

# Guide du Développeur 4

Administration, Statistiques & Documentation



Aya

## Tes responsabilités :

-  Gestion des utilisateurs (CRUD)
-  Paramètres de l'application
-  Module statistiques et rapports
-  Données de test (seed.sql)
-  Documentation complète + UML

## Table des matières

<b>1 Ton rôle dans l'équipe</b>	<b>2</b>
1.1 Fichiers à créer . . . . .	2
<b>2 Tâche 1 : Gestion des Utilisateurs</b>	<b>3</b>
2.1 AdminController.php — Partie Utilisateurs . . . . .	3
2.2 Vues pour la gestion des utilisateurs . . . . .	4
<b>3 Tâche 2 : Gestion des Paramètres</b>	<b>5</b>
3.1 Modèle Parametre.php . . . . .	5
3.2 Page Paramètres (parametres.php) . . . . .	5
<b>4 Tâche 3 : Module Statistiques</b>	<b>6</b>
4.1 StatistiqueController.php . . . . .	6
4.2 Statistiques à calculer . . . . .	6
4.3 Page Statistiques (statistiques.php) . . . . .	7
<b>5 Tâche 4 : Exports Excel et PDF</b>	<b>8</b>
5.1 Export Excel . . . . .	8
5.2 Export PDF . . . . .	8
<b>6 Tâche 5 : Créer les données de test (seed.sql)</b>	<b>9</b>
<b>7 Tâche 6 : Documentation</b>	<b>10</b>
7.1 Diagrammes UML à produire . . . . .	10
7.2 Documentation technique . . . . .	11
7.3 Guide utilisateur . . . . .	12
<b>8 Planning suggéré</b>	<b>13</b>
<b>9 Checklist finale</b>	<b>13</b>
9.1 Code . . . . .	13
9.2 Documentation . . . . .	13

## 1 Ton rôle dans l'équipe

Tu gères les fonctionnalités d'administration réservées au directeur, ainsi que toute la documentation du projet.

### À savoir

#### Particularité de ce rôle :

- Tu fais du code PHP (gestion utilisateurs, stats)
- Tu fais aussi de la documentation (UML, guide utilisateur)
- Tu crées les données de test pour toute l'équipe

### 1.1 Fichiers à créer

app/models/

Parametre.php	→ Gestion des paramètres
LogActivite.php	→ Journal d'activité

app/controllers/

AdminController.php	→ Gestion utilisateurs + paramètres
StatistiqueController.php	→ Stats et rapports

app/views/directeur/

utilisateurs/	
index.php	→ Liste des utilisateurs
create.php	→ Formulaire création
edit.php	→ Formulaire modification
show.php	→ Détail utilisateur
parametres.php	→ Page paramètres
statistiques.php	→ Page statistiques

database/

seed.sql	→ Données de test
----------	-------------------

docs/

documentation-technique.pdf

guide-utilisateur.pdf

uml/

diagramme-cas-utilisation.png

diagramme-classes.png

diagramme-sequence-\*.png

## 2 Tâche 1 : Gestion des Utilisateurs

Le directeur peut créer, modifier et désactiver les comptes utilisateurs.

### 2.1 AdminController.php — Partie Utilisateurs

#### ☰ Tâche

Méthodes à créer :

**1. users()**

- Récupérer la liste de tous les utilisateurs
- Permettre la recherche par nom/email
- Permettre le filtrage par rôle
- Afficher la vue index.php

**2. createUser()**

- Afficher le formulaire de création

**3. storeUser()**

- Vérifier le token CSRF
- Valider les données (nom, prénom, email, rôle)
- Vérifier que l'email n'existe pas déjà
- Générer un mot de passe temporaire
- Créer l'utilisateur en base
- Envoyer un email avec le mot de passe temporaire
- Rediriger avec message de succès

