

Documentation de Podz

Application web de publication de podcasts

Podz

Projet TPI - 2022

Samuel Roland

Analyse préliminaire

Table des matières

- Analyse préliminaire
 - Introduction
 - Objectifs
 - Planification initiale
- Analyse / Conception
 - Concept
 - Technologies utilisées
 - Base de données: MCD
 - Base de données: MLD
 - Maquettes
 - Choix de conception
 - Stratégie de test
 - Risques techniques
 - Planification
 - Dossier de conception
 - Upload d'un fichier audio d'épisode
 - Composants réutilisables
- Réalisation
 - Dossier de réalisation
 - Description des tests effectués
 - Erreurs restantes
 - Liste des documents fournis
- Conclusions
- Annexes
 - Résumé du rapport du TPI / version succincte de la documentation
 - Sources – Bibliographie
 - Journal de travail
 - Manuel d'Installation
 - Manuel d'Utilisation
 - Archives du projet

Introduction

todo: import sections du canva.

Objectifs

Planification initiale

Analyse / Conception

Concept

Technologies utilisées

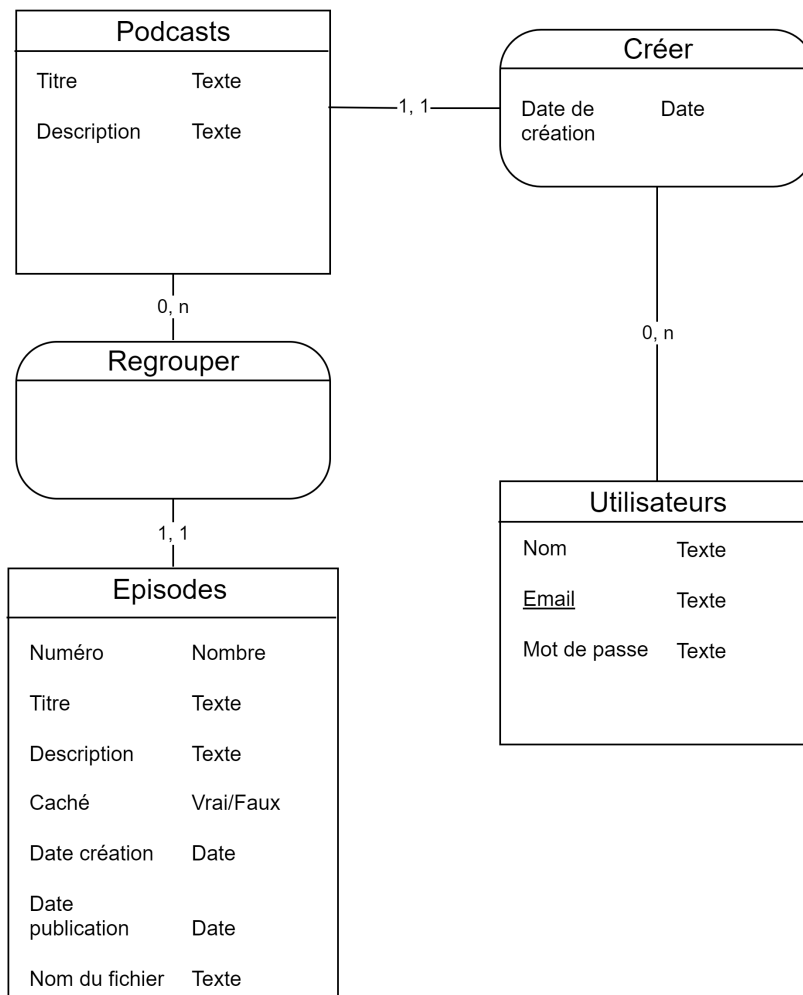
J'ai choisi la stack TALL (TailwindCSS - AlpineJS - Livewire - Laravel) pour ce projet, car je suis à l'aise avec ces 4 frameworks et qu'ils permettent d'être très productif pour développer une application web.

Petits aperçus de ce que sont ces frameworks:

- **Laravel**: un framework PHP basé sur le modèle MVC et en POO.
- **Livewire**: un framework pour Laravel permettant de faire des composants fullstack réactifs. L'idée est d'utiliser la puissance de Blade et du PHP pour gérer des interactions frontend (écrites normalement en Javascript).
- **AlpineJS**: un petit framework Javascript relativement simple à apprendre, utilisée ici pour gérer certaines interactions que Livewire ne permet pas, ou qui ne concernent des états d'affichage (là où des requêtes sur le backend serait inutile notamment). Les composants s'écrivent inline (sur les balises HTML directement).
- **TailwindCSS**: un framework CSS, semblable à Bootstrap mais centré autour des propriétés CSS (en ayant des classes utilitaires - "utility-first") au lieu de tourner autour de composants. C'est très puissant pour construire rapidement des interfaces, en écrivant quasiment jamais de CSS pur, et pour faire du responsive c'est très pratique.

Base de données: MCD

Schéma: MCD Podz - TPI
Auteur: Samuel Roland
Version: v2
Date de version: 09.05.2022



En dehors des champs évidents qui n'ont pas besoin d'explications, voici quelques aspects techniques demandant des explications.

Dans Episodes:

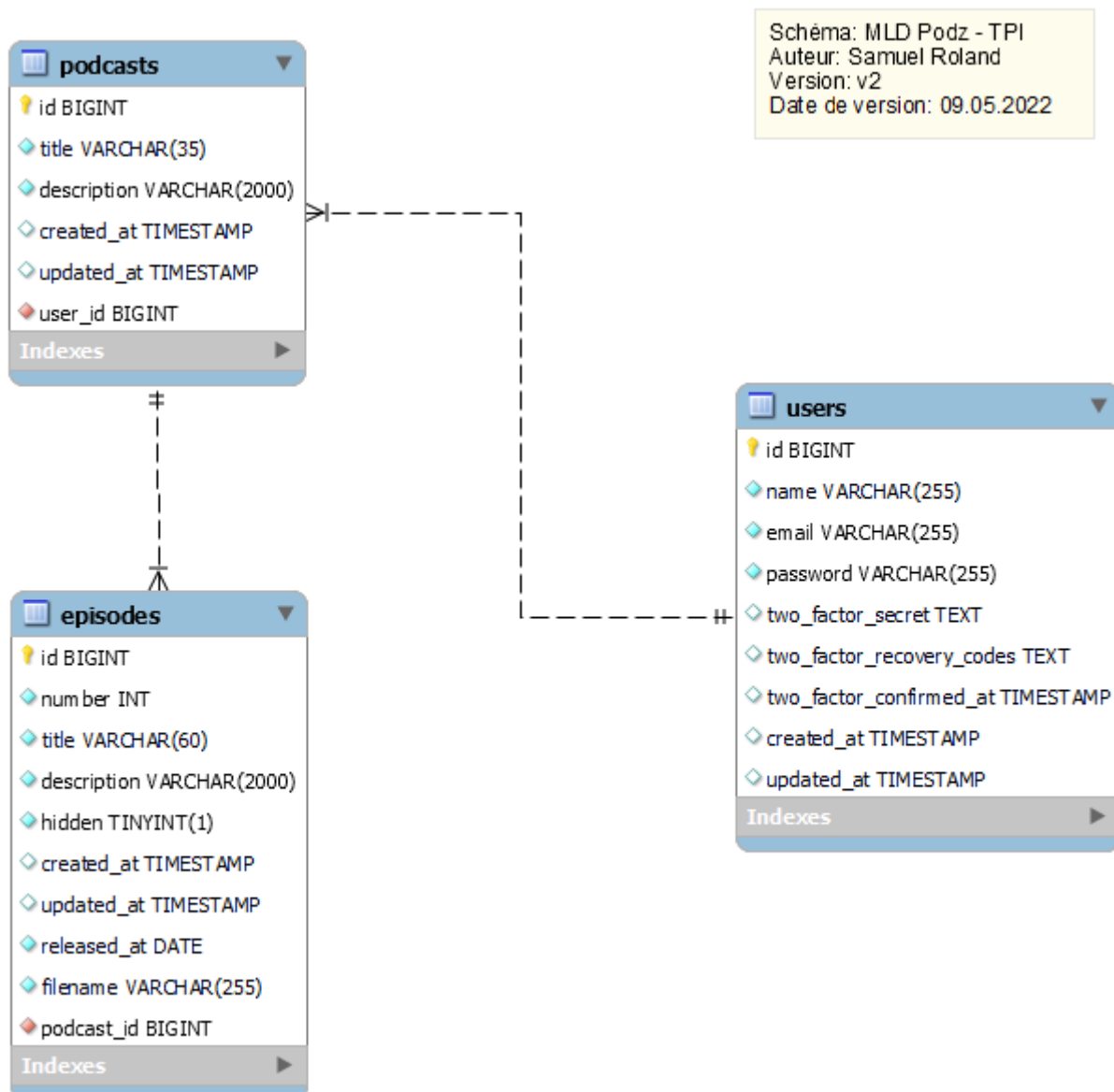
- Les combinaisons du Numéro et du podcast, ainsi que le titre et le podcast, sont uniques (exemple: on ne peut pas avoir 2 fois l'épisode 4 du podcast "Summer stories", et on ne peut pas avoir 2 fois un épisode nommé "Summer 2020 review" du podcast "Summer stories").
- La date de création est définie par la date de création de l'épisode sur la plateforme (avec l'upload du fichier), peu importe ses autres informations (la publication ou l'état caché n'a pas d'influence sur cette date). Cette date ne change jamais et ne sert/est affichée qu'à l'auteur.
- La date de publication peut être dans le passé ou mais dans le futur. Si elle est dans le futur, l'épisode n'est pas encore publié (jusqu'à la date définie). Ceci permet de programmer dans le futur une publication.

- Le champ Caché est par défaut à Faux et n'a pas d'effet dans ce cas. S'il est Vrai, l'épisode ne sera pas visible dans les détails du podcast.

Dans Podcasts:

- La combinaison du titre et de l'auteur est unique. Exemple: Michelle ne peut pas publier 2 podcasts s'appelant "My story", par contre Michelle et Bob peuvent chacun publier 1 podcast nommé "My story".

Base de données: MLD



todo: documenter spécificités. todo: tables et champs gérés par Laravel...

Maquettes

Pour pouvoir utiliser les fonctionnalités requises, voici la liste complète des pages existantes et leur maquette.

- Page Connexion
- Page Inscription

- Page Liste des podcasts
- Page Détails d'un podcast (visiteur)
- Page Edition des détails d'un podcast (auteur)
- Page Détails d'un podcast (auteur)
- Page Création d'un podcast

Page Connexion

Podz v1.0

Podcasts

[Connexion](#) [Inscription](#)

Podz v1.0

Nom d'utilisateur ou email

Mot de passe

Login

Page Inscription

Podz v1.0

Podcasts

Connexion [Inscription](#)

Podz v1.0

Username

Email

Password

Confirm Password

[Register](#)

Page Liste des podcasts

Cette page est visible publiquement et est la page par défaut de l'application, on y accède également via le bouton Podcasts en haut à gauche. On peut cliquer sur un podcast pour accéder à ses détails.

Podz v1.0

Podcasts

Connexion [Inscription](#)

Podcasts

Parcourez et découvrez tous les podcasts publiés sur Podz.

Histoire d'été en Italie - vacances 2021

par JohnDoe 3 épisodes

Chaque jour de mes vacances d'été raconté en 5 minutes.

Le monde des technos

par JohnDoe 12 épisodes

Des news, des exclusivités et beaucoup de fun dans cette ...

La chance

par Alice Y 19 épisodes

Question philosophique sur la chance et la malchance...

Humour et sarcasme

par Simon Curve 52 épisodes

La science du rire décortiquée.

