

# Dossier Projet DAN

2024 - Titre de Concepteur développeur d'applications (CDA)

**Niveau 6**



**Projet D.A.N**

Droit Au Numérique

Ruben Trottein

# Sommaire

## Contenu

|      |  |    |
|------|--|----|
| I.   | Présentation.....                                      | 5  |
| A.   | English Summary : .....                                | 5  |
| B.   | Introduction .....                                     | 5  |
| II.  | Notre Philosophie .....                                | 6  |
| A.   | Un questionnaire sur les usages .....                  | 6  |
| III. | Cadre du Projet .....                                  | 8  |
| A.   | Les 3 axes D.A.N .....                                 | 8  |
| B.   | Qualification des outils .....                         | 9  |
| C.   | Type de solutions et objectifs des applications .....  | 10 |
| D.   | Cibles et portée : .....                               | 10 |
| IV.  | Méthodologie et organisation .....                     | 11 |
| A.   | Contenu spécifique du projet .....                     | 11 |
| B.   | Etapas et échéances.....                               | 11 |
| V.   | Outils d'organisation et de planification.....         | 12 |
| A.   | Kanban notion pour le suivi du développement.....      | 12 |
| B.   | Détails .....  | 13 |
| C.   | Versioning et DevOps .....                             | 13 |
| D.   | Gantt Notion pour les corolaires du projet D.A.N ..... | 13 |
| VI.  | Conception UI/UX .....                                 | 14 |
| A.   | Zoning.....  | 14 |
| 1.   | Zoning (landing page).....                             | 14 |
| 2.   | Formulaires .....                                      | 16 |
| 3.   | Articles .....   | 17 |
| B.   | Wireframes .....                                       | 19 |
| 1.   | Landing.....   | 19 |
| 2.   | Article.....   | 21 |
| C.   | Maquettes.....   | 25 |
| 1.   | Landing.....   | 25 |
| 2.   | Articles .....   | 25 |
| 3.   | Formulaires .....                                      | 26 |
| VII. | Charte Graphique.....                                  | 27 |
| A.   | Palette de couleurs : .....                            | 27 |
| B.   | Polices : .....  | 27 |

|        |   |    |
|--------|---|----|
| VIII.  | Conception de la base de données .....                | 28 |
| A.     | DuckDrive API.....                                    | 28 |
| B.     | Hedgehog School.....                                  | 28 |
| IX.    | Conception de back-end .....                          | 29 |
| A.     | Use Case Duck Drive (API).....                        | 29 |
| B.     | Use Case Hedgehog School .....                        | 29 |
| X.     | Diagrammes de séquence.....                           | 30 |
| XI.    | Conception multi-couches : .....                      | 31 |
| A.     | MVC.....  | 31 |
| B.     | Architecture à 3 niveaux .....                        | 33 |
| XII.   | Sécurité .....  | 34 |
| A.     | Attaque XSS.....                                      | 34 |
| B.     | CSRF.....   | 34 |
| C.     | Injection SQL .....                                   | 34 |
| D.     | Attaque brute force .....                             | 34 |
| E.     | Politique de mot de passe (CNIL ou ANSSI), RGPD ..... | 34 |
| F.     | Hashage des mots de passe .....                       | 34 |
| XIII.  | Politique de test.....                                | 35 |
| A.     | Test Unitaire.....                                    | 35 |
| B.     | Test d'Intégration.....                               | 36 |
| C.     | Test Fonctionnel.....                                 | 37 |
| D.     | Test de non-régression .....                          | 39 |
| E.     | Test de montée de charge .....                        | 39 |
| XIV.   | Déploiement et mise en production.....                | 40 |
| A.     | API : .....   | 40 |
| B.     | Symfony : .....                                       | 40 |
| XV.    | Veille technologique .....                            | 41 |
| A.     | Veille globale.....                                   | 41 |
| B.     | Veille de sécurité.....                               | 41 |
| XVI.   | Difficultés rencontrées.....                          | 41 |
| XVII.  | Post-projet .....                                     | 42 |
| XVIII. | CGU et RGPD.....                                      | 43 |
| XIX.   | Documents annexes et liens : .....                    | 43 |
| A.     | Lien de la présentation : .....                       | 43 |
| B.     | Questionnaire d'accueil : .....                       | 43 |
| XX.    | Contact.....  | 43 |
| A.     | Adresse de l'entreprise .....                         | 43 |
|        | 54 rue des Acacias 75000 Paris .....                  | 43 |

|       |                 |    |
|-------|-----------------|----|
| B.    | Email.....      | 43 |
| C.    | Téléphone ..... | 43 |
| XXI.  | Crédits .....   | 43 |
| A.    | Images.....     | 43 |
| B.    | Vidéos.....     | 43 |
| C.    | Autres.....     | 43 |
| XXII. | Annexes.....    | 44 |

# I. Présentation

## A. English Summary :

### About Me

Welcome to this presentation, I'm Ruben Trottein

As a holder of Business Organization field degree, I have a significant experience in data migration. In addition, I have an English language technical proficiency as I have worked with international companies in the domain. After working in document digitalization for a year, I specialized in software environment, as I have always been a kind of computer aficionado. I trained in 2018 to become web Developer, then I underwent a training to be Java development engineer, however, I ended up as a web dev. freelance trainer.

I now am graduating from this Application Conception and Development training, my final objective is to work in the associative field around the concept of digital illiteracy and that's what this project is all about.

### About the project

Nowadays, there is a strong dependence around electronic devices and Internet. Everyone uses their smartphones. However, professionally speaking, computers are kings. From this observation, we came to emphasize the differences between rejecting, knowing and mastering numeric tools, and ultimately focusing around three profiles and solutions in order to bring them to the level of comfort.

The project is called D.A.N., which stands for 'Droit Au Numérique' (roughly translatable as "Digital Rights," analogous to "Human Rights"). The philosophy is to give everyone the weapons to be at ease in a world where internet and computer science are more and more important. And the answers we provide are tutorials, file storage and file visualization.

Our main subject here will be the tutorial site, named Hedgehog school. Its goal is to help reluctant people to learn procedures and use of everyday life websites and software so they can refer to it instead of actually learning to master the tools. Of course, every type of tutorial is accepted, so it can be simple or complex.

In a technical point of view, the website will consist of a node.js API, written in RESTful JavaScript, serving tutorials media and content while a Symfony (PHP) app with an SQL database manages users, templating and user experience.

## B. Introduction

*L'importance de la souveraineté des données pour l'utilisateur ainsi que la liberté fondamentale que constitue l'accès au numérique a été notre leitmotiv dans la conceptualisation de l'initiative D.A.N*

Droit Au Numérique A été créée en 2023, et est une initiative à but non-lucratif ayant pour objectif de rééquilibrer l'accès aux ressources et outils multimédia.

Notre motivation est de démocratiser l'utilisation des outils numériques par la formation et la mise a disposition d'applications gratuites, personnelles, libres et sécurisées.

## II. Notre Philosophie



“Celui qui déplace la montagne, c'est celui qui commence à enlever les petites pierres.” -  
*Confucius*<sup>1</sup>

Trop souvent l'énormité des tâches qui nous incombe finit par nous écraser. Notre philosophie consiste à guider nos adhérents et leur offrir des outils simples leur permettant de maîtriser leurs données et fichiers, les garder dans un environnement qui leur appartient et à survivre et évoluer au sein d'un siècle de plus en plus inscrit dans la révolution numérique.

Contrairement à la plupart des acteurs de la Tech, nous faisons le choix de travailler dans un environnement associatif et étatique afin de se soustraire au maximum aux logiques de rentabilité. Nous espérons proposer des services viables et utiles qui resteront pérennes dans le temps.

### A. Un questionnaire sur les usages

Nous tentons d'aider les utilisateurs à choisir leur prestations et leurs interlocuteurs avec un petit quiz simple qui aide à déterminer leur profil selon un système simple à 3 modèles:

- **Hérisson**, qui manque de souplesse et d'aise avec les usages de l'informatique qu'on va chercher à aider, former et équiper avec patience et pédagogie.
- **Paresseux**, qui a juste besoin de passer d'un usage loisir à un usage professionnel et administratif des outils, ou tout simplement passer de l'usage du smartphone à celui de l'ordinateur, qu'on va chercher à consolider et à motiver.
- **Canard**, qui peut tout faire mais aura besoin ou envie de participer à des ateliers, aider l'association ou des proches ou passer de la pratique éclairée à une activité professionnelle dans le secteur de la tech. Notre objectif sera de le cadrer et de le professionnaliser tout en le soutenant

Pour ce faire nous utiliserons un outil de prétri et d'évaluation orienté surtout sur les débutants pour aider ces derniers à mettre un mot sur leurs difficultés et leurs besoins pratiques tandis que les utilisateurs d'outils numériques souhaitant se professionnaliser seront orientés plutôt sur de l'accompagnement ou de la formation

## Prototype du questionnaire

Usage de l'informatique et besoins  
numériques



### III. Cadre du Projet

*A la genèse de l'élaboration du projet D.A.N, nous avons pu constater un certain nombre de manquements auxquels nous avons apporté les solutions suivantes*

1. Beaucoup de personnes sont éloignées du monde numérique aujourd'hui et rencontrent des difficultés graduelles à faire des tâches auparavant à leur portée (paiement des impôts, achats, utilisation d'outils...).
2. Les profils réfractaires à la technologie devraient pouvoir évoluer dans la vie quotidienne sans subir un handicap dû à leur inaptitude ou refus des produits et concept et surtout comprendre ce qu'on leur demande et ce qui leur arrive.
3. De par la difficulté de compréhension et du volume de choses à apprendre, des détails importants tels que la sécurité sont complètement négligés.
4. Malgré un usage quotidien et maîtrisé des technologies, beaucoup de profils ne savent pas utiliser les ordinateurs.
5. Des utilisateurs de Smartphones ne savent pas protéger ou retrouver leurs données personnelles.
6. Des utilisateurs de Smartphones sont parfois frustrés par l'exiguïté des applications mobiles qui les limite dans leur utilisation et leurs possibilités.
7. Les utilisateurs avancés de la Tech sont souvent isolés et se démotivent par manque d'option au bout d'un certain niveau de compétence
8. Une fois conscient des problèmes du big data et de la protection des données personnelles, le manque de solutions libres peut empêcher de s'en préoccuper activement
9. Beaucoup d'utilisateurs Tech sont également pieds et poings liés par des technologies propriétaires hégémoniques telles que Google, Azure ou AWS

#### A. Les 3 axes D.A.N

*Les trois axes D.A.N sont des propositions de points par lesquels aborder le problème de la fracture numérique*

##### 1. Lutte contre l'illélectronisme

Chaque citoyen doit avoir accès facilement à toutes les ressources du quotidien qui nécessitent l'outil informatique, ou, à défaut, d'une formation sur les usages numériques suffisante pour y accéder de son propre chef.

L'illélectronisme, théorisé aux alentours de 2002<sup>i</sup> est au fil des usages de plus en plus proche d'une situation de handicap, et les personnes âgées comme les plus jeunes le subissent tout autant, mais de façon très différente :

- les premiers ont grandi dans un monde sans numériques et n'ont pas pris le train
- les seconds ont grandi un Smartphone à la main et n'ont finalement jamais connu l'informatique traditionnelle, essentielle dans une vie d'adulte et dans les milieux professionnels



## 2. Souveraineté sur les fichiers

Nous estimons que les fichiers personnels, devenus essentiels à chacun devraient rentrer dans la propriété inaliénable d'un être humain au titre de ses droits fondamentaux.

Chaque Citoyen devrait donc pouvoir avoir accès à des applications auto-hébergées pour gérer ses propres fichiers gratuitement et sans les confier à un tiers.

L'accès et l'information sur ce genre d'application et de pratiques doit aussi être généralisé afin que les utilisateurs aient réellement le choix de leurs outils ainsi que du devenir de leurs données.

## 3. Quiétude face aux ingérences

Chaque citoyen doit être protégé des ingérences commerciales, publicitaires ou étatiques dans son usage d'internet. La liberté passe par la possibilité de faire les choses essentielles véritablement gratuitement et non toujours en échange de données ou de motivation possiblement intrusive.

# B. Qualification des outils

## 1. Projet utile et profitable

On sait que le projet peut être utilisé par une grande frange de la population, et que les outils individualisés peuvent proposer une solution pour une grande variété de profils.

## 2. Besoin pour les particuliers et associations

Le caractère personnalisable des outils permet leur utilisation dans les petites structures limitées en personnel et en organisation, proposant une sécurité et une souveraineté accrue aux responsables a un coût maîtrisé.

## 3. Technologies modernes

Nos outils sont conformes aux bonnes pratiques et utilisent des bibliothèques, API et langages de programmation modernes et amenés à évoluer dans le temps.

## 4. Business management moderne

Notre business model, s'il n'a pas pour objectif la rentabilité, permet la pérennité des solutions dans le temps et s'inscrit dans un cycle d'intégration continue qui garantit aux logiciels de rester à jour ou au moins d'être capables de mise à jour autonomes complètes. Le modèle coopératif, ayant fait ses preuves, sera utilisé en interne pour conserver un fonctionnement tourné vers l'objectif plutôt que vers la rentabilité

## 5. Proposition sécurisée

La volonté de Cloud personnel et la politique stricte de souveraineté des données garantissent des solutions sans intermédiaire capables de fonctionner en réseau privé, VPN, intranet ou n'importe quel autre circuit fermé, ce qui limite la possibilité d'intrusion ou d'atteinte à l'intégrité des données et projets existants.

## 6. Objectifs et indicateurs orientés métier

Les applications D.A.N n'ont pas d'autre objectif chiffré que rentrer dans leur frais de fonctionnement. Open Source, utilisable offline et décentralisée, la solution définitive devrait avoir pour seul objectif l'accessibilité et la pédagogie

## C. Type de solutions et objectifs des applications

### 1. Application mobile de visualisation

Pour diffuser les contenus vidéo et audio sans les mettre forcément à disposition rapidement et facilement (interrogation de l'API DuckDrive). Possibilité de partager également.

### 2. Portail pédagogique

Pour développer et organiser des contenus de formation, des tutoriaux et conseils mais aussi des documentations légères, cheatSheets ou wikis visant à échanger des informations et aider des utilisateurs, le tout irrigué par les contenus DuckDrive

### 3. FTP en ligne avec page d'accueil et Dashboard

Pour organiser, sauvegarder en ligne et partager en visualisation des fichiers personnels et alimenter les deux autres applications. L'API a pour vocation d'envoyer les fichiers dans une solution au choix (petit hébergeur mutualisé) ou, idéalement dans un espace de stockage personnalisé choisi par l'utilisateur (son ordinateur, son hébergeur ou serveur perso, ou sa solution cloud).

## D. Cibles et portée :

### 1. Cibles à atteindre :

- Particuliers soucieux de leur souveraineté sur leurs fichiers et données
- Associations n'ayant pas d'intérêt ou de moyens pour partager, transmettre et sécuriser leurs données en ligne
- Associations ou PME n'ayant pas de budget Cloud ou d'outil convenable
- Particuliers ou entreprises ayant besoin d'un Cloud rapide et/ou limité
- Particuliers, associations, entreprises souhaitant obtenir des informations
- Individus souhaitant partager une information sans intermédiaires
- Développeurs souhaitant héberger facilement une partie de leurs données ou fonctionner en circuit fermé
- Débutants souhaitant s'améliorer dans un domaine ou un autre
- Personnes souhaitant partager de la connaissance ou des techniques

### 2. Portée du site (locale, nationale, internationale)

- Portée locale (Logiciel en français)
- Extensible puisque le logiciel sera libre et gratuit [traduction envisageable]
- Techniquement rien n'empêche que le contenu d'un article ou de médias soit international.
- Le repo de l'application restant disponible et public il peut être utilisé n'importe où.

## IV. Méthodologie et organisation

### A. Contenu spécifique du projet

#### 1. DuckDrive (API)

Base :

- Stockage des fichiers plus lourds (multimédia, contenu texte lourd)
- Mise à disposition des fichiers avec une API RESTful

A terme :

- Icones, aperçus
- 2 pages (Dashboard avec explorateur et formulaire d'upload + page d'accès rapide aux favoris)
- Volet de visualisation rapide

#### 2. Hedgehog School

- Page de cours (vidéo + contenu markdown / Wysiwyg)
- Index des cours et recherche
- Dashboard cours proposés / suivis / favoris /
- Système de likes ou notes pour indexer les contenus qualitativement
- Module de connexion et de profil pour décrire le domaine de ses cours et l'expertise de leur(s) auteur(s)

A terme :

- Système de réputation pour créer un contrôle critique sur le contenu

#### 3. Sloth Tube (n'a pas vocation à être terminée pour le moment)

- Page de base streaming
- Favoris / Liste d'attente / Lecture Auto

#### 4. Pages Communes

- 404 !
- Page CGU / RGPD
- Page de présentation du projet D.A.N

### B. Etapes et échéances

*Fin Mai 2024*

- Conceptualisation => Spécifications techniques et Cahier des charges

*Fin Juin 2024*

- Maquettage => Proposition de wireframes et Proof of Concept

*Fin Décembre 2024*

- Prototypage => Mise en place du code du projet en environnement dev en déploiement continu

*Janvier 2025*

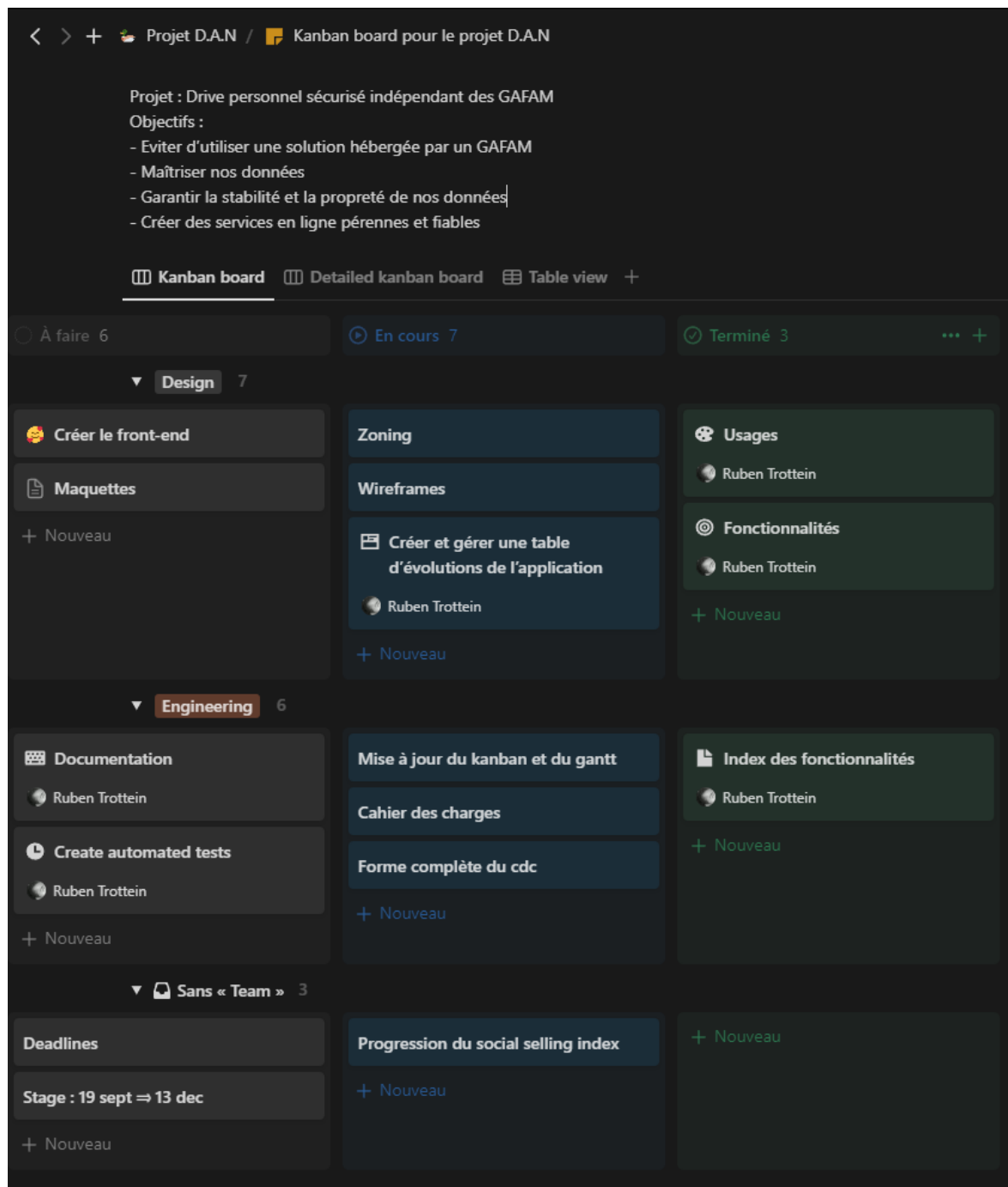
- Déploiement => Mise en Production des solutions

# V.Outils d'organisation et de planification

## A. Kanban notion pour le suivi du développement

Nous avons choisi le Kanban pour suivre le développement en raison de sa polyvalence et de sa praticité. Le Kanban D.A.N permet d'éviter de lancer trop de tâches à la fois et de suivre les prochaines tâches importantes.

Le back log contient peu ou prou le contenu défini dans ce dossier, et les tâches sont précisées le cas échéant à partir d'User stories autour des objectifs de chaque fonctionnalité.



## B. Détails

La mise en place du projet passera par ces outils, répartis par étapes :

- Spécification : Définir les besoins et fonctionnalités utiles (Notion, StarUML)
- Maquettage : Création d'un wireframe et d'une POC (Balsamiq / Figma / draw.io)
- Prototypage: Création d'un modèle de BDD, choix des technologies (Jmerise)
- Déploiement : Mise en Service en environnement test (Symfony / node.JS)

## C. Versioning et DevOps

Le GitHub du projet contient trois dépôts.

- Un dépôt général qui contient l'ensemble de la doc et les proof of concept dans un dossier a part
- Un repo d'intégration pour l'application Symfony Hedgehog School
- Un repo d'intégration pour l'API node.js DuckDrive API

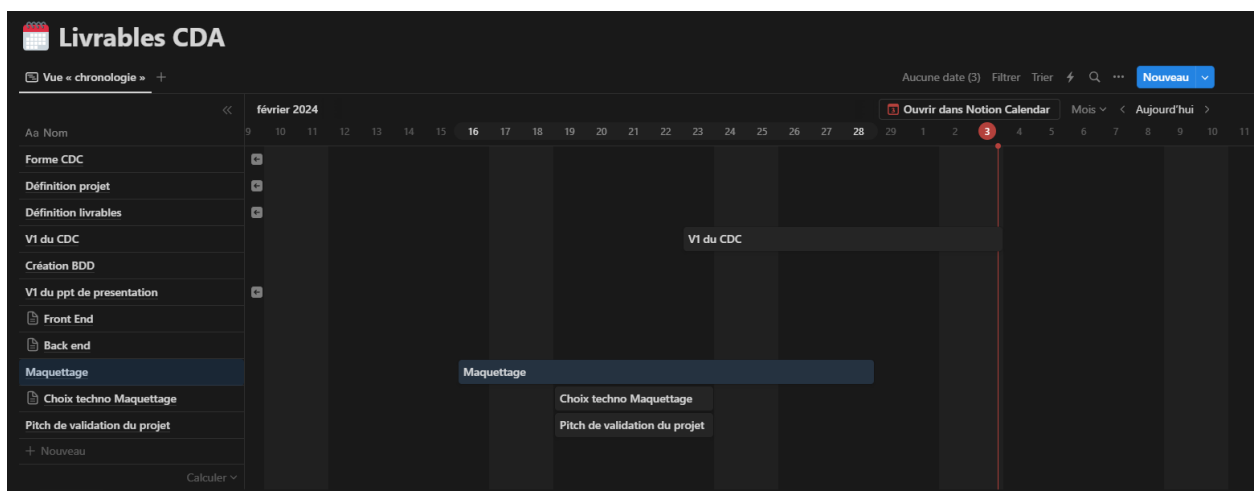
Le CI/CD sera géré via **Github Actions** à partir des deux derniers repo jusqu'à la version de production de l'application qu'on déploiera séparément sur un OVH après chaque mise à jour majeure grâce à l'intégration Git disponible sur l'hébergeur.

**Vercel** déploie l'API durant les phases de test.

A terme, nous pensons à un déploiement **Docker**.

## D. Gantt Notion pour les corolaires du projet D.A.N

Autour du projet, nous avons organisé les sous projets (documentation, modélisation, etc. ) via un Gantt Notion



Ici nous pouvons suivre l'avancée des autres grandes étapes du projet afin de les terminer en temps et en heure et surtout de lancer les sprints a des moments précis, même si la criticité n'est pas respectée (le livrable n'est pas aussi critique que l'application).

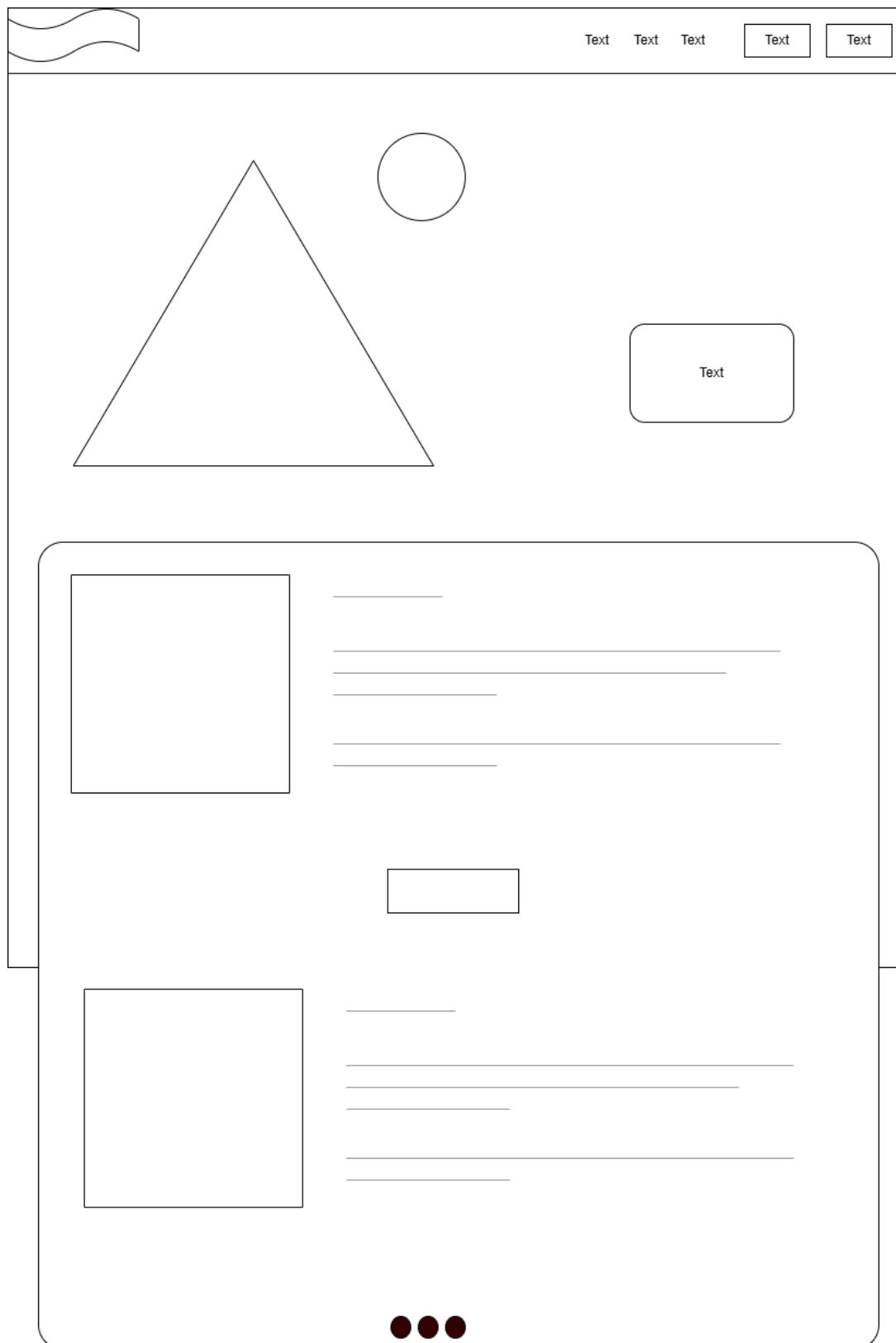
Une fois les impératifs de temps gérés, nous passons en méthode Kanban pour la phase de développement, gérée par priorités.

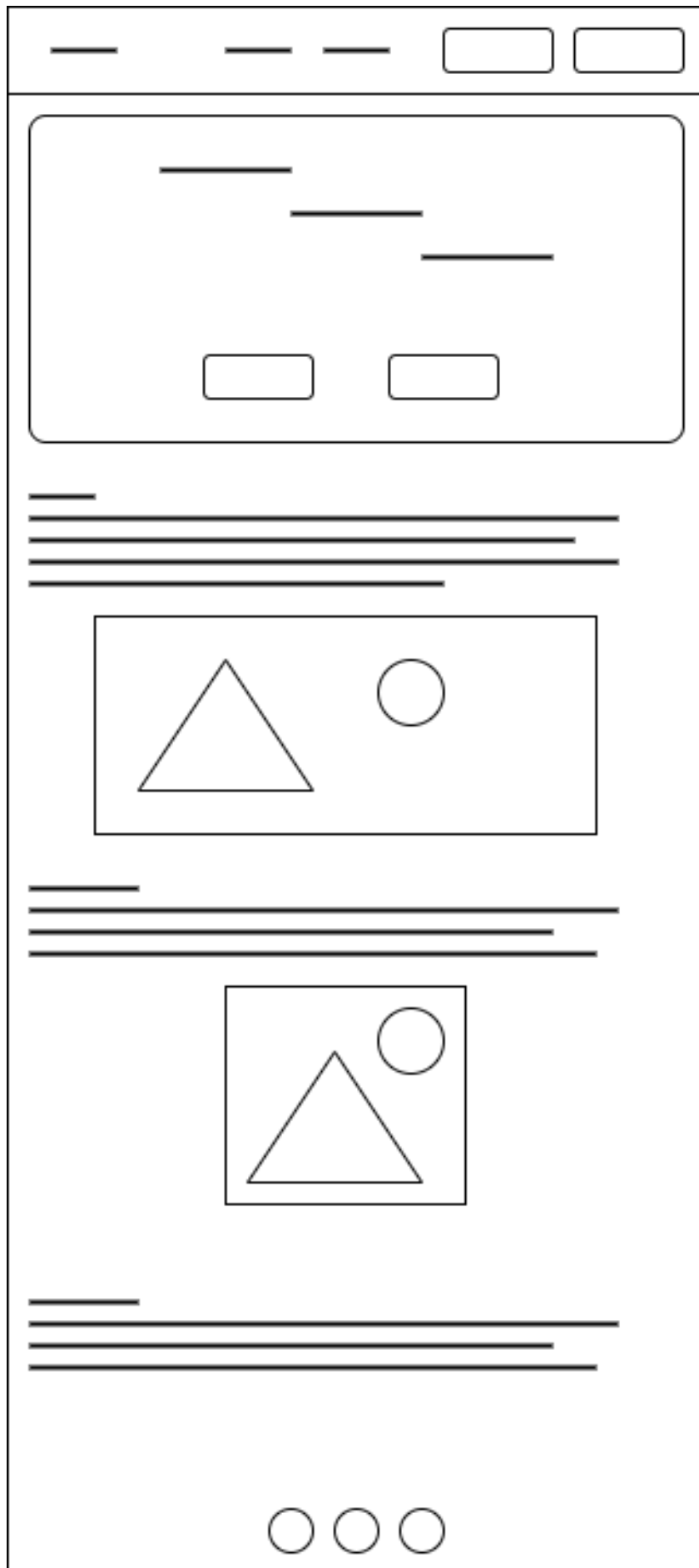
# VI. Conception UI/UX

## A. Zoning

### 1. Zoning (landing page)

Page d'accueil avec texte et Call to action





## 2. Formulaires

A vertical sidebar containing the following elements from top to bottom:

- A short horizontal line.
- A text input field.
- A large text area.
- A short text input field.
- A short text input field.
- A short text input field.
- A large text area.
- A small button.

Formulaires d'inscription connexion et edition d'articles  
(input text, number et textarea)

A main content area containing the following elements:

- A text input field.
- A text input field.
- A large text area.
- A button.

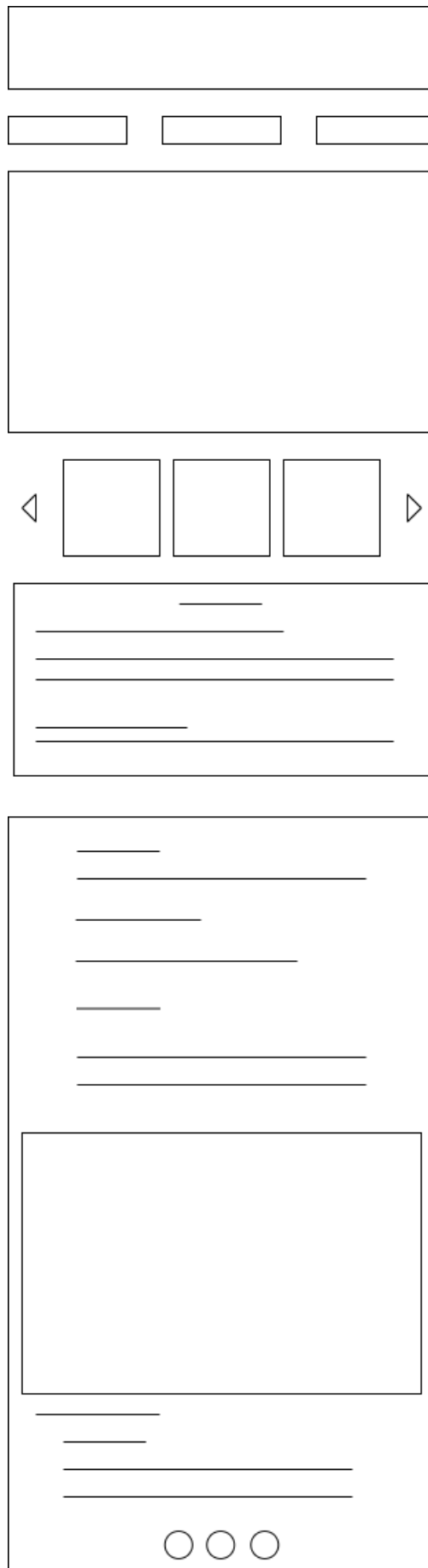


### 3. Articles

Page de consultation d'un article, onglets dynamique, récapitulative des medias et vidéo principale + contenu intro et contenu du chapitre

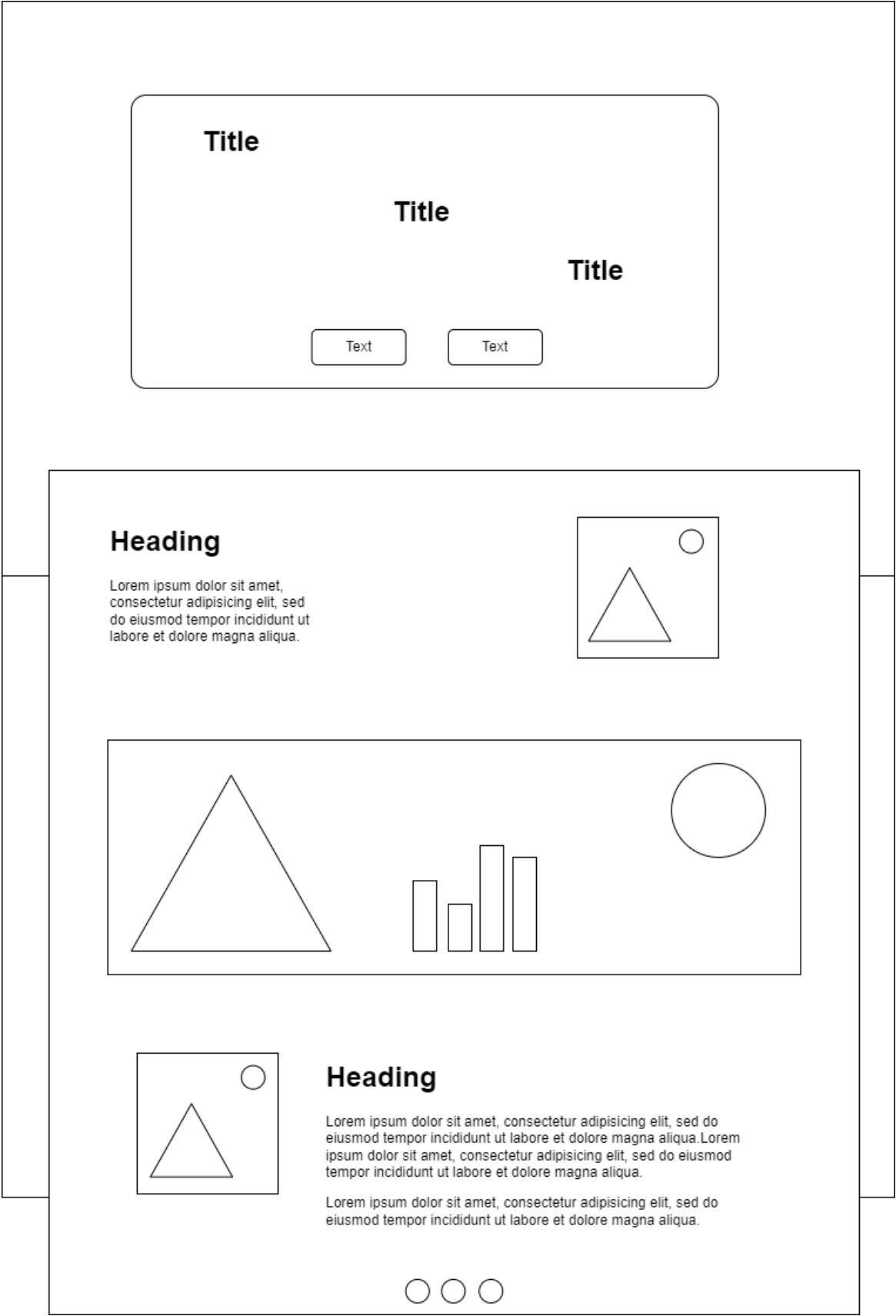


Lecture d'article sur mobile. Boutons de chapitre fixes.