**4. editUser(\$id)**

- Récupérer l'utilisateur
- Afficher le formulaire pré-rempli

**5. updateUser(\$id)**

- Vérifier CSRF
- Mettre à jour les informations
- Rediriger avec message de succès

**6. toggleUser(\$id)**

- Activer ou désactiver le compte (toggle)
- Un compte désactivé ne peut plus se connecter

**7. resetUserPassword(\$id)**

- Générer un nouveau mot de passe temporaire
- L'envoyer par email à l'utilisateur

## 2.2 Vues pour la gestion des utilisateurs

### ☰ Tâche

#### **index.php** — Liste des utilisateurs

- Barre de recherche
- Filtres par rôle (dropdown)
- Tableau avec : Nom, Prénom, Email, Rôle, Statut, Actions
- Badge vert/rouge pour le statut (actif/inactif)
- Boutons : Voir, Modifier, Activer/Désactiver
- Bouton "Nouvel utilisateur"

#### **create.php** — Formulaire création

- Champs : Nom, Prénom, Email, Rôle (select), Téléphone, Bureau
- Le mot de passe sera généré automatiquement

#### **edit.php** — Formulaire modification

- Mêmes champs que création, pré-remplis
- Option pour réinitialiser le mot de passe

#### **show.php** — Détail utilisateur

- Toutes ses informations
- Date de création, dernière connexion
- Historique d'activité (si tu implémentes les logs)

## 3 Tâche 2 : Gestion des Paramètres

### 3.1 Modèle Parametre.php

#### 💡 Concept

Les paramètres sont stockés en base sous forme **clé/valeur**.

Exemple :

- `delai_minimum` → 48 (heures avant une séance)
- `email_notifications` → true
- `annee_universitaire_debut` → 2024-09-01

#### ☰ Tâche

Méthodes à créer :

- `get($cle)` : Récupérer la valeur d'un paramètre
- `set($cle, $valeur)` : Modifier un paramètre
- `getAll()` : Tous les paramètres

### 3.2 Page Paramètres (parametres.php)

#### ☰ Tâche

Paramètres à pouvoir configurer :

- **Délais**
  - Délai minimum avant une séance pour soumettre (ex : 48h)
  - Durée de blocage après tentatives échouées (ex : 15 min)
- **Notifications**
  - Activer/désactiver les emails automatiques
- **Année universitaire**
  - Date de début
  - Date de fin
- **Motifs prédefinis**
  - Liste de motifs types (Conférence, Maladie, etc.)
  - Pouvoir en ajouter/supprimer

## 4 Tâche 3 : Module Statistiques

### 4.1 StatistiqueController.php

#### ☰ Tâche

Méthodes à créer :

**1. index()**

- Afficher la page statistiques avec tous les graphiques

**2. getData()**

- Retourner les données en JSON pour les graphiques (AJAX)
- Filtrable par période (ce mois, ce semestre, cette année)

**3. exportExcel()**

- Générer un fichier Excel avec les demandes
- Filtrable par période et statut

**4. exportPDF()**

- Générer un rapport PDF
- Inclure les graphiques et statistiques clés

### 4.2 Statistiques à calculer

#### ☰ Tâche

Requêtes SQL à écrire :

- Nombre de demandes par mois
- Nombre de demandes par statut
- Taux d'approbation (approuvées / total soumises)
- Temps moyen de traitement (soumission → approbation)
- Top 5 des professeurs avec le plus de demandes
- Répartition par type (changement vs annulation)

Concepts SQL utiles :

- COUNT(\*) pour compter
- GROUP BY pour regrouper par mois/statut
- AVG() pour les moyennes
- DATEDIFF() pour calculer les durées

### 4.3 Page Statistiques (statistiques.php)

#### ☰ Tâche

##### Éléments à afficher :

- Sélecteur de période (ce mois, ce semestre, cette année, personnalisé)
- Graphique en barres : Demandes par mois
- Graphique camembert : Répartition par statut
- Tableau : Top professeurs demandeurs
- