

# P1 - Note de clarification

---

Version 3 (29 septembre 2017)

## Contexte et objectifs

---

Le projet prend place dans le cadre de l'UV NA17 (conceptions de bases de données en autonomie).

## Objectifs généraux du projet

Le projet EtuDoc est un système en ligne de gestion de documents internes à l'UTC, s'appuyant sur une base de données relationnelles dans un premier temps. Il permettra de centraliser l'ensemble des travaux étudiants (rapports de stage, mémoires, rapports...), afin de les rendre accessibles et partageables.

Les administrateur.ice.s pourront de plus supprimer, archiver et modifier les documents de la base de données.

## Groupe 10

- Anh Tu Nguyen
- Alexandre Brasseur
- Kyllian Chartrain
- Camille Beaudou

## Contraintes technologiques

- Le projet doit être réalisé *via* une architecture LAPP (Linux, Apache, PostgreSQL, PHP).

## Livrables et dates butoir

---

Date	Livrable
15/09	Rédaction de la note de clarification & du product backlog (V1)
22/09	Analyse : Modèles conceptuel et logique de données + V2 de la NDC et PB
29/09	Première version de la base de données relationnelle
Non définie	Base de données non-relationnelle
Non définie	Application de démonstration

## Cahier des charges : reformulation & précisions

---

## Documents stockés

Un document stocké est identifié par :

- un code unique
- le code et le nom de l'UV associée
- les auteur.ice.s (identifié.e.s grâce à leur login UTC)
- les autres personnes impliquées dans sa production
- la date de publication (automatiquement associée au document)
- un titre
- une courte description
- le type (par exemple : rapport de stage, thèse, rapport de TX)
- la licence
- la catégorie (par exemple : biologie, mécanique des fluides, histoire, philosophie). Un document pourra éventuellement apparaître dans plusieurs catégories.
- un indicateur d'archivage du document
- le contenu du document.

Idéalement, les documents seront classés par catégorie. Cela permettra aux utilisateur.ice.s d'explorer rapidement la base de données afin de trouver ce qui les intéresse, sans qu'ils ou elles n'aient besoin de connaître les documents ou la base de données au préalable.

Les documents confidentiels ne sont pas gérés par le système, tous les documents sont lisibles par tou.te.s les utilisateur.rice.s.

## Types d'utilisateurs

Il y a 3 types d'utilisateur.rice.s :

- Anonyme : l'utilisateur.rice n'est pas connecté.e, il ou elle peut seulement consulter les documents.
- Membre : l'utilisateur.rice est connecté.e via le CAS et peut consulter et ajouter des documents. Il ou elle peut aussi modifier et supprimer ses propres documents.
  - Enseignant.e
  - Étudiant.e
- Administrateur.rice : il ou elle peut modifier, supprimer et archiver tout document (c'est-à-dire le rendre inaccessible au public). Il ou elle peut aussi ajouter des licences.

La connexion au CAS ne sera pas gérée dans le cadre du projet, aussi nous considérerons que chaque internaute a accès au dépôt de documents.

## Objectifs détaillés -- users story

### Dépôt de documents par les membres de l'UTC

L'utilisateur.rice (membre de l'UTC, connecté.e) souhaite déposer un document. Pour ce faire, il ou elle renseigne l'ensemble des informations requises par la base de données : type, titre, UV concernée, co-auteur.rice.s éventuel.le.s, autres personnes impliquées, une description courte, une catégorie et une licence. Le nom de l'utilisateur.rice est déjà connu puisqu'il ou elle est connecté.e *via* son login UTC. Ensuite, il ou elle pourra uploader le document.

## Consultation de documents

Un.e internaute (membre de l'UTC connecté.e ou non) peut consulter l'ensemble des documents déposés sur la plateforme EtuDoc. L'internaute pourra naviguer plus rapidement dans la base de données grâce à la classification rapide explicitée ci-dessus.