1. Landing

Landing (background vidéo, avec connexion en popup)



**Title**

Text Text Text

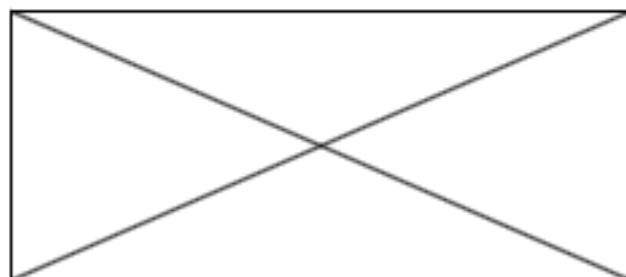
Text

Text



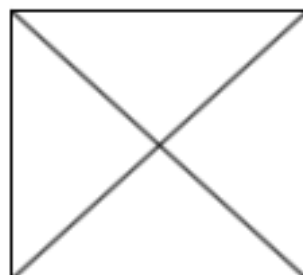
## Heading

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.



## Heading

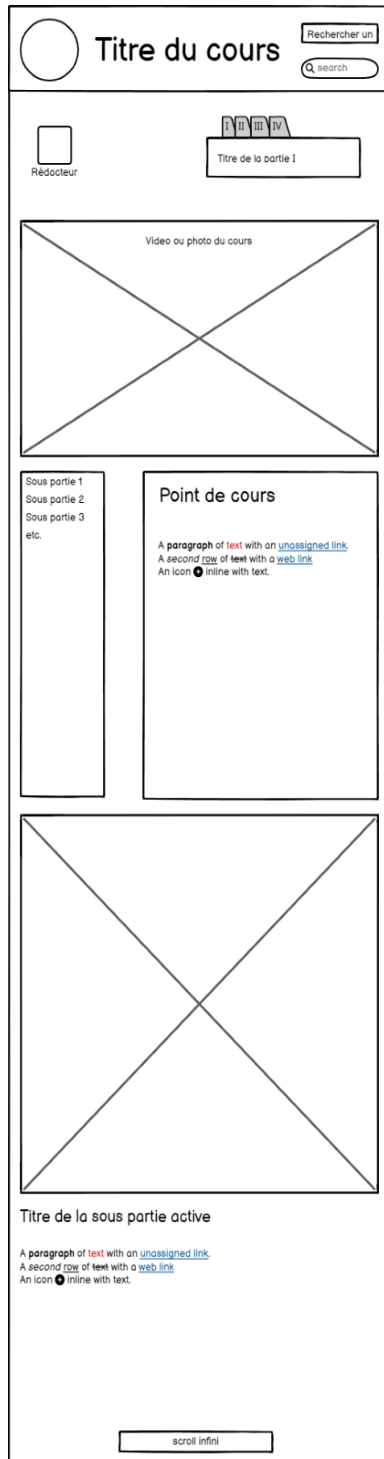
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.



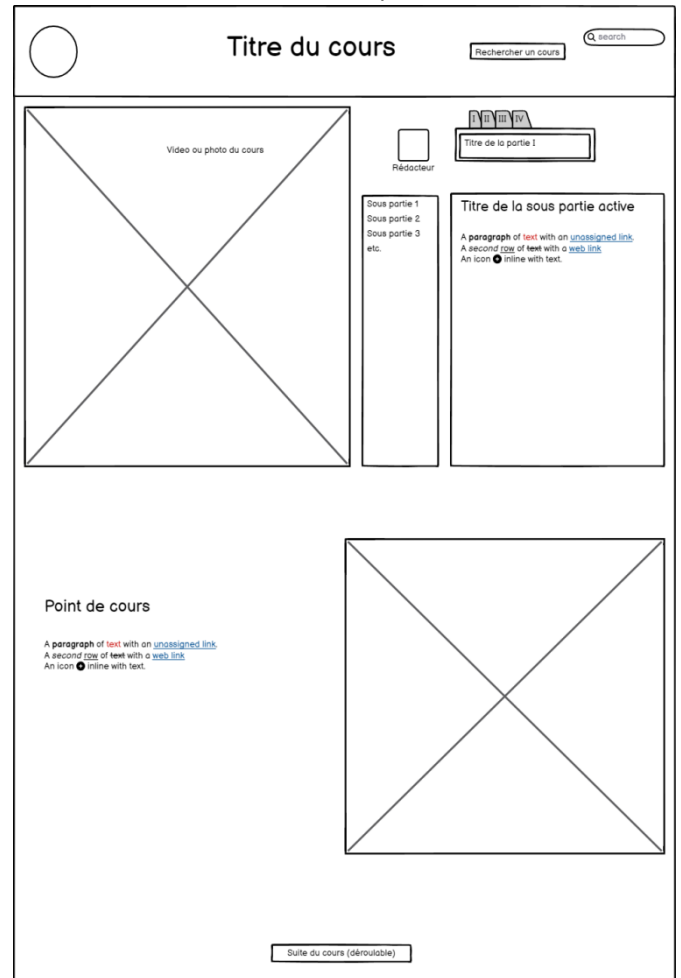
## 2. Article

Page de cours (desktop / mobile)

Mobile



Desktop



# Title

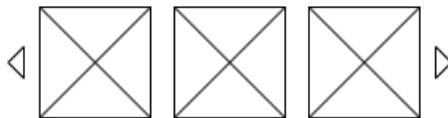
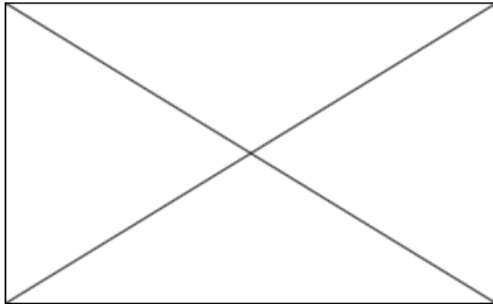
Text

Text

Text

Text

Text



Text

Text

Text

## Heading

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

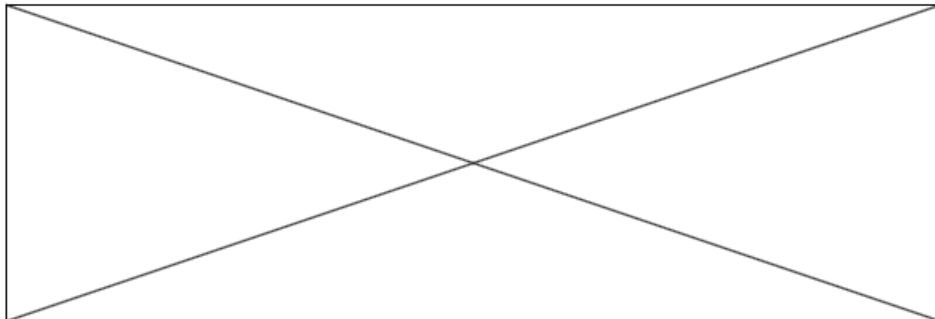
## Heading

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

## Heading

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.





## Titre de l'article

Tags: Tag2 Tag3 Tag4

Chapitre 1

Chapitre 2

Chapitre 3

## Chapitre 1

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

di consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla pariatur?

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui do

### Notes et commentaires

#### Heading




Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

#### Heading



Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

## 3. Formulaire

**Titre** [Accueil](#) [Articles](#) [Déconnexion](#)

## Formulaire

**Text**

**Text**


**Password**

**Textarea**

☐ **Checkbox**

**Submit**



 Titre

AccueilArticlesProfil

connexioninscription

Formulaire

Text

Text

Password

Textarea

Checkbox

Submit

## C. Maquettes

1. Landing

2. Articles

Page de visualisation des cours tutoriels ou articles

Page 1

https://www.draw.io

Titre du cours / tuto

HomeArticlesProfile

Chap 1Chap 2Chap 3

Titre de chapitre ou de l'intro

Wed Jun 26 2024 13:30:41

adipiscing elit, iste repellat dolore perferendis numquam

lorem

adipiscing elit, iste repellat dolore perferendis numquam

0:00/3:53

Titre

Variable Text

Titre video 1

Titre video 2

Titre video 3

Titre video 4

Titre video 4

Titre du Chapitre

Section 1.10.32 du "De Finibus Bonorum et Malorum" de Cicéron (45 av. J.-C.)

"Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro

Traduction de H. Rackham (1914)

"But I must explain to you how all this mistaken idea of denouncing pleasure and praising pain was born and I will give you a complete account of the system, and expound the actual teachings of the great explorer of the truth, the master-builder of human happiness. No one rejects, dislikes, or avoids pleasure itself, because it is pleasure, but because those who do not know how to pursue pleasure rationally encounter consequences that are extremely painful. Nor again is there anyone who loves or pursues or desires to obtain pain of itself, because it is pain, but because occasionally circumstances occur in which toil and pain can procure him some great pleasure. To take a trivial example, which of us ever undertakes laborious physical exercise, except to obtain some advantage from it? But who has any right to find fault with a man who chooses to enjoy a pleasure that has no annoying consequences, or one who avoids a pain that produces no resultant pleasure?"

Section 1.10.33 du "De Finibus Bonorum et Malorum" de Cicéron (45 av. J.-C.)

"At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum hic tenetur a sapiente delectus, ut aut reiciendis voluptatibus maiores alias consequatur aut perferendis doloribus asperiores repellat."

Traduction de H. Rackham (1914)

"On the other hand, we denounce with righteous indignation and dislike men who are so beguiled and demoralized by the charms of pleasure of the moment, so blinded by desire, that they cannot foresee the pain and trouble that are bound to ensue; and equal blame belongs to those who fail in their duty through weakness of will, which is the same as excuse through resignation from the spirit and aim. There is no excuse for this conduct, and

Chapitre terminé Remonter

Titre

because it is pain, but because occasionally circumstances occur in which toil and pain can procure him some great pleasure. To take a trivial example, which of us ever undertakes laborious physical exercise, except to obtain some advantage from

Wed Jun 26 2024 13:30:41

★ ★ ★ ★ ★

Titre

because it is pain, but because occasionally circumstances occur in which toil and pain can procure him some great pleasure. To take a trivial example, which of us ever undertakes laborious physical exercise, except to obtain some advantage from it? But who has any right to find fa

Wed Jun 26 2024 13:30:41

★ ★ ★ ★ ★

© P.A.N Project - 2024

3. Formulaires

26

## VII. Charte Graphique

Thème général a base de couleurs sobres (Noirs, verts, ocres et or). La couleur « Rebecca Purple » pour sa signification et pour mettre en lumière les call to action avec une couleur Gold pour les liens important

### A. Palette de couleurs :

|                                  |
|----------------------------------|
| RebeccaPurple: #1a7b69           |
| LIENS : Gold: #ffd700            |
| BUTTONS : RebeccaPurple: #663399 |

#### Alertes

|                                 |
|---------------------------------|
| INFO : Dodger Blue : #1e90ff    |
| SUCCESS : Pine Green : #1a7b69  |
| DANGER : Persian Red : # BB4430 |
| WARNING : Vanilla: # f3dfa2     |

#### Noirs et blancs

|                       |
|-----------------------|
| Steel Gray: # 948b8b  |
| Raisin Black: #231F20 |
| Linen: # EFE6DD       |
| White smoke : #f9f9f9 |

### B. Polices :

Dafont :

#### Deuxieme Rang

Voix ambiguë d'un cœur qui au zéphyr préfère les jattes de kiwis. 1234567890

#### Secret Code

Voix ambiguë d'un cœur qui au zéphyr préfère les jattes de kiwis. 1234567890

#### Vogue

VOIX AMBIGUë D'UN Cœur QUI AU Zéphyr PRéFère LES JATTES DE KIWis. 1234567890

Google Font :

#### Montserrat

Voix ambiguë d'un cœur qui au zéphyr préfère les jattes de kiwis. 1234567890

#### Edu Australia VIC WA NT Hand Precursive

Voix ambiguë d'un cœur qui au zéphyr préfère les jattes de kiwis. 1234567890

Polices de secours : Trebuchet, Lucida, Arial, sans serif ;

# VIII. Conception de la base de données

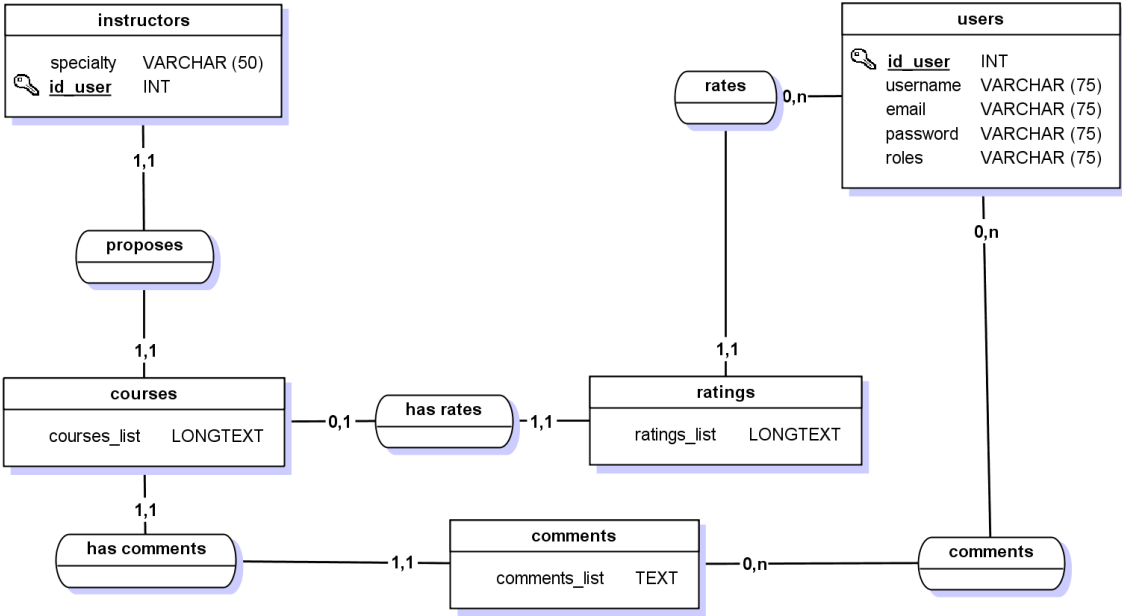
## A. DuckDrive API

| Article         |
|-----------------|
| title           |
| author          |
| content         |
| createdAt       |
| updatedAt       |
| category        |
| image           |
| alt             |
| intro           |
| chapters        |
| chaptersTitles  |
| chaptersContent |

Le modèle **article** de notre API doit servir a stocker des texte plus ou moins complexes. Grâce au **chapters**, on a la possibilité de garder le texte ordonné, mais la propriété **content** permet de garder un texte simple sous la main.

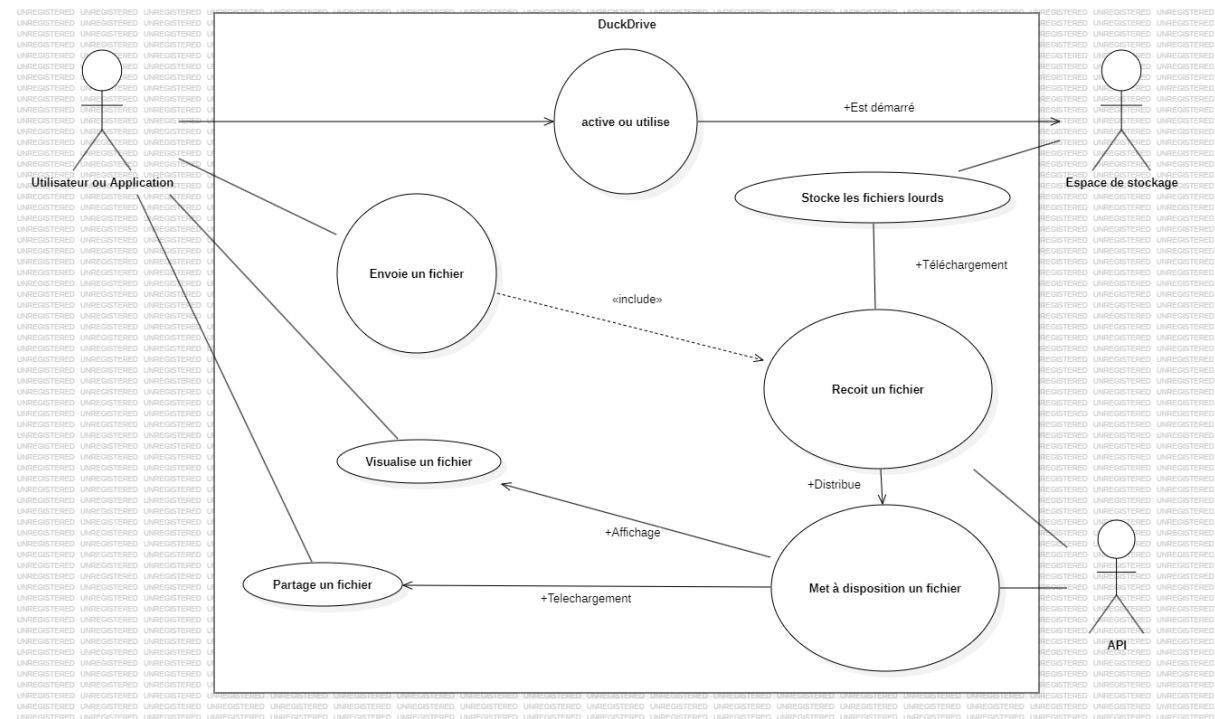
Seuls les propriétés **id**, **author**, et **createdAt** / **updatedAt** sont requises afin de garantir la compatibilité du modèle avec tout type de contenu (esprit « drive » ) notamment en dehors de Hedgehog School.

## B. Hedgehog School

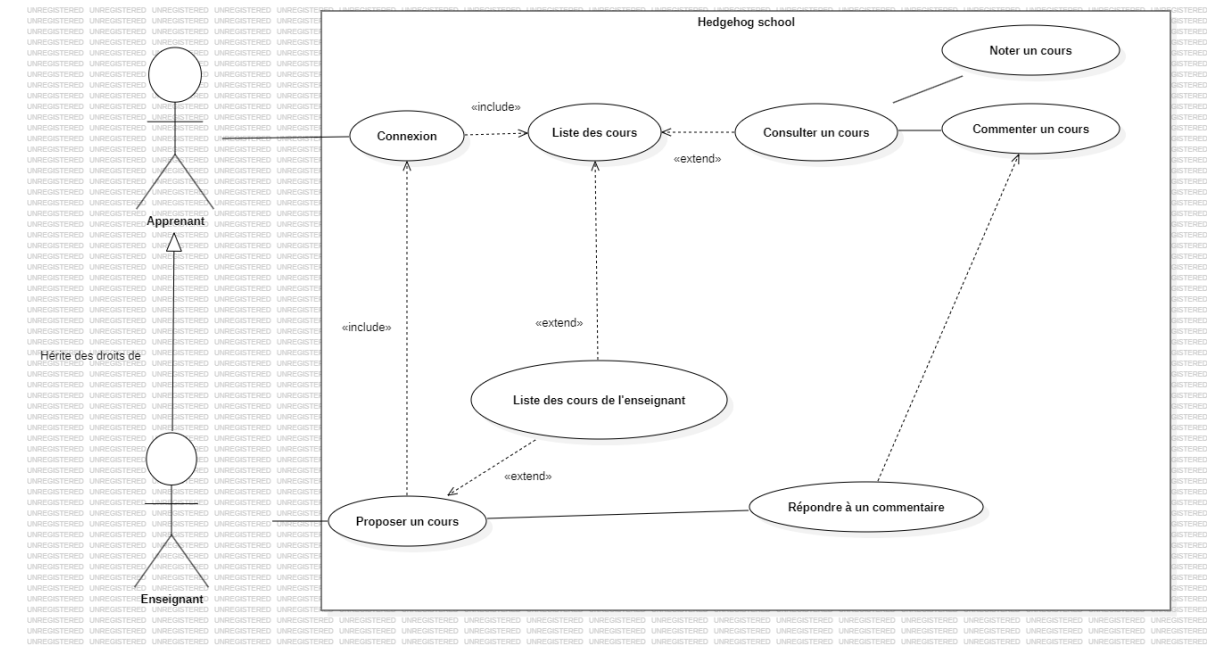


# IX. Conception de back-end

## A. Use Case Duck Drive (API)

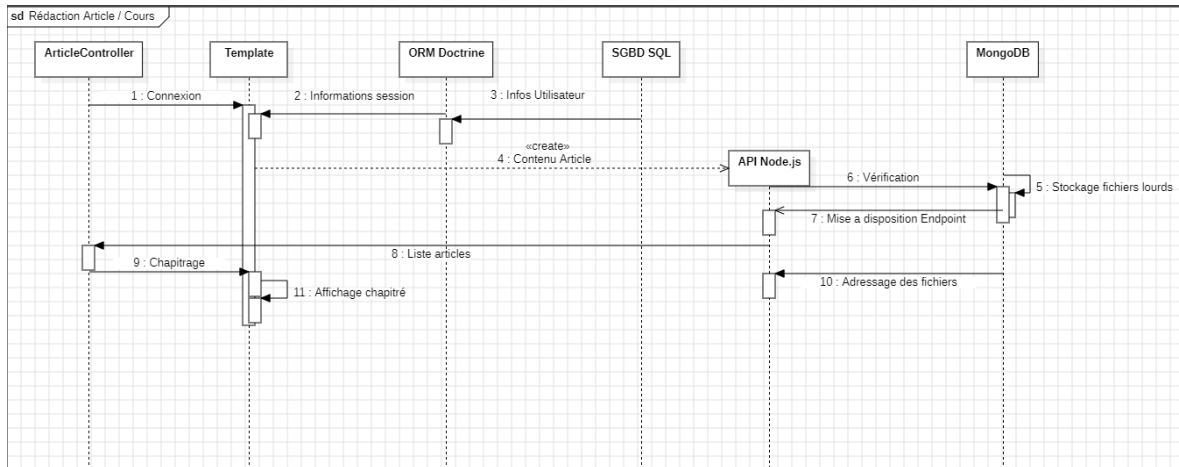


## B. Use Case Hedgehog School



## X. Diagrammes de séquence

Affichage d'un article sur l'application Hedgehog School

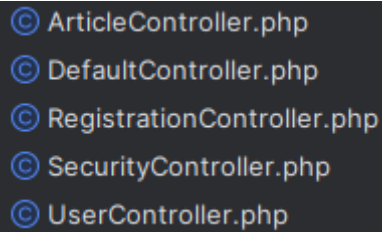


Ici, diagramme de séquence pour la création d'articles avec le stockage distant via l'API pour illustrer la conception multi-couches (CF chapitre suivant)

## XI. Conception multi-couches :

### A. MVC

Notre application dispose d'une structure de type Model View Controller (MVC). Nous utilisons des **contrôleurs** pour gérer routes et fonctions, organisées par **modèles**.

