche qui vise à rendre les forges plus adaptées et faciles à utiliser par des enseignants qui y déposent leurs ressources pédagogiques (cours, exercices) et c'est pour cela que nous cherchons à tester de nouvelles façons plus visuelles de représenter les documents dans une forge.

Application destinée à être intégrée à l'application ForgeEditor¹ proposée par le Groupe Thématique Numérique Forges² dont le but est de favoriser l'utilisation de la Forge des Communs Numériques Éducatifs³ par les enseignants pour la création, le partage et la maintenance de ressources éducatives libres⁴.

Objectifs :

- Fournir une application web permettant de représenter sous la forme de perles la sortie JSON d'une API de "Forge Logicielle"(GitHub, GitLab, Forgejo)



Travaux demandés:

Étude comparative des différentes "bibliothèques JavaScript" disponibles pour réaliser le rendu graphique.

¹ ForgeEditor : <https://forgeeditor-0e3372.forge.apps.education.fr/recherche?category=projects>

² GNum Forges: <https://gtnum.forge.apps.education.fr/>

³ Forge des communs numériques éducatifs : <https://docs.forge.apps.education.fr/>

⁴ Ressources éducatives libres : <https://www.unesco.org/fr/legal-affairs/recommendation-open-educational-resources-oer>

Utilisation du framework Angular 19 pour la réalisation d'un composant générique capable de communiquer avec les différents types de forges logicielles (GitHub, GitLab, Forgejo)

Réalisation d'une application web pour afficher

Fonctionnalités attendues :

- Permettre à l'utilisateur de choisir le type de forge et l'adresse de la forge à interroger.
- Offrir plusieurs requêtes disponibles (Projet, Sujet, Utilisateur).
- Proposer différentes représentations graphiques pour l'affichage selon un critère de sélection

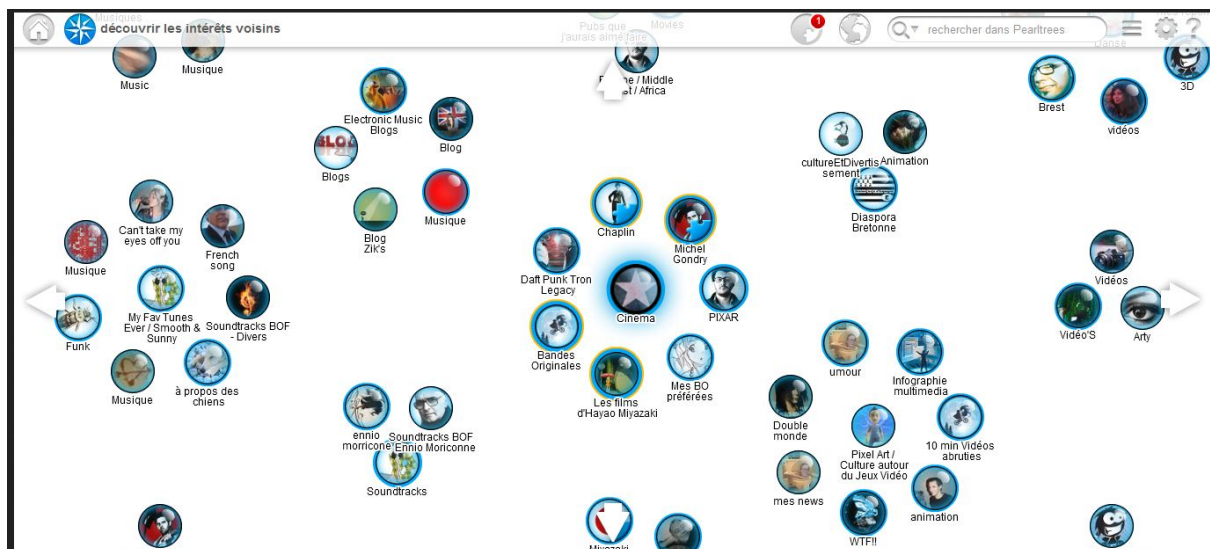
Ressources à consulter :

- GitLAB : <https://docs.gitlab.com/api/rest/>
- Github : <https://docs.github.com/en/rest?apiVersion=2022-11-28>
- Forgejo : <https://forgejo.org/docs/latest/user/api-usage/>

JSON⁵

```
0: { id: 3684, description: "Coordonnées d'un dessin repéré du personnage Komit, animal-totem de la forge des communs numériques éducatifs (cycle 4 du collège, niveau 5e)", name: "perlomathsd-r-komit", ... }
1: { id: 3362, description: "Coordonnées d'un dessin repéré de la tête d'un chien (cycle 4 du collège, niveau 5e)", name: "perlomathsd-r-chien", ... }
2: { id: 3336, description: "Coordonnées d'un dessin repéré du personnage 'Tomate' du webcomic libre / open source Pepper&Carrot (cycle 4 du collège, niveau 5e)", name: "perlomathsd-r-tomate", ... }
3: { id: 3335, description: "Coordonnées d'un dessin repéré du personnage 'Oiseau' du webcomic libre / open source Pepper&Carrot (cycle 4 du collège, niveau 5e)", name: "perlomathsd-r-oiseau", ... }
4: { id: 3334, description: "Coordonnées d'un dessin repéré du personnage du 'Boss final' du webcomic libre / open source Pepper&Carrot (cycle 4 du collège, niveau 5e)", name: "perlomathsd-r-bossfinal", ... }
5: { id: 3333, description: "Coordonnées d'un dessin repéré du personnage Zangemann du conte libre / open source Ada&Zangemann (cycle 4 du collège, niveau 5e)", name: "perlomathsd-r-zangemann", ... }
6: { id: 3332, description: "Coordonnées d'un dessin repéré du personnage Ada du conte libre / open source Ada&Zangemann (cycle 4 du collège, niveau 5e)", name: "perlomathsd-r-ada", ... }
7: { id: 3330, description: "Coordonnées d'un dessin repéré du personnage Carrot du webcomic libre / open source Pepper&Carrot (cycle 4 du collège, niveau 5e)", name: "perlomathsd-r-carrot", ... }
8: { id: 2253, description: "Le logiciel perlomathsd-r permet de générer des dessins repérés (consigne et correction). Ce logiciel a été développé avec L'c L'ENC et avec L'EAC. Notions aussi abordées : Droit d'auteur, licences et culture libre, celle des communautés libristes.", name: "perlomathsd-r", ... }
9: { id: 1833, name: "JackpotAffine", name_with_namespace: "Degrangem / JackpotAffine", ... }
10: { id: 1285, description: "Secret Not Tellable est un jeu sérieux dans lequel les élèves doivent trouver un mot de passe respectant certaines contraintes _ liées au programme de SNT 1", name: "Secret Not Tellable", ... }
11: { id: 1034, description: "Un site pour expliquer ce que sont les Ressources Éducatives Libres : \nhttps://drane-lyon.forge.apps.education.fr/rel", name: "Ressources Éducatives Libres", ... }
```

pearl



⁵ JSON: https://developer.mozilla.org/fr/docs/Web/JavaScript/Reference/Global_Objects/JSON

<https://www.pearltrees.com/fr/>

Environnement technologique :

- Framework javascript - Angular (<https://Angular.dev>)
- API (REST- GRAPHQL)
- JSON
- Librairie Javascript graphiques (OpenSource)

Organisation

■ Planification

- Diagramme de Gantt⁶ prévisionnel pour l'ensemble des quatre semaines
- Tableau de répartition des tâches entre les étudiants
- Maquette graphique⁷ de la solution envisagée

■ Communication :

- Utilisation de Mattermost pour la communication
- Tenue d'un journal quotidien sur la forge de l'université ⁸(Objectifs de la journée, tâches réalisées Problèmes rencontrés) au format Markdown⁹
- Tenue d'un diagramme de Gantt rectificatif
- Réunion hebdomadaire -- mercredi matin à 10h30

■ Programmation :

- Utilisation du Framework Javascript Angular 19 pour la réalisation de l'application demandée
- Interrogation de L'API REST¹⁰ et/ou GraphQL¹¹ de GitLab
- Utilisation de Git¹² pour la gestion de version de l'application
- Utilisation de la forge de l'université pour le suivi du projet et la conservation du code de l'application

⁶ Diagram de Gantt : https://fr.wikipedia.org/wiki/Diagramme_de_Gantt

⁷ Éditeur graphique Draw.io : <https://www.drawio.com/>

⁸ Forge de l'université : <https://git.univ-lemans.fr/tfore/json2pearl>

⁹ Markdown : <https://www.markdownguide.org/getting-started/>

¹⁰ API REST GitLab: <https://docs.gitlab.com/api/rest/>

¹¹ API GraphQL : <https://docs.gitlab.com/api/graphql/>

¹² Outil de gestion de versions - Git: <https://git-scm.com/>

Descriptif de l'application des compétences :

- Compétence 1==
 - AC1 = Choisir et implémenter les architectures adaptées
 - Choix des structures de données (Listes, collection , etc..)
 - Respecter une organisation Model/ VUE /Contrôleur (MVC)¹³ pour le développement de l'application
 - AC2 = Intégrer des solutions dans un environnement de production
 - Extraction de données à partir d'API REST et GRAPHQL
 - Framework javascript Angular Imposé pour l'implémentation
 - Objectif ==> proposer à l'issue du stage une librairie de composants (open source) ex: Angular Material¹⁴
- Compétence 2 ==
 - Anticiper les résultats de diverses métriques (temps d'exécution, occupation mémoire))
 - Utiliser les outils disponibles pour les navigateurs¹⁵
 - Choisir et utiliser de bibliothèques et méthodes dédiées au domaine d'application (..)
 - => Faire une étude comparative des bibliothèques graphiques disponibles
 - ==> Proposer une solution argumentée sur la bibliothèque que vous aurez choisie
- Compétence 6 ==
 - Organiser et partager une veille numérique
 - Choisir un outil pour assurer votre veille technologique, présenter une proposition argumentée pour justifier le choix de votre outil de veille technologique¹⁶.

¹³ Modèle-vue-contrôleur (MVC): <https://fr.wikipedia.org/wiki/Mod%C3%A8le-vue-contr%C3%B4leur>

¹⁴ Angular Material <https://material.angular.dev/>

¹⁵ Angular DEvTool: <https://angular.dev/tools/devtools>

¹⁶ Veille technologique collaborative -YesWiki : <https://yeswiki.net/?PagePrincipale>