## Fonction recherche

Pour plus d'efficacité, l'utilisateur.rice (membre de l'UTC ou non) peut également utiliser la fonction "recherche"

- Une recherche simple recherche des termes-clefs dans la base de données (dans tous les attributs des documents)
- Une recherche avancée permet de choisir dans quels attributs des documents les termes doivent être recherchés.

## Modification de documents (administrateur.ice)

Un.e administrateur.ice peut modifier tous les attributs de tous les documents.

## Suppression ou archivage des documents (administrateur.rice)

Un.e administrateur.rice peut supprimer ou archiver tout document de la base de données.

## Gestion des licences (administrateur.ice)

Les administrateur.rice.s peuvent ajouter de nouvelles licences que les utilisateur.rice.s pourront ensuite associer à leurs documents.

## Éléments à implémenter

- Dépôt de documents
- Suppression de documents
- Archivages de documents (administrateur.rice.s)
- Modifications des attributs des documents (administrateur.rice.s)
- Classification rapide
- Fonctions de recherche
- Ajout de licences

---

# Product Backlog

---

## Phase 1

---

## Note de clarification & product backlog

- ☒ Rédaction de la note de clarification
- ☒ Rédaction du product backlog
- ☒ Retravail de la première version de la note de clarification
- ☒ Retravail de la première version du product backlog
- ☒ Retravail des secondes versions

## Modélisation

- ☒ Modélisation conceptuelle (UML)
  - ☒ Détermination des classes
  - ☒ Détermination des clés primaires
  - ☒ Détermination des relations
- ☒ Modélisation logique (Relations)
  - ☒ Traduction des classes
  - ☒ Traduction des associations
  - ☒ Vérification de la normalisation

## Création de la base de données relationnelle

- ☐ Implémentation de la base de données à partir du modèle relationnel :
  - ☐ Création des tables adéquates
  - ☐ Ajout des attributs et des relations
  - ☐ Mise en place des associations entre les différentes tables
  - ☐ Ajout d'éléments "tests" pour vérifier le bon fonctionnement de la base de données
- ☐ Gestion des requêtes de consultations de documents
- ☐ Gestion des requêtes de dépôt de documents
- ☐ Implémentation de la classification
- ☐ Gestion des requêtes de consultation à partir de la classification (sélection de documents à partir de leur type)
- ☐ Gestion des requêtes de recherche (simple)
- ☐ Gestion des requêtes de recherche (avancée)
- ☐ Création des possibilités d'administration
  - ☐ Requête de suppression de documents
  - ☐ Requête d'archivage de documents
  - ☐ Requête de modification de documents
  - ☐ Ajout des droits d'administration à un.e utilisateur.rice
  - ☐ Suppression des droits d'administration d'un.e administrateur.rice

## Phase 2

---

La phase 2 n'est pas encore détaillée.

---

## Modifications et apports par rapport à la première version

---

### Note de clarification

- Précision des livrables et des dates butoirs
- Relecture & corrections (orthographe, syntaxe)
- Précision du contexte et des objectifs généraux
- Précision des contraintes technologiques
- Ajout de *users stories* pour préciser les objectifs du projet

### Product backlog

- Description plus détaillée + séparation en tâches plus petites
- Relecture & corrections (orthographe, syntaxe, correspondance avec le sujet)

## Modifications et apports par rapport à la seconde version

---

### Note de clarification

---

- Correction des erreurs syntaxiques

### Product backlog

---

## Ressources utilisées pour la correction de la version 2

---

- Note de clarification NA17, groupe 8 (Vincent Lainé, Valentin Le Gauche, Haozhou Zhang, Marion Coisnard)
- Product Backlog NA17, groupe 8 (Vincent Lainé, Valentin Le Gauche, Haozhou Zhang, Marion Coisnard)
- Note de clarification et product backlog NA17, groupe 2 (Yuwei Luo, Amaury Gelin, Waël Hamdan et Claire Guyot)
- Note de clarification NA17, groupe 1 (Romain Fayolle, Pierre Louis Allain, Martin Dizier, Huang Xiaoyan)

