

---

# Travail pratique : Créer une fiction interactive multi-plateforme (Laravel + Vue.js)

## Table des matières

- [Table des matières](#)
- [Introduction](#)
- [Objectifs pédagogiques](#)
- [Consignes générales](#)
- [Évaluation](#)
  - [Critères Backend](#)
  - [Critères frontend](#)
- [Contraintes techniques](#)
- [Conseils](#)
- [Livrables et rendu](#)
- [Feedback](#)

## Introduction

Ce projet vous invite à créer une application complète mêlant **backend Laravel** et **frontend Vue.js**, sous forme d'une fiction interactive où les choix influencent le récit.

L'objectif est de concevoir un système de navigation dynamique, stocké en base de données, et rendu disponible à travers une API REST que le frontend consommera.

## Objectifs pédagogiques

À l'issue de ce travail pratique, les personnes qui étudient devraient être capables de :

- Concevoir et développer une application web complète de type SPA (Single Page Application).
- Implémenter un backend Laravel structuré avec des routes RESTful.
- Créer une API versionnée avec validation, middleware, et relations Eloquent.
- Gérer un frontend Vue.js pour naviguer dans une histoire à choix multiples.
- Interagir avec une API (GET, POST, etc.) et afficher dynamiquement les contenus.

---

## Consignes générales

Vous développerez une application web en deux parties :

- **Backend Laravel** : responsable de la gestion des histoires, chapitres, choix et authentification.
- **Frontend Vue.js** : responsable de l'affichage interactif de la fiction, utilisable sur navigateur et mobile.

Le thème "histoire interactive" est recommandé mais non obligatoire. Si vous avez une idée de projet plus personnel, validez-la avec l'équipe enseignante.

Ex. : une application dans laquelle une personne lit un paragraphe et doit choisir entre plusieurs actions ("entrer dans la grotte", "fuir", etc.), chaque choix menant à un autre chapitre.

Exemples:

- <https://book.onivers.com/book/nc>
- Life is Strange, DONTNOD Entertainment
- <https://www.choiceofgames.com/>

Outils pour créer des histoires interactives : <https://twinery.org/>

## Évaluation

Chaque partie du projet sera évaluée selon plusieurs catégories. Le barème repose sur plusieurs critères, chacun valant 2 point.

- 0 point - Le travail est insuffisant
- 1 point - Le travail est réalisé
- 2 points - Le travail est bien réalisé (sans la nécessité d'être parfait)

Les parties frontend et backend sont notées séparément.

Note maximale : (nombre de points obtenus / nombre de points maximum) × 5 + 1

## Critères Backend

## Critères rendu

---

| #     | Critère                                                                               | Points |
|-------|---------------------------------------------------------------------------------------|--------|
| 1     | Migrations, modèles et relations correctement définis (Story, Chapter, Choice, etc..) | 2      |
| 2     | Validation via <code>FormRequest</code> pour les opérations de création/modification  | 2      |
| 3     | API RESTful versionnée ( <code>/api / . . .</code> ) avec routes bien nommées         | 2      |
| 4     | Contrôleurs correctement structurés, logique métier claire                            | 2      |
| 5     | Authentification Laravel implémentée                                                  | 2      |
| 6     | Middleware protégeant les routes d'édition/ajout                                      | 2      |
| 8     | Gestion des erreurs API (404, 422, etc.)                                              | 2      |
| 9     | Réponses structurées (JSON clair, status code adaptés)                                | 2      |
| 10    | Code lisible, commenté, structuré                                                     | 2      |
| 11    | Le projet est fonctionnel de bout en bout                                             | 2      |
| 12    | README structuré expliquant le projet et son installation                             | 2      |
| 13    | Utilisation du contrôle de version (commits clairs, branches si possible)             | 2      |
| TOTAL |                                                                                       | 26     |

---

**Critères présentation** (Les informations ci-dessous sont à titre indicatif et peuvent être adaptées)

| #     | Critère                                                              | Points |
|-------|----------------------------------------------------------------------|--------|
| 1     | L'élocution est claire et compréhensible                             | 2      |
| 2     | Les informations sont claires et bien présentées                     | 2      |
| 3     | Les contenus présentés sont pertinents et supportent l'argumentation | 2      |
| 4     | Le temps imparti est respecté                                        | 2      |
| 5     | Les réponses aux questions sont pertinentes                          | 2      |
| 6     | Les retours sont écoutés et pris en compte                           | 2      |
| TOTAL |                                                                      | 12     |

---

**Critères frontend**

(Les informations ci-dessous sont à titre indicatif et peuvent être adaptées)

---

**Critères rendu**

| #     | Critère                                                                   | Points |
|-------|---------------------------------------------------------------------------|--------|
| 1     | Affichage d'une liste d'histoires (récupérées via l'API)                  | 2      |
| 2     | Navigation dynamique entre les chapitres selon les choix                  | 2      |
| 3     | Affichage lisible, clair et réactif du contenu                            | 2      |
| 4     | Interface responsive (mobile/desktop)                                     | 2      |
| 5     | Affichage conditionnel                                                    | 2      |
| 6     | Sauvegarde de progression                                                 | 2      |
| 7     | Code lisible, commenté, structuré                                         | 2      |
| 8     | Qualité de l'expérience utilisateur (design, fluidité)                    | 2      |
| 9     | Gestion correcte des erreurs utilisateur côté frontend (choix invalides)  | 2      |
| 10    | Le projet est fonctionnel de bout en bout                                 | 2      |
| 11    | README structuré expliquant le projet et son installation                 | 2      |
| 12    | Utilisation du contrôle de version (commits clairs, branches si possible) | 2      |
| TOTAL |                                                                           | 24     |