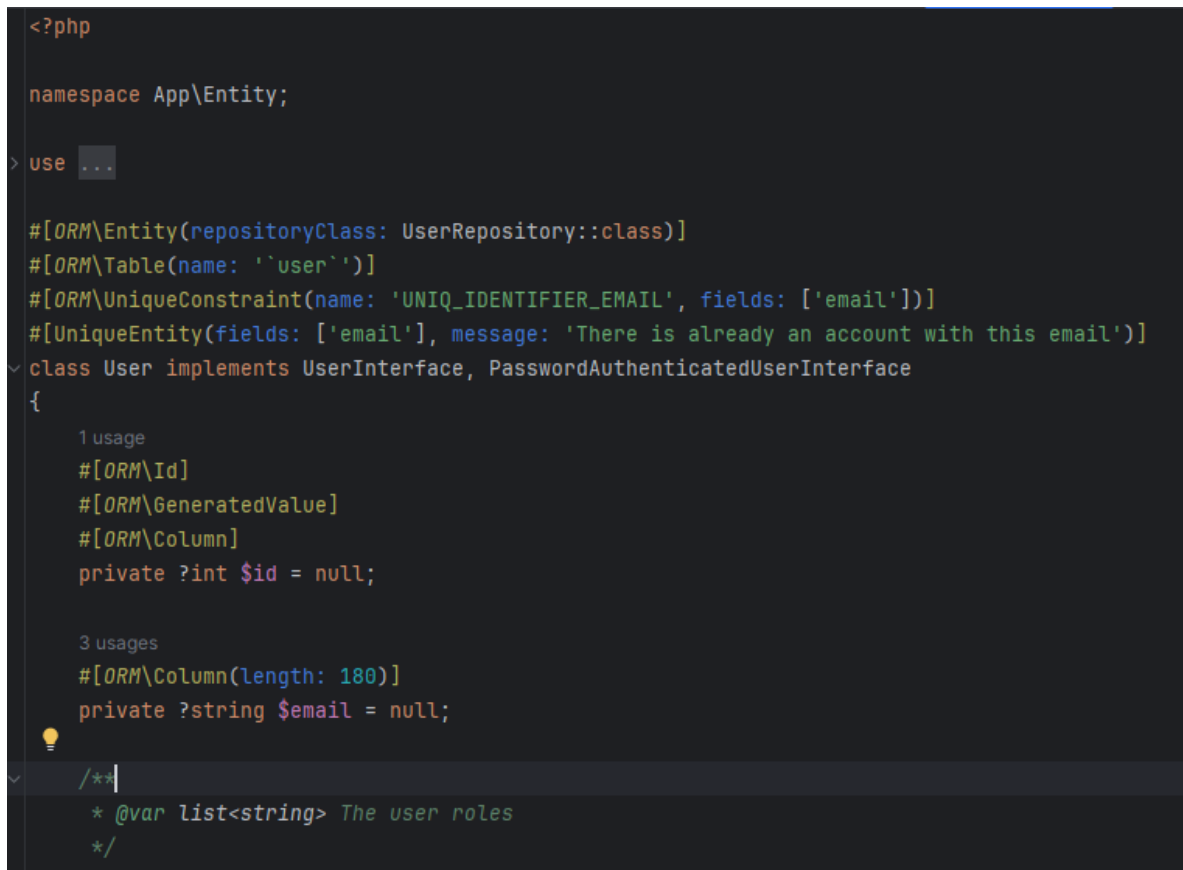


A dark-themed list of five PHP controller files, each preceded by a blue copyright symbol (©):

- ArticleController.php
- DefaultController.php
- RegistrationController.php
- SecurityController.php
- UserController.php

Les modèles en question sont d'une part les **entités symfony** sur lesquelles les contrôleurs sont basés,

*e.g entité **User** de Hedgehog School (symfony)*



```
<?php

namespace App\Entity;

> use ...

#[ORM\Entity(repositoryClass: UserRepository::class)]
#[ORM\Table(name: 'user')]
#[ORM\UniqueConstraint(name: 'UNIQ_IDENTIFIER_EMAIL', fields: ['email'])]
#[UniqueEntity(fields: ['email'], message: 'There is already an account with this email')]
class User implements UserInterface, PasswordAuthenticatedUserInterface
{
    1 usage
    #[ORM\Id]
    #[ORM\GeneratedValue]
    #[ORM\Column]
    private ?int $id = null;

    3 usages
    #[ORM\Column(length: 180)]
    private ?string $email = null;

    /**
     * @var list<string> The user roles
     */
```

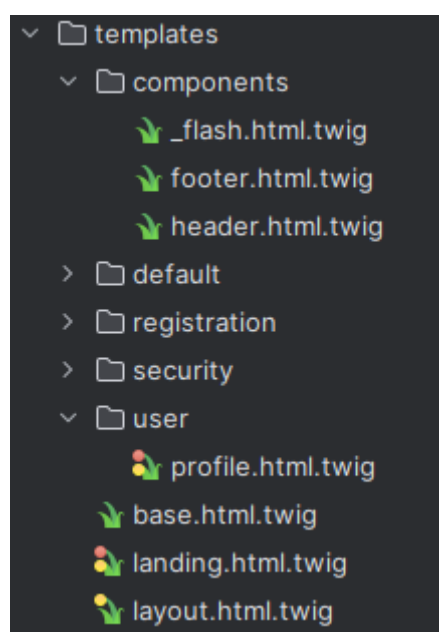
D'autre part, les objets reçus via l'API DuckDrive (notamment les medias et articles) sous la forme de schémas **mongoose**

e.g le modèle **Article** de DuckDrive (node.js)

```
const mongoose = require('mongoose');
const Schema = mongoose.Schema;

const ArticleSchema = new Schema({
  title: {
    type: String,
    required: true
  },
  image: {
    type: String
  },
  alt: {
    type: String
  },
  intro: {
    type: String
  },
  content: {
    type: String
  },
});
```

Nos vues sont gérées au moyen du templating via **twig**. Avec des éléments d'interfaces dynamiques (**include** et **renderController**)

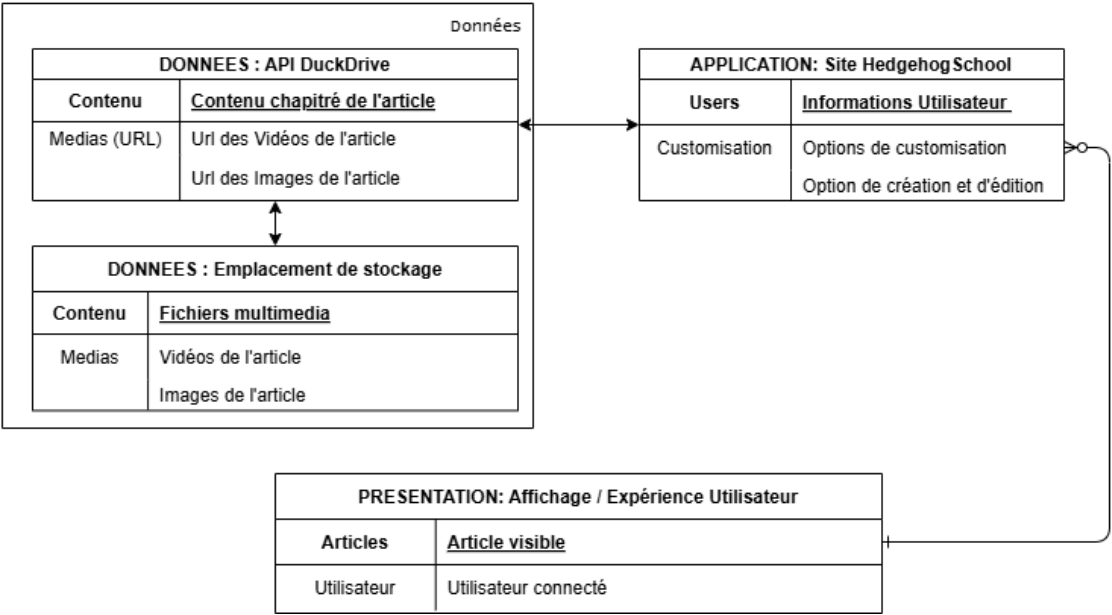




B. Architecture à 3 niveaux<sup>ii</sup>

L'application fonctionne avec une synergie de deux logiciels. L'objectif est de pouvoir gérer le fonctionnement du site (gestion des utilisateurs, customisation, édition) dans une application Symfony légère et fluide tandis que tout ce qui est plus lourd et constitutif du contenu du site sera servi par l'API DuckDrive, conçue pour l'envoi de fichier et qui sera amenée à stocker des contenus lourds comme des medias. Attention cependant, une petite partie des données, nécessaire au fonctionnement du site, sera interne a Symfony (les utilisateurs principalement) et donc stocké dans une BDD relationnelle SQL

Architecture de l'application



## XII. Sécurité

Notre site cherche à être le plus sécurisé possible, même si son code a vocation à rester ouvert et personnel. Par conséquent nous devons protéger les données des utilisateurs des six types d'attaques communs. Voici ce que nous avons mis en place :

### A. Attaque XSS

Nous évitons les attaques XSS pour tous les formulaires et en utilisant les fonctions de formulaires Symfony afin de bénéficier de l'échappement automatique des balises HTML et de la tokenisation de ceux-ci ce qui empêche d'envoyer la moindre balise et donc le moindre script.

### B. CSRF

Les CSRF Tokens de chaque formulaire sont attendus à la soumission, ce qui empêche aux utilisateurs malveillants de profiter d'une session ouverte

### C. Injection SQL

Toutes les requêtes SQL sont préparées et protégées par l'ORM doctrine dans Symfony. En ce qui concerne l'API, elle requête côté serveur avec une structure RESTful qui ne permet pas d'injecter dans MongoDB.

### D. Attaque brute force

Les mots de passe autorisés par les Passwords Symfony sont automatiquement mis aux normes modernes, nous permettant de suivre les bonnes pratiques de sécurité. Les algorithmes de hachage, chiffrement, cryptage et gestion des mots de passe sont également dynamiques et évolutifs grâce à des solutions comme `PasswordHasher` intégrée dans notre application.

### E. Politique de mot de passe (CNIL ou ANSSI), RGPD

La politique de mot de passe du site s'aligne avec celle du CNIL par souci de portée de l'application (nationale). Les données sont accessibles et supprimables sur simple demande de l'utilisateur connecté.

### F. Hashage des mots de passe

`PasswordHasher` choisit automatiquement un algorithme de hachage moderne et efficace pour notre application et garantit la fiabilité du hachage.

## XIII. Politique de test

### A. Test Unitaire

Nous testons les éléments à leur fonctionnement dans l'API afin de vérifier que les données sont bien accessibles via les requêtes

Par exemple : on teste la fonction `createArticle` indépendamment des autres parties de l'application.

*e.g Output du test de la fonction de création d'article via JEST dans l'API (node.js)*

```
FAIL controllers/ArticleController.spec.js
  • createArticle > should create a new article when title is provided

    expect(jest.fn()).toHaveBeenCalledWith(...expected)

    - Expected
    + Received

    Object {
      "message": "Nouvel article crée",
    -   "summary": Any<Object>,
    +   "summary": "Test Article",
    },

    Number of calls: 1
```

Ici nous avons un envoi vide de requête sur l'API qui est donc rejeté par l'application (Absence du champ requis 'title').

## B. Test d'Intégration

Ce type teste plusieurs composants travaillant ensemble, par exemple ici nous avons choisi la connexion à une base de données et le démarrage du serveur.

*e.g Test de la connexion à la BDD Mongo (JEST, node.js)*

```
• MongoDB Connection and Server Startup > devrait se connecter à MongoDB et démarrer le serveur

TypeError: startServer is not a function

13 |
14 |     test('devrait se connecter à MongoDB et démarrer le serveur', async () => {
> 15 |         server = await startServer(); // Démarre le serveur
    |                        ^
16 |         expect(server).toBeTruthy(); // Vérifie que le serveur est démarré
17 |     });
18 |

at Object.startServer (index.spec.js:15:24)

• MongoDB Connection and Server Startup > devrait lancer une erreur en cas de connexion MongoDB échouée

TypeError: startServer is not a function

21 |         throw new Error("Échec de connexion à MongoDB");
22 |     });
> 23 |     await expect(startServer()).rejects.toThrow("Échec de connexion à MongoDB");
    |                        ^
24 |     });
25 | });

at Object.startServer (index.spec.js:23:22)
```

`describe('MongoDB Connection and Server Startup')` est un test d'intégration. Il vérifie la connexion à MongoDB via `connectDB` et s'assure que le serveur démarre correctement, ou qu'il gère bien une connexion échouée. Ce test implique une interaction avec MongoDB, donc il teste bien l'intégration de notre application avec la base de données.