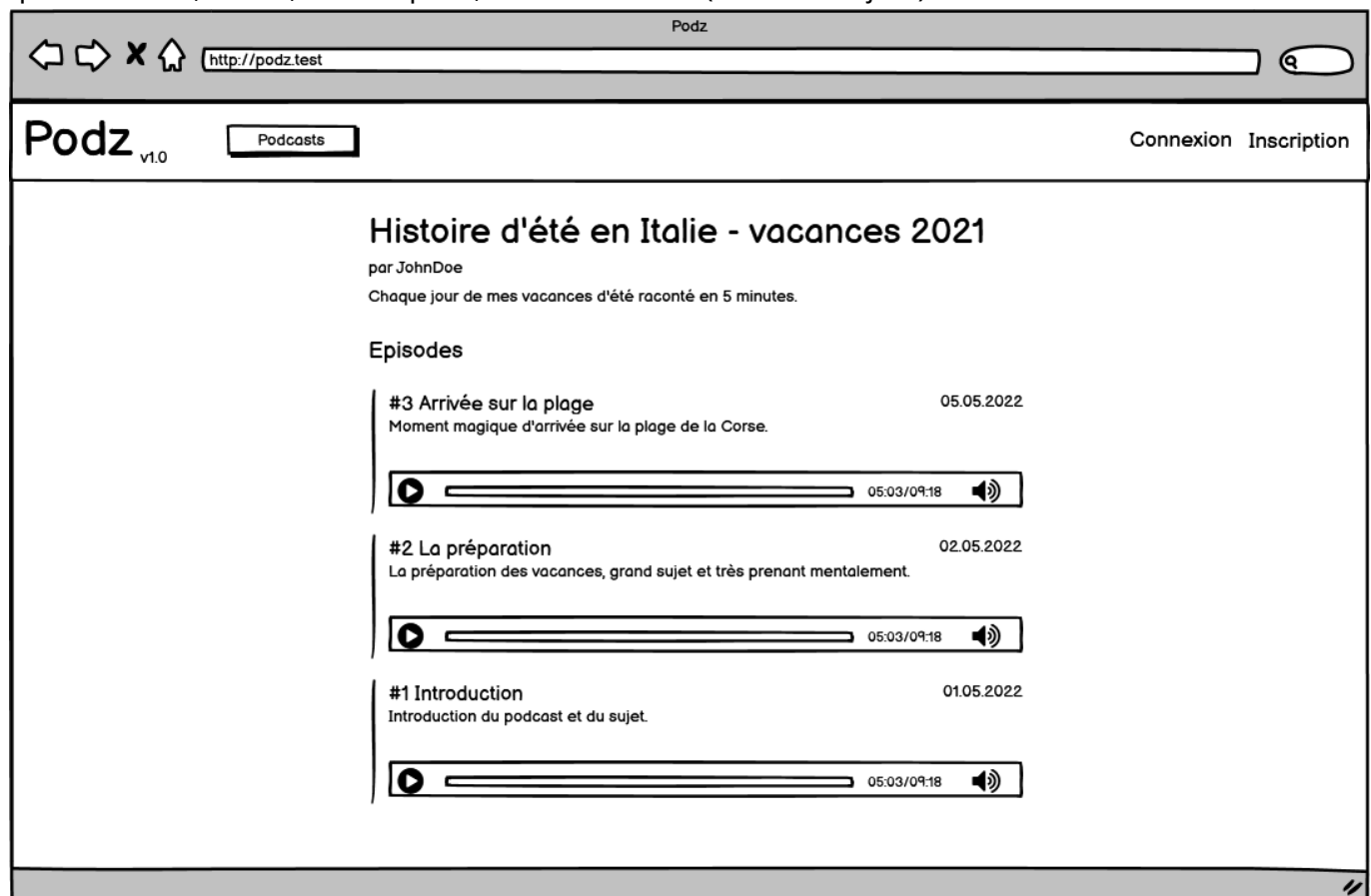
Entrepreneuriat 2.0

par Hanna Beans 3 épisodes

Comment changer son fonctionnement actuel ?

Page Détails d'un podcast (visiteur)

Les visiteurs ne voient que les épisodes qui sont visibles et qu'une partie de leurs informations. Ils ne voient que le numéro, le titre, la description, l'audio et le date (arrondie au jour).



Page Edition des détails d'un podcast (auteur)

L'auteur d'un podcast peut gérer les détails de son podcast, autant le titre et la description que les détails et la liste des épisodes. Nous sommes le 09.05.2022 dans cette maquette, l'épisode 4 est caché et le 5 est planifié pour le 10.05.2022. Ici l'auteur crée un 5 ème épisode planifiée qui ne sera publié que le lendemain à 15h08. Il peut aussi éditer les anciens épisodes en cliquant sur l'icône de stylo, ce qui passe l'épisode en

mode édition (et permet ainsi de modifier).

A Web Page

http://pretpi.test

Podz_{v1.0}

Podcasts

John Doe ▼

Histoire d'été en Italie - vacances 2021

par JohnDoe

Chaque jour de mes vacances d'été raconté en 5 minutes.

Enregistrer

Nouvel épisode

#5 ☐ Caché

Fichier audio (mp3, m4a)

Aperçu

05:03/09:18

Episodes

#4 ☒ Caché

Aperçu

05:03/09:18

#3 Arrivée sur la plage Créé le 05.05.2022 Publié le 08.05.2022 à 05:50

Moment magique d'arrivée sur la plage de la Corse.

05:03/09:18

Page Détails d'un podcast (auteur)

L'auteur voit évidemment toutes les informations de ses podcasts contrairement au visiteur. (Pour les podcasts d'autres auteurs, il voit la vue visiteur). Nous sommes le 10.05.2022 dans cette maquette, l'épisode 4 est caché et le 5 est planifié pour le 10.05.2022. L'épisode 4 est caché parce que l'auteur a décidé après

A Web Page

X

http://pretpl.test

Podz

v1.0

Podcasts

John Doe ▼

Histoire d'été en Italie - vacances 2021

par JohnDoe

Chaque jour de mes vacances d'été raconté en 5 minutes.

