

# CAHIER DES CLAUSES TECHNIQUES PARTICULIERES (CCTP)

ACCORD-CADRE DE SERVICE

**2025-048**

Mise en oeuvre d'un service d'hébergement pour la  
plateforme d'enseignement CoursEnLigne

## Table des matières

1. Contexte .....	3
2. Existant .....	3
3. Objectifs.....	3
4. Expression des besoins .....	4
4.1. Architecture existante .....	5
4.2. Architecture cible .....	5
4.2.1. Éléments constitutifs du projet .....	5
4.2.2. Configuration .....	6
4.3. Interface utilisateur .....	7
4.3.1. Accès et authentification .....	7
4.4. Politique de sauvegarde .....	7
4.5. Disponibilité et objectifs de niveau de service .....	7
5. Prestations attendues et unités d'oeuvre .....	8
5.1. Prestations UO1 : mise en oeuvre de la solution.....	8
5.2. Prestation UO 2 : évolution de la plateforme.....	8

# 1. Contexte

L'Université Paris Nanterre propose à ses usagers la plateforme d'enseignement « CoursEnLigne » (appelée ci-dessous « CEL »). Celle-ci accueille les formations dans le cadre de dispositifs, permet de dispenser les formations à distance et accompagne et complète les enseignements présentiels. Plus de 6 000 cours y sont diffusés chaque année par 2 000 enseignants à 30 000 étudiants.

L'université souhaite être accompagnée dans la mise en œuvre technique d'un service d'hébergement de la plate-forme et les logiciels qui y sont associés.

# 2. Existant

La plateforme CEL est constituée du système d'apprentissage à code source ouvert Moodle et des logiciels et outils annexes. Ces outils permettent l'intégration de Moodle dans le système d'information de l'université et complètent les fonctionnalités de Moodle afin de répondre aux besoins des usagers étudiants, enseignants et personnels.

L'ensemble de ces logiciels fonctionne sur un socle GNU/Linux, serveur HTTP, base de données et moteurs d'exécution PHP, Python et NodeJS.

# 3. Objectifs

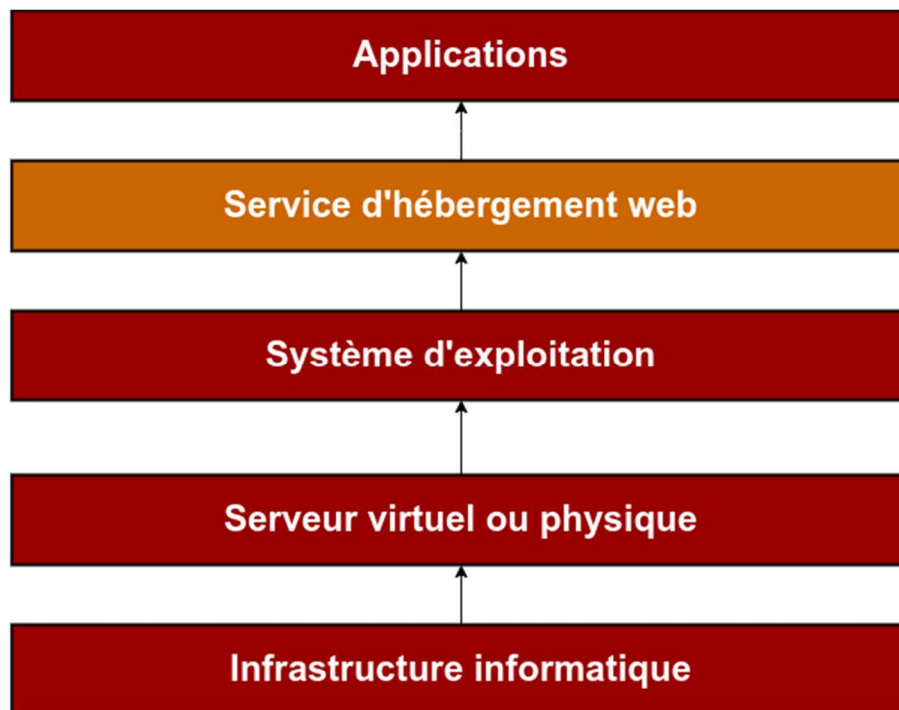
L'Université Paris Nanterre a pour objectif d'assurer la pérennité et l'évolution de l'hébergement web mis en place actuellement et souhaite déployer une solution répondant aux exigences suivantes :

- basée sur des logiciels à code source ouvert dont les équipes en interne ont les compétences opérationnelles
- s'intègre dans l'infrastructure numérique interne de l'université
- respectueuse des standards et règles de l'art du métier
- répondante aux exigences de sécurité, performance et disponibilité
- permette l'autonomie et l'agilité des équipes internes dans l'exploitation

Le projet consiste à mettre en place un service d'hébergement de sites web installé sur l'infrastructure serveurs et réseau de l'université.

Les objectifs principaux du projet sont :

- mettre en œuvre un service d'hébergement sites web qui répond au besoin actuel
- assurer la maintenance fonctionnelle, la mise à jour son l'évolution
- mettre en œuvre des processus et moyens d'exploitation en autonomie
- proposer un support de qualité permettant d'assurer la pérennité de la solution



## 4. Expression des besoins

La solution actuellement en production permet d'héberger :

- 8 plateformes Moodle en production, 4 plateformes de test et développement
- 3 applications Drupal en production, 2 de test et développement
- 11 applications PHP CodeIgniter (version 3 et 4)
- 3 applications PHP Symfony (version 5.4 et 6.4)
- 2 applications Vue JS
- des outils de déploiement, gestion et monitoring écrits en Bash, Python et PHP

Pour l'année universitaire 2023 (septembre 2023 - juin 2024) la plateforme CEL a été utilisée par 32 980 usagers actifs dont 31028 étudiants et 1952 enseignants.

Nous avons enregistré :

- 4 117 106 visites
- 18 105 756 pages vues, 13 215 549 pages vues uniques
- 648 451 téléchargements, 539 420 téléchargements uniques
- temps moyen de génération de page 1,21s.

Les caractéristiques du serveur qui héberge la plateforme sont:

- 32 GB de RAM
- 12 coeurs
- 6 TO d'espace disque

La plateforme CEL occupe 1,2 TO d'espace disque.

## 4.1. Architecture existante

Les technologies suivantes sont actuellement mises en place sur nos serveurs d'hébergement web :

- Ubuntu 22.04 LTS
- Serveur HTTP : NGINX 1.18
- Serveur de base de données : MariaDB 10.6.18
- Serveur messagerie postfix (fonctionne en mode relai)
- PHP-FPM 8.1.2
- Certificats TLS (Let's Encrypt)
- Python 3.10.12
- NodeJS 20.18.1
- Openjdk 11.0.26

## 4.2. Architecture cible

La solution doit permettre l'hébergement et la gestion de sites web par projet.

### 4.2.1. Éléments constitutifs du projet

Chaque projet est constitué de :

- espace de stockage public
- espace de stockage privé
- espace de stockage d'exécution (dossiers temporaires)
- espace de sauvegarde d'urgence
- fichiers de configuration
- fichiers de journaux

Dans l'espace de stockage public sont déposés les fichiers servis par le serveur web et accessibles à l'extérieur.

L'espace de stockage privé est utilisé par l'application pour stocker des fichiers nécessaires au fonctionnement, mais ces fichiers ne sont pas accessibles à l'extérieur.

Le stockage temporaire est utilisé par le projet pour les fichiers de lecture et écriture temporaires utilisés lors de l'exécution.

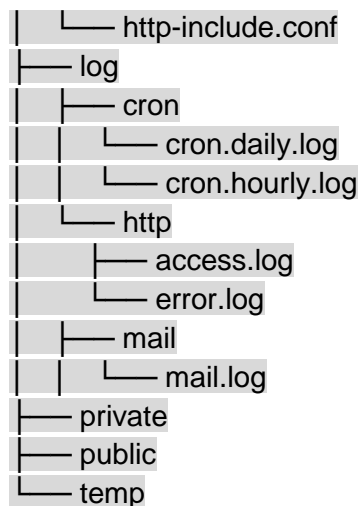
L'espace de sauvegarde d'urgence contient une sauvegarde du projet et les fichiers annexes (configuration, base de données) permettant une restauration d'urgence du projet en cas d'incident d'exploitation ou mise à jour.

Les fichiers de configuration constituent l'ensemble des variables et propriétés du projet comme le nom du projet, l'état, l'environnement d'exécution, les tâches programmées, l'URL publique, etc.

Les fichiers de journaux sont l'ensemble de journaux d'accès, erreurs et notifications.

Exemple de l'arbre des fichiers d'un projet type:

```
.
├── backup
├── conf
│   ├── config.yml
│   └── php-include.ini
```



La taille totale de l'espace de stockage par projet est variable selon les besoins du projet.

## 4.2.2. Configuration

La configuration de chaque projet est définie par les propriétés suivantes:

- identificateur unique
- état
- accès
- environnement d'exécution

Les propriétés sont renseignées dans le fichier de configuration principal.

### État

Les états d'un projet sont :

- actif: le projet est en production
- inactif : le projet n'est pas exécuté, les tâches programmées sont désactivées, mais le projet est fonctionnel, l'ensemble de la configuration et données sont en place, avec la possibilité de l'activer si besoin
- archivé : le projet n'est pas exécuté, les données et l'ensemble de journaux sont consultables (remontée de données pour vérifications et preuve)

### Accès

La propriété accès détermine l'accès au projet. Trois valeurs possibles:

- public: accessible sur Internet
- restreint: limité à une liste de plages ou adresses IP
- auth: (mécanisme HTTP auth)

### Environnement d'exécution

L'environnement d'exécution est une combinaison de versions de technologies qui permettent de faire fonctionner des différentes versions d'applications. Plusieurs environnements doivent être proposés par le service d'hébergement.