### C. Test Fonctionnel

Ici notre *setup* crée un environnement de test pour connecter et ajouter des utilisateurs

*e.g* préparation de l'environnement Symfony (PHPUnit) pour des tests sur les utilisateurs

```
protected function setUp(): void
{
    $this->client = static::createClient();
    $container = static::getContainer();
    $em = $container->get('doctrine.orm.entity_manager');
    $userRepository = $em->getRepository(User::class);

    // Remove any existing users from the test database
    foreach ($userRepository->findAll() as $user) {
        $em->remove($user);
    }

    $em->flush();

    // Create a User fixture
    /** @var UserPasswordHasherInterface $passwordHasher */
    $passwordHasher = $container->get('security.user_password_hasher');

    $user = (new User())->setEmail('email@example.com');
    $user->setPassword($passwordHasher->hashPassword($user, plainPassword: 'password'));

    $em->persist($user);
    $em->flush();
}
```

Une fois notre *setup* fini on peut tester la validité des email puis des autres champs, vérifier qu'on ne révèle pas quel champ exactement pose problème, et tester que nos erreurs sont bien gérées

La classe `LoginControllerTest` (`setUp()` + `testLogin()`) est un test fonctionnel pour plusieurs raisons :

1. Il teste le flux complet de l'authentification en simulant des interactions réelles utilisateur :
  - Navigation vers la page de login
  - Soumission de formulaires
  - Suivi des redirections
  - Vérification du contenu affiché
2. Il teste plusieurs scénarios d'utilisation :
  - Tentative de connexion avec un email inexistant
  - Tentative avec un mot de passe incorrect
  - Connexion réussie avec des identifiants valides
3. Il utilise l'infrastructure complète de l'application :
  - Utilise un vrai client HTTP (`WebTestCase`)
  - Interagit avec la base de données
  - Utilise le container de services
  - Utilise le système de routing
  - Utilise le système de sécurité de Symfony
4. Il vérifie des aspects de sécurité importants :
  - Les messages d'erreur ne révèlent pas si l'utilisateur existe
  - Le hachage des mots de passe
  - Les redirections appropriées

*e.g on teste la connexion, la pérennité de la session et des données utilisateurs et la levée d'erreur(PHPUnit, Symfony)*

```
class LoginControllerTest extends WebTestCase
{
    public function testLogin(): void
    {
        // Denied - Can't login with invalid email address.
        $this->client->request( method: 'GET', uri: '/login');
        self::assertResponseIsSuccessful();

        $this->client->submitForm( button: 'connexion', [
            '_username' => 'doesNotExist@example.com',
            '_password' => 'password',
        ]);

        self::assertResponseRedirects( expectedLocation: '/login');
        $this->client->followRedirect();

        // Ensure we do not reveal if the user exists or not.
        self::assertSelectorTextContains( selector: '.alert-danger', text: 'Invalid credentials.');
```

```
        // Denied - Can't login with invalid password.
        $this->client->request( method: 'GET', uri: '/login');
        self::assertResponseIsSuccessful();

        $this->client->submitForm( button: 'Sign in', [
            '_username' => 'email@example.com',
            '_password' => 'bad-password',
        ]);

        self::assertResponseRedirects( expectedLocation: '/login');
        $this->client->followRedirect();

        // Ensure we do not reveal the user exists but the password is wrong.
        self::assertSelectorTextContains( selector: '.alert-danger', text: 'Invalid credentials.');
```

```
        // Success - Login with valid credentials is allowed.
        $this->client->submitForm( button: 'Sign in', [
            '_username' => 'email@example.com',
            '_password' => 'password',
        ]);

        self::assertResponseRedirects( expectedLocation: '/');
        $this->client->followRedirect();

        self::assertSelectorNotExists( selector: '.alert-danger');
        self::assertResponseIsSuccessful();
    }
}
```

### D. Test de non-régression

On prend le test de la fonction `GetArticleById()` pour la non-régression. Si il y a un souci dans le modèle de données ou des requêtes qui bougent dans les Templates twig ou l'envoi de donnée des contrôleurs, exécuter ce test permet de savoir que le code et le gros des requêtes fonctionne toujours.

*Ce test est en place dès le début et échouera au moindre souci avec MongoDB (Jest, node.js)*

```
describe('getArticleById', () => {
  // Retrieves an article successfully when a valid ID is provided
  it('should return the article when a valid ID is provided', async () => {
    const req = { params: { id: '672e3ef7e8e85397b5d435e7' } };
    const res = {
      json: jest.fn(),
      status: jest.fn().mockReturnThis()
    };
    const article = { title: 'Test Article', content: 'Test Content' };
    jest.spyOn(Article, 'findById').mockResolvedValue(article);

    await getArticleById(req, res);

    expect(Article.findById).toHaveBeenCalledWith(new mongoose.Types.ObjectId(req.params.id));
    expect(res.json).toHaveBeenCalledWith(article);
  });

  // Handles invalid article ID format gracefully
  it('should return 400 status when the ID format is invalid', async () => {
    const req = { params: { id: 'invalid-id' } };
    const res = {
      json: jest.fn(),
      status: jest.fn().mockReturnThis()
    };

    await getArticleById(req, res);

    expect(res.status).toHaveBeenCalledWith(400);
    expect(res.json).toHaveBeenCalledWith({ message: 'Invalid article ID format' });
  });
});
```

### E. Test de montée de charge

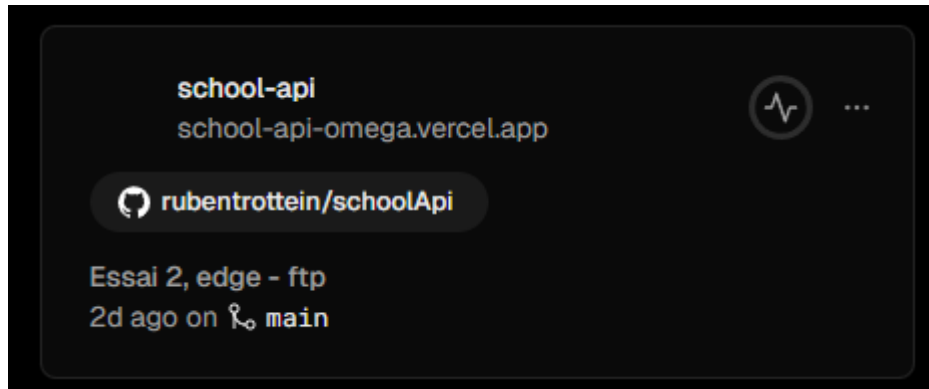
WIP ajouter un test de montée de charge (PHPUnit, Symfony)

## XIV. Déploiement et mise en production

### A. API :

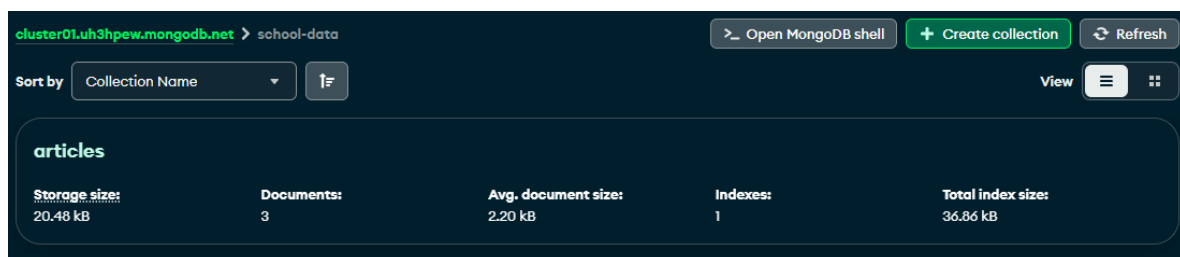
**Vercel** déploie facilement une API node.js avec un backend **mongoDb** ouvert a Vercel (mongoDBAtlas) depuis un dépôt géré avec **Git**.

Projet vercel



Grâce à MongoDBCompass on peut vérifier la BDD de l'API a tout moment.

Collection MongoDB



A chaque push sur la branche main, un nouveau déploiement est effectué. Les données de l'API sont disponibles sur simple interrogation de l'API (Sera protégée par une clé API en production, désactivée en dev pour simplifier le développement)

### B. Symfony :

Le déploiement continu (CD) est géré par **Github Actions**, qui permet une integration continue grâce au versioning de **Git**



## XV. Veille technologique

### A. Veille globale

Ma veille s'organise autour des problématiques métier dues à l'illectronisme. Elle passe par une veille de la sociologie des bénéficiaires et des avancées en terme de politique et de philosophie autour des problèmes d'accès a des solutions.

On cherche en priorité des clients et des cas d'usage pour l'application :

- Veille dans les mairies pour les projets en relation avec la fracture numérique
- Suivi des avancées en termes de tutoriels
- Test et utilisation de LMS différents pour les tutoriels en général dans ma veille métier.
- Suivi des technologies efficaces et reconnues modernes

Outils de veille

- Comptes de réseaux sociaux (LinkedIn, X, Youtube)
- Newsletter de mairies
- Recherche d'emploi autour du domaine

### B. Veille de sécurité

L'objectif de nos application étant de rester simple et robustes (et ultimement, déconnectées) nous surveillons

A l'international :

- les recommandations OWASP et notamment son top ten pour garder nos application basiquement sûres

En France :

- celles de l'ANSSI pour les réglementations en vigueur
- celles de la CNIL pour le RGPD et la souveraineté des données afin d'avoir des versions propres et sécurisées durables dans le temps

## XVI. Difficultés rencontrées

Frameworks et dépendances

Difficile de dépendre de code qui n'émane pas de nous. Nous avons choisi de limiter l'accès aux solutions toutes prêtes à la fois dans le concept du site et son design.

Stocker sans offre pro

Complicé de gérer finement des données avec Vercel ou MongoDB sans engager des frais.

L'objectif de souveraineté nous a amené a choisir un stockage déporté personnel (OVH) mais le produit final devrait y avoir gagné une versatilité dans le choix du stockage.

D'une entité à l'autre : node.js et Symfony

Les échanges entre l'API et l'application sont le principal challenge, auquel vient s'ajouter le fetch de fichiers ailleurs. La V1 ne gerera donc pas les soucis de téléchargement et d'affichage, mais des solutions de thumbnail et de streaming seront utilisées en fin de production.

Chapitrage des articles

Le contenu doit être suffisamment modulaire pour permettre une navigation basique dans les pavés de texte qui constitueront des articles Hedgehog School.

Il a été difficile d'envisager toutes les solutions, et l'éditeur Wysiwyg est parfois capricieux.

### Versioning de trois éléments

Le déploiement continu concernant l'API, le site et l'espace de stockage alourdit considérablement les phases de test et le développement voire même le lancement quotidien de la solution tout comme son exportation.

### Situation personnelle

Ayant dû me déplacer plusieurs fois J'ai eu beaucoup de soucis d'export et de versioning qui m'ont considérablement ralenti (perte de données, merge interminables dus au fait que j'ai travaillé sur la mauvaise branche etc. qui viennent s'ajouter aux ralentissements dus à la multi couche

## XVII. Post-projet

### A. Elaboration d'une FAQ

Il est important d'avoir des retours utilisateurs pour améliorer l'application autour de besoins métier et de connaître les failles en général.

### B. Version fermée et installable (pour les associations)

L'application devrait normalement être une solution desktop installable permettant de travailler offline afin d'être personnelle et customisable, tout en protégeant des petits systèmes modestes tels que ceux des associations et particuliers.

La forme finale de toute la solution devrait permettre :

- D'envoyer un client sur une machine personnelle ou à un ami
- De stocker depuis sa propre machine tant qu'elle est allumée ou avec un Wake Up Lan disponible
- D'être 100% inutilisable online
- De créer des ponts directs entre machines

### C. Versions thématiques de l'application

Les retours utilisateurs pourraient permettre d'adapter l'application à des logiques métiers différentes de la formation, ce qui permettrait de faire des versions résilientes de la solution pour des environnements plus précis

### D. Suite logicielle D.A.N

L'application de visualisation reste à concevoir. Une interface pour le DuckDrive également afin de faciliter le management des fichiers et de gérer facilement le stockage externe

### E. Refonte graphique

Le Template est actuellement relativement hasardeux et peut être pas assez générique. J'ai installé de quoi changer le thème, j'aimerais en proposer plusieurs qui permettraient de simplifier ou complexifier l'interface en fonction de l'utilisateur ainsi que de rendre l'expérience utilisateur plus agréable

## XVIII. CGU et RGPD

Les logiciels demeurent libres et modifiables par l'utilisateur. Licence d'utilisation CC0/MIT selon les cas. Seul le support et la formation sont assurées par nos services

Les fichiers sont tous reliés a un auteur. A la clôture du compte de celui-ci nous supprimons l'ensemble de ses fichiers selon son désir.

