



## M3104 – Programmation Web côté serveur (2020/21)

### Sprint 1 – document de lancement

#### Objectifs

- Appropriation de l'environnement de développement (serveur de développement, IDE PhpStorm, dépôt Git) et d'eduScrum
- Conception et implémentation d'une application web simple, reposant sur une base de données (développement en PHP objet)
- Éléments simples d'interface utilisateur (e.g. formulaires), transmission de paramètres à un script serveur (sans JS)
- Structuration d'un projet Web (et premiers éléments informels liés au pattern MVC)

#### Apprentissages liés

3 TP de 1h30 sont programmés tout au long de ce 1<sup>er</sup> projet pour travailler plus en détail sur des notions qui pourraient vous permettre d'aller plus loin dans la réalisation du 1<sup>er</sup> sprint (TP 3 et TP 4) et préparer le suivant (TP 5) :

- TP3 : Interactions client/serveur : AJAX
- TP4 : Génération dynamique de documents (images, PDF, etc.)
- TP5 : Application web à état, POO avancée, *design patterns*

#### Organisation du sprint 1

Ce sprint durera 15h (x4 développeurs) réparties sur 8 semaines. Après la formation des équipes d'étude, le sprint débutera par une réunion de planification et se terminera par une revue de sprint (évaluée pour la note pratique) et une rétrospective. L'analyse détaillée des activités à réaliser et leur planification seront l'objet de la réunion de planification.

#### Exigences – présentation du besoin client

Le client souhaite la production d'une application web de type CMS (*Content Management System*, ou système de gestion de contenus).

Le but de ce sprint sera d'en poser les bases et de réaliser la partie permettant à un auteur de gérer ses articles (saisie, modification, publication, etc. d'un ou plusieurs articles).

Cette application sera naturellement multi-utilisateur et un utilisateur donné ne pourra gérer que les articles pour lesquels il est autorisé à le faire (ses propres articles et ceux pour lesquels il en a reçu le droit).

Un article est en mode « brouillon » tant qu'il n'a pas été « publié » par son auteur (ou par un autre utilisateur ayant le niveau d'autorisation nécessaire).

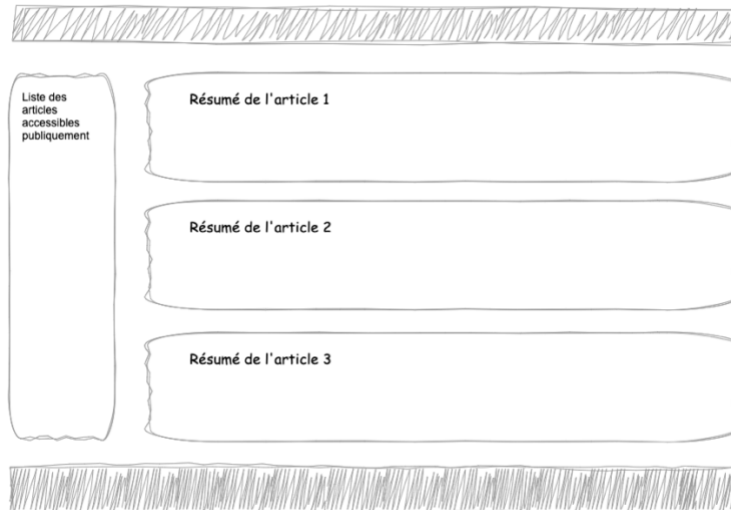
Des métadonnées (mots-clés, thématique, auteur, etc.) seront associées à chaque article et permettront de gérer les droits et d'effectuer des recherches sur la base des articles.

Il sera également possible de visualiser (et éventuellement d'imprimer) une synthèse (auteur, titre, date de publication, résumé, etc.) des articles accessibles à l'utilisateur connecté.

Cas d'utilisation :

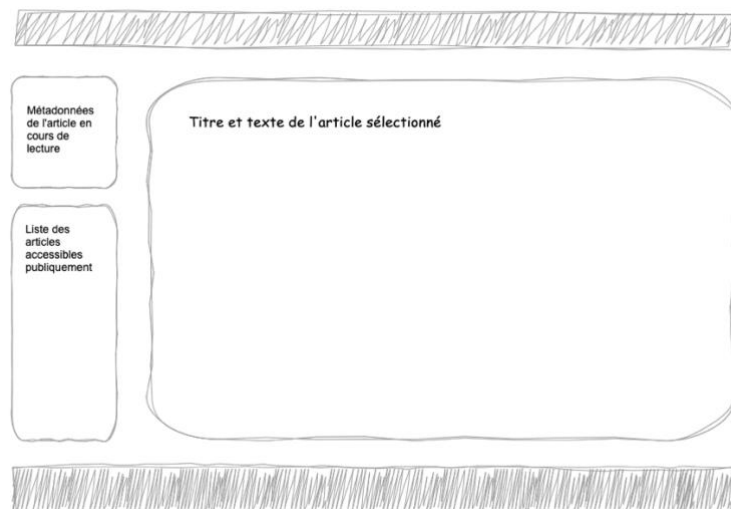
- Page d'accueil → navigation non connectée
- Page d'accueil → navigation non connectée (lecture d'un article)
- Page d'accueil → connexion
- Page d'accueil → connecté
- Affichage d'un article non publié

### Page d'accueil > Navigation non connectée



En mode non connecté, l'application permet de lire les articles publiés. Elle présente une liste de tous les articles (titres) ainsi qu'une mise en avant des derniers articles (titre et résumé). Les articles peuvent ensuite être visualisés en entier après sélection par l'utilisateur.

### Page d'accueil > Navigation non connectée (lecture d'un article)



En mode non connecté, il est seulement possible de lire les articles publiés (avec les métadonnées associées).

## Page d'accueil > Connexion

L'usage de l'application peut se faire selon deux modes : connecté, permettant l'accès aux fonctions d'édition et de gestion, ou non connecté, permettant seulement la lecture des articles publiés.

La fonctionnalité « connexion » est obligatoire à implémenter, mais le mécanisme permettant de créer un compte est facultatif durant ce premier sprint (une inscription directe dans la base de données est possible).

## Page d'accueil > Connecté

Une fois connecté, l'utilisateur peut accéder à un certain nombre d'outils (ajouter un article, visualiser / modifier son profil, etc.) et à la liste de tous les articles qui lui sont accessibles (publiés ou non).

## Affichage d'un article non publié

Lorsqu'un article non publié est sélectionné, l'utilisateur connecté peut le publier, le modifier (que ce soit le texte ou les métadonnées), le supprimer ou l'archiver (il n'est pas publié, mais n'est plus non plus un brouillon, il est dans un état spécifique : archivé).

## Contraintes de mise en œuvre

Pour ce premier sprint, il n'y a pas de contraintes particulières sur l'architecture de votre application. Vous utiliserez PDO pour gérer la connexion à la base de données et des requêtes préparées pour récupérer les données.