Par exemple:

- plateforme Moodle 3.9 requiert PHP 7.4
- plateforme Moodle 4.1 requiert PHP 8.1 ou 8.3
- plateforme Moodle 5.0 requiert PHP 8.3

Exemple de fichier de configuration type:

```
name: moodle-test
php_version: '8.3'
running_state: on
env: test
database:
  type: mariadb
  version: '10.6'
  host:
  name: moodle-test
  user: moodle-test
  pass: XXXX
```

### 4.3. Interface utilisateur

Une interface utilisateur de ligne de commande (CLI) est fournie par le soumissionnaire. Celle-ci doit permettre la création, mise à jour et suppression des projets web. Ces commandes doivent permettre la gestion des projets web en autonomie.

À partir des modifications du fichier de configuration, la commande de mise à jour permet d'adapter la configuration des différents services utilisés par le projet, comme par exemple la configuration NGINX et PHP.

#### 4.3.1. Accès et authentification

L'authentification et l'accès aux serveurs et réseau est géré par la DRI.

À l'installation d'un nouveau serveur, la DRI fournira un compte sécurisé au prestataire pour exécuter ses missions et si besoin aux utilisateurs qui exploitent les services proposés sur le serveur.

La connexion s'effectuera selon la procédure d'accès pour tous les prestataires de l'Université par la VDI, pas d'accès en direct en SSH .

### 4.4. Politique de sauvegarde

Les serveurs sont sauvegardés par la DRI. La politique de sauvegarde est : quotidiennement en soirée (lun-sam) sur 5 semaines.

### 4.5. Disponibilité et objectifs de niveau de service

Le service CoursEnLigne est de haute importance pour l'établissement. L'objectif de disponibilité, hors les fenêtres de de maintenance planifiées, est de 99% (calculé sur 365 jours/an).

La solution mise en place doit inclure des moyens de surveillance, monitoring et alerte sur les technologies déployées et gérées par le prestataire.

Le prestataire doit assurer une maintenance préventive afin de garantir la sécurité et la disponibilité des services.

Dans le cadre d'une Garantie du Temps de d'Intervention (GTI) et Garantie de Temps de Rétablissement (GTR) les délais sont les suivants:

**Incident critique:** incident affectant la disponibilité de l'ensemble ou de la majorité des services

- GTI 2 heures, GTR 4 heures

**Incident significatif:** incident affectant une partie des services

- GTI 4 heures, GTR 8 heures

**Incident modéré:** incident dégradant le fonctionnement ou la performance des services

- GTI 4 heures, GTR 1 jour ouvrable

**Incident mineur:** demande d'assistance ou conseil d'exploitation

- GTI 1 jour, GTR 3 jours ouvrables

## 5. Prestations attendues et unités d'oeuvre

### 5.1. Prestations UO1 : mise en oeuvre de la solution

Sur l'infrastructure serveur et réseau, mise à disposition par la DRI, installer et configurer l'ensemble des services nécessaires pour mettre en production la plateforme CEL.

Installation et configuration de :

- NGINX
- MariaDB
- PHPFPM et l'ensemble des extensions nécessaires
- Serveur de messagerie (relai)
- Certificats TLS (Let's Encrypt)
- NodeJS
- Python
- OpenJDK

Le prestataire est responsable de:

- la conformité de la configuration des services
- la sécurisation des services installés sur le serveur
- la veille et l'application des dernières mises à jours de sécurité du serveur et les services installés

### 5.2. Prestation UO 2 : évolution de la plateforme

En cours de marché, l'Université se réserve la possibilité de réaliser des prestations hors forfait. Ces prestations hors forfait sont des prestations d'évolutions de la plateforme.

Le titulaire devra répondre aux sollicitations dans un délai de quinze (15) jours par la remise d'un devis et un calendrier de réalisation. Les parties pourront organiser des réunions de cadrage de projet afin de définir ensemble le périmètre, le calendrier et le prix de la prestation.



Le prix de ces prestations sera réalisé sur la base des prix unitaires définie par le titulaire dans la partie hors forfait du bordereau des prix (BP). A l'envoi du bon de commande, le prix de la prestation est forfaitaire.

Il est envisagé de commander les prestations hors forfait suivantes en cours de marché :

- Déploiement de plusieurs environnements d'exécution afin de permettre l'évolution des projets.
- Mise en place de cache Redis.
- Mise en place d'une solution de répartition de charge entre plusieurs serveurs.