## XIX. Documents annexes et liens :

### A. Lien de la présentation :

[https://www.canva.com/design/DAF6sXVYI3Q/FVZI8wNTKFcZ8yvtqZUkSQ/edit?utm\\_content=DAF6sXVYI3Q&utm\\_campaign=designshare&utm\\_medium=link2&utm\\_source=sharebutton](https://www.canva.com/design/DAF6sXVYI3Q/FVZI8wNTKFcZ8yvtqZUkSQ/edit?utm_content=DAF6sXVYI3Q&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton)

### B. Questionnaire d'accueil :

<https://forms.office.com/Pages/DesignPageV2.aspx?subpage=design&FormId=-BYwN6I57U6A0hAM6UjZYENXknfPEnFEoVLah48JL4RUNFc4RzY1Tk9ZVIRGVjIHWEG5S1hDUVdXTC4u&Token=ad18ab737e4d4e73a813c06d664acbbe>

## XX. Contact

### A. Adresse de l'entreprise

54 rue des Acacias 75000 Paris

### B. Email

contact]@danproject.com

### C. Téléphone

+33659185156

## XXI. Crédits

### A. Images

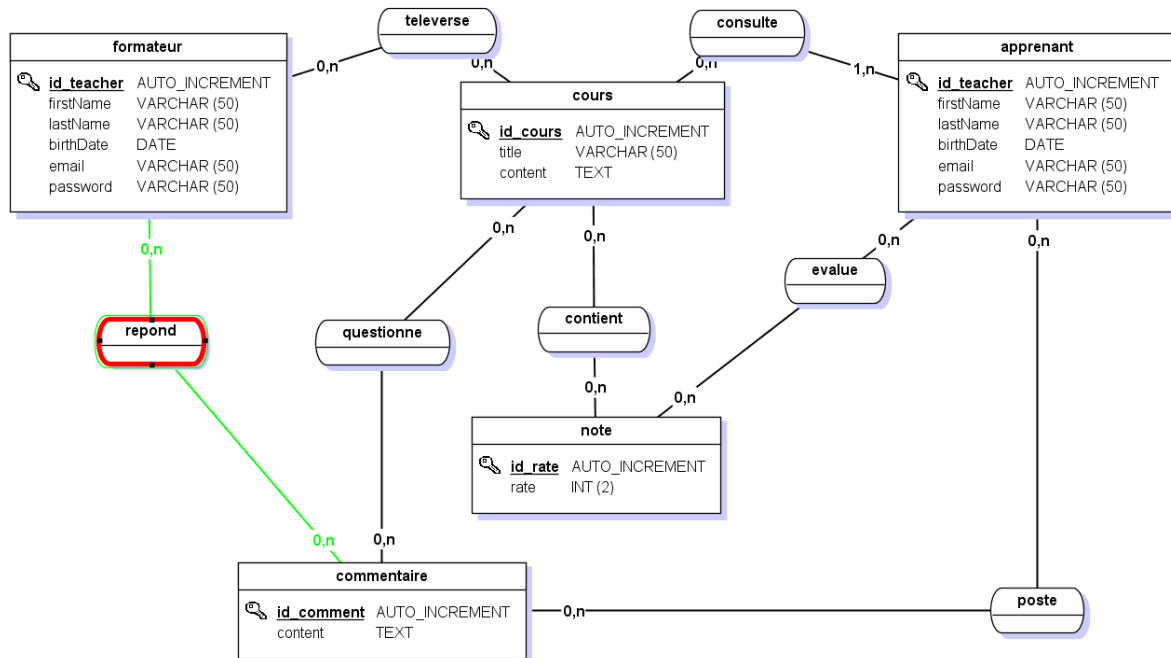
- Paresseux - Image par [WikilImages](#) de [Pixabay](#)
- Hérisson - Image par [Mr\\_SRC](#) de [Pixabay](#)
- Canard - Image par [Christel SAGNIEZ](#) de [Pixabay](#)
- Canards - Image par [Mabel Amber, who will one day](#) de [Pixabay](#)

### B. Vidéos

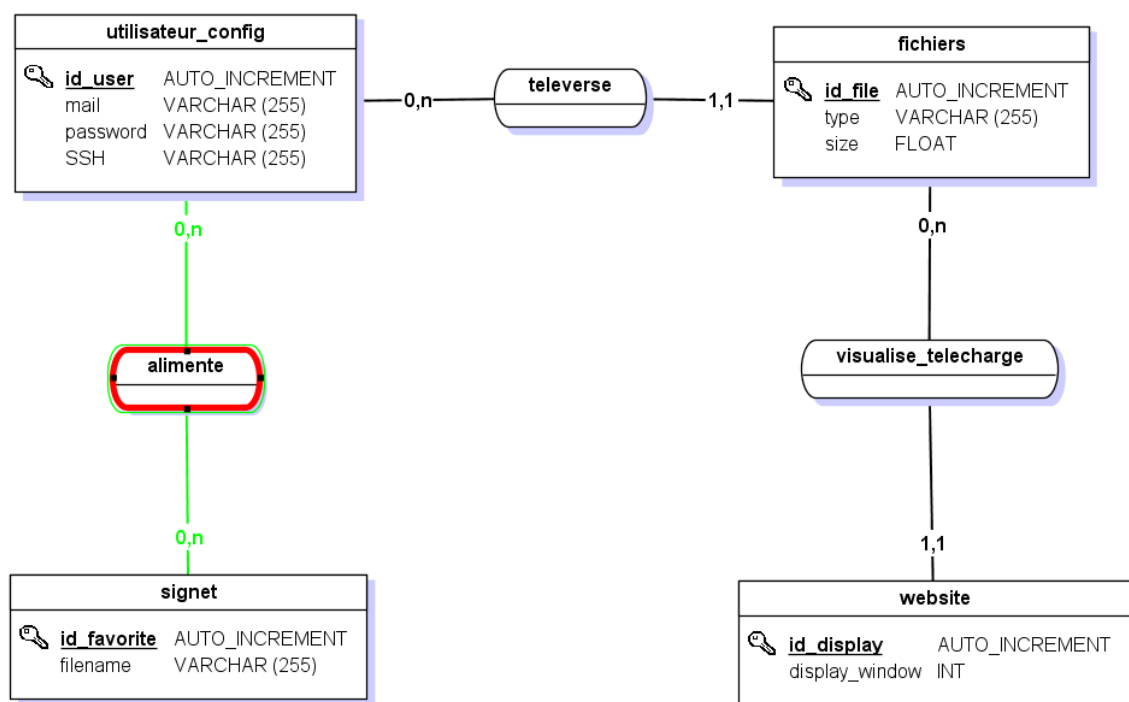
### C. Autres

## XXII. Annexes

1. Fonctionnement general des bdd synchronisées (representation des interaction des utilisateurs avec un cours obtenu depuis l'API)

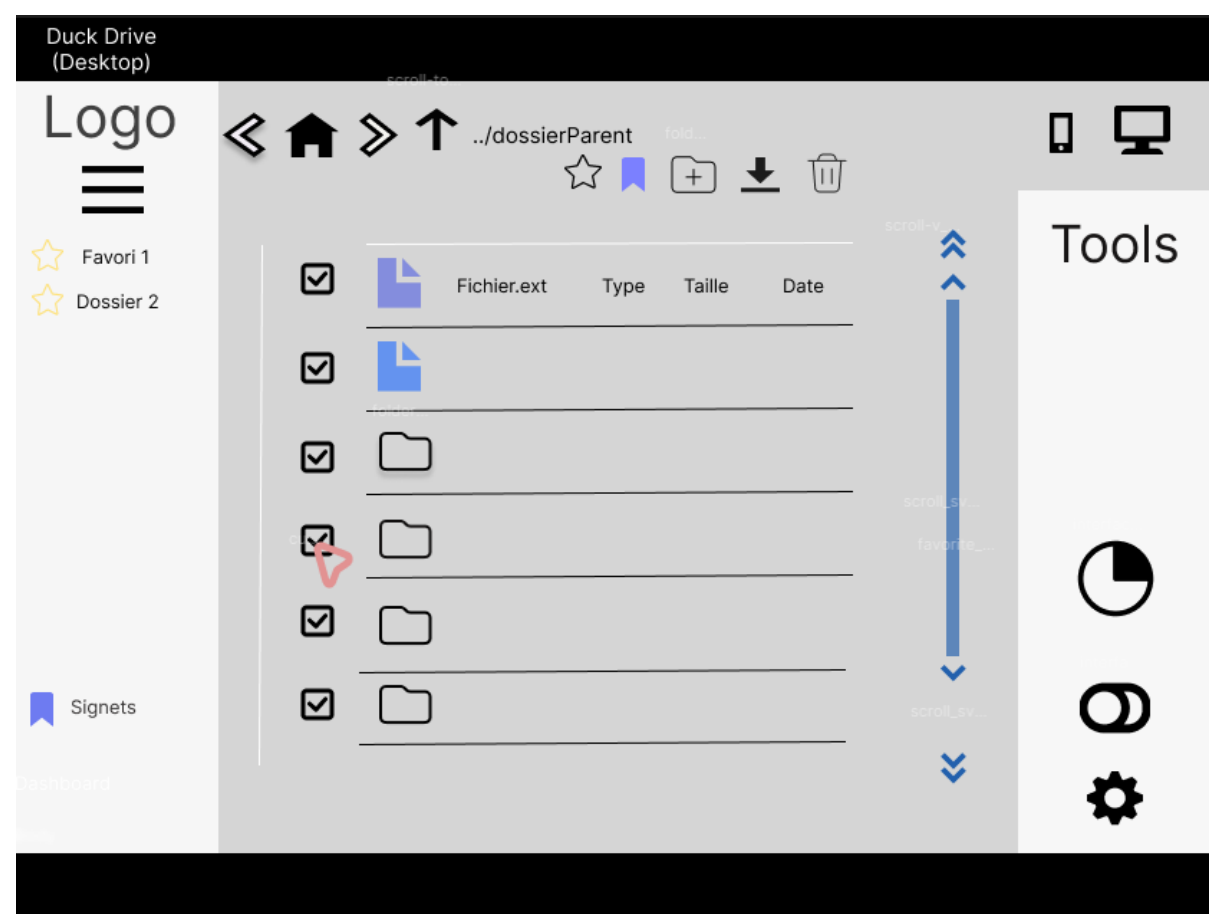


2. MCD de la future application de visualization de l'API DuckDrive

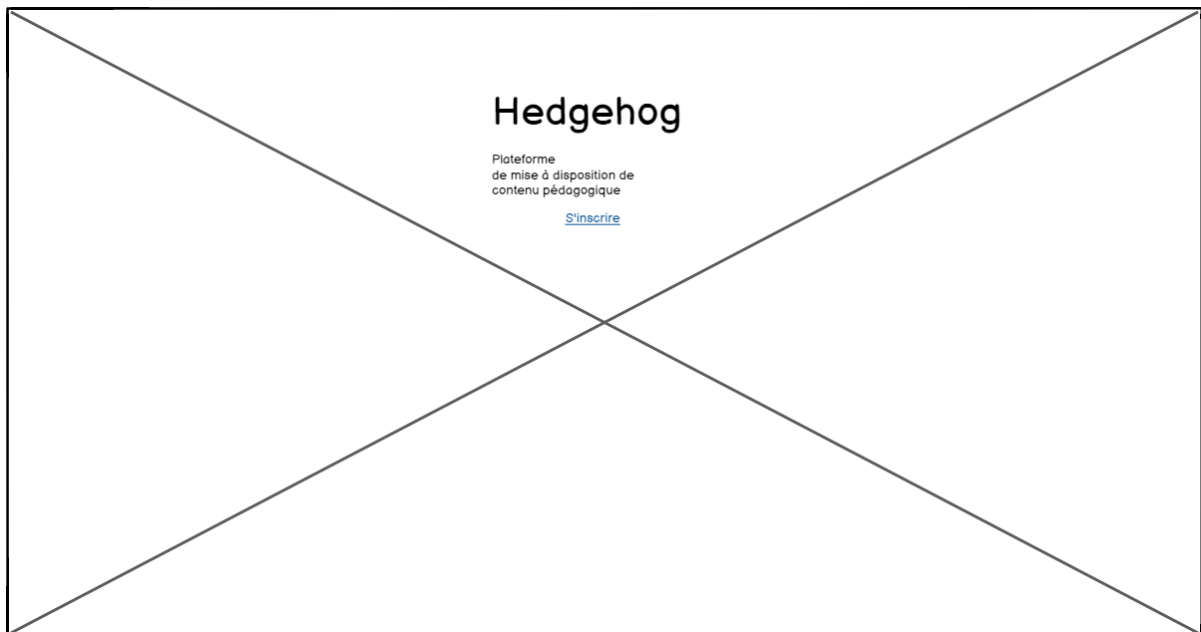


3. Wireframes DuckDrive

Zoning de l’explorateur de fichiers (Dashboard PWA)



Version popup des formulaires derrière une page d'accueil ou de demande de connexion



Entrez votre E-Mail

Entrez / Choisissez un mot de passe

☐ J'accepte les conditions d'utilisation du site

S'inscrire

{Authentication Google Auth}



---

<sup>i</sup> Régis Bogot, « Le fossé numérique en France », *Cahier de Recherche* n° 177, [CREDOC](#), novembre 2002, 87 p. [[lire en ligne](#)]

<sup>ii</sup> <https://www.ibm.com/fr-fr/topics/three-tier-architecture>