Episodes

#5 Super découverteCréé le 09.05.2022Publication planifiée pour le 10.05.2022 à 15:08L'hôtel a une piscine ! C'est trop bien.

05:03 / 09:18

#4 Rencontre secrèteCréé le 05.05.2022Publié le 08.05.2022 08:12épisode surprise

05:03 / 09:18

#3 Arrivée sur la plageCréé le 05.05.2022Publié le 08.05.2022 05:50Moment magique d'arrivée sur la plage de la Corse.

05:03 / 09:18

#2 La préparationCréé le 02.05.2022Publié le 02.05.2022 à 18:00La préparation des vacances, grand sujet et très prenant mentalement.

05:03 / 09:18

#1 IntroductionCréé le 01.05.2022Publié le 01.05.2022 à 18:30Introduction du podcast et du sujet.

05:03 / 09:18

Simple formulaire pour créer un nouveau podcast, avec affichage des erreurs en dessous des champs si

jamais les valeurs rentrées sont invalides.

A Web Page

http://pretpi.test

Podz v1.0

Podcasts

John Doe ▼

Créer un podcast

Titre

Description

Créer

Choix de conception

- Sur la page Podcasts, il y a un résumé des descriptions des podcasts, qui se limitent à 130 caractères (+3 petits points), puisque la description est trop longue pour être affichée entièrement et l'utilisation de `text-overflow: ellipsis` en CSS sur plusieurs lignes n'est pas très simple. Raccourcir en PHP était donc l'autre solution. Un attribut `summary` de la classe `Podcast` permet de récupérer ce résumé. Si la description est plus courte que 130 caractères, la description est utilisée.

Stratégie de test

Risques techniques

Planification

Dossier de conception

Upload d'un fichier audio d'épisode

J'ai décidé de fixer la taille maximum d'upload de fichiers à 150MB. Cette limite est fixée dans l'application, au niveau de la validation à la création d'un épisode. Les 2 valeurs dans la configuration de PHP (fichier

`php.ini`) doivent être augmentées au dessus de 150MB: `upload_max_filesize` et `post_max_size`.

Composants réutilisables

Le composant Field Un composant blade permettant d'abstraire les éléments communs de tous les champs de formulaire, avec quelques réglages possibles. L'affichage du label, le design basique, l'affichage des erreurs de validations.

Propriétés du composant

Nom	Type	Requis	Description
<code>name</code>	String	X	Le nom technique du champ, utilisé pour le <code>name</code> de l'input et par le <code>@error()</code>
<code>label</code>	String		Nom du label au dessus du champ
<code>type</code>	String		Type de l'<input>. Par défaut <code>text</code> . Si <code>textarea</code> est donné, une balise <textarea> est utilisée à la place.
<code>placeholder</code>	String		Un placeholder qui est ajouté directement sur le champ
<code>cssOnField</code>	String		Des classes CSS qui sont ajoutées directement sur le champ

Tous les autres attributs non reconnus sont transférés à la `div` racine du composant, ce qui permet d'ajouter du style ou d'autres attributs HTML. Tous les attributs commençant par `wire:model` sont ajoutés au champ pour permettre l'utilisation de ce composant avec Livewire.

Exemple d'utilisation:

```
<form action="{ route('podcasts.index') }}" method="POST">
< x-field label="Title" name="title"></x-field>
< x-field label="Description" type="textarea" name="description"></x-field>
< x-field label="Date de naissance" type="date" name="user.date"></x-field>
[... ]
</form>
```

Un autre exemple d'utilisation dans le cas d'un formulaire géré par Livewire:

```
<div>
  < x-field wire:keyup.enter="update" placeholder="Rentrez un titre court
et marquant." label="Title" name="podcast.title"
wire:model.lazy="podcast.title"></x-field>
  < x-field label="Description" type="textarea" name="podcast.description"
wire:model.lazy="podcast.description">
  </x-field>
  @csrf
  <button wire:click.prevent="update" class="btn mt-
1">Enregistrer</button>
</div>
```

Réalisation

Dossier de réalisation

Description des tests effectués

Erreurs restantes

Liste des documents fournis

Conclusions

Annexes

Résumé du rapport du TPI / version succincte de la documentation

Sources – Bibliographie

- Icônes: les icônes ont été copié-collées (en SVG) depuis heroicons.com, elle sont publiées sous licence MIT.
- [Liste des Types de médias, par l'IANA](#). Cette ressource m'a été utile pour trouver les types MIME des fichiers audios .ogg, .opus, .mp3 et .m4a.

Journal de travail

Manuel d'Installation

Manuel d'Utilisation

Archives du projet