**Critères présentation** (Les informations ci-dessous sont à titre indicatif et peuvent être adaptées)

| #     | Critère                                                              | Points |
|-------|----------------------------------------------------------------------|--------|
| 1     | L'élocution est claire et compréhensible                             | 2      |
| 2     | Les informations sont claires et bien présentées                     | 2      |
| 3     | Les contenus présentés sont pertinents et supportent l'argumentation | 2      |
| 4     | Le temps imparti est respecté                                        | 2      |
| 5     | Les réponses aux questions sont pertinentes                          | 2      |
| 6     | Les retours sont écoutés et pris en compte                           | 2      |
| TOTAL |                                                                      | 12     |

---

## Contraintes techniques

- Backend Laravel  $\geq 10.x$
- Frontend Vue.js  $\geq 3.x$
- Base de données relationnelle (SQLite, MySQL ou PostgreSQL)
- Projet disponible sur GitHub
- Une documentation minimale (README .md) doit permettre de tester facilement l'application
- Vous pouvez stocker le token de session dans un cookie pour gérer l'authentification dans le frontend.

## Conseils

- Ne cherchez pas à faire complexe : commencez simple, itérez ensuite.
- Travaillez de manière incrémentale et validez chaque étape.
- Testez tôt et souvent.
- Un bug non corrigé vaut moins de points qu'une fonctionnalité simple mais fonctionnelle.

## Livrables et rendu

Vous devez fournir :

- L'URL du dépôt GitHub (backend + frontend)
- Un fichier README .md clair pour expliquer l'installation et les choix techniques

**Rendu final** : le lundi 12 mai 2025 à 23h59.

Une présentation aura lieu du côté frontend pour l'examen de WebMobUI.

Les présentations côté backend auront lieu lors de la dernière séance de cours, à la semaine 12 (du 12 au 16 mai 2025). Vous aurez 6 minutes pour présenter votre projet, suivies de 2 minutes de questions. Et se feront devant toute la classe.

La classe 52/1 passera le mardi 13 mai 2025 de 13h15 à 16h30 avec ce planning :

| Heure | Nom          | Prénom |
|-------|--------------|--------|
| 13h15 | Abessolo     | Joshua |
| 13h24 | Bartolacelli | Cédric |
| 13h32 | Bugnon       | Léa    |

---

| Heure | Nom                | Prénom    |
|-------|--------------------|-----------|
| 13h40 | Chappalley         | Robin     |
| 13h48 | Desgalier          | Michaël   |
| 13h56 | Du Bois            | Jonas     |
| 14h04 | Espinasse          | Christel  |
| 14h12 | Hungerbühler       | Alexandre |
| 14h20 | Imfeld             | Théo      |
| 14h28 | Jaccard            | Mathilde  |
| 14h36 | Joliat             | Thibaud   |
| 14h55 | Makumbi Mbuka      | Agathe    |
| 15h03 | Mangold            | Laure     |
| 15h11 | Mayor              | Joanah    |
| 15h19 | Nussbaumer         | Thibaud   |
| 15h27 | Rosset             | Doriane   |
| 15h35 | Rossier            | Estelle   |
| 15h43 | Rurangirwa         | Vincent   |
| 15h51 | Teixeira Gonçalves | Samuel    |
| 15h59 | Tille              | Cédrine   |
| 16h07 | Tranchida          | Anna      |
| 16h15 | Zima-Tukala        | Aurélia   |

La classe 52/2 passera le vendredi 16 mai 2025 de 08h30 à 12h00 avec ce planning :

| Heure | Nom           | Prénom     |
|-------|---------------|------------|
| 08h30 | Bossel        | Malory     |
| 08h38 | Maret         | Maximilien |
| 08h46 | Wojciechowski | Victor     |
| 08h54 | Magnenat      | Loriane    |

---

| Heure | Nom                | Prénom    |
|-------|--------------------|-----------|
| 09h02 | Ançay              | Mathilde  |
| 09h10 | Vecchi             | Valentine |
| 09h18 | Crettex            | Lorie     |
| 09h26 | Pinard             | Jonathan  |
| 09h34 | Bolay              | Estelle   |
| 09h42 | Chautems           | Emma      |
| 09h50 | Al-Dabboussi       | Najwa     |
| 10h25 | Degirmenci         | Jasmine   |
| 10h33 | Pottier            | Cristian  |
| 10h41 | Salvi              | Florian   |
| 10h49 | Lehner             | Ellen     |
| 10h57 | Mamin              | Lilou     |
| 11h05 | Schmutz            | Colin     |
| 11h13 | do Rego dos Santos | Kevin     |
| 11h21 | Bouriot            | Marc      |
| 11h29 | Pinto Vale         | Daniel    |
| 11h37 | Favre              | Joé       |

## Feedback

Les notes et remarques vous seront envoyées via Teams.

Pour toute question ou réclamation, contactez l'équipe enseignante. Chaque personne est encouragée à proposer des améliorations ou poser des questions sur le projet via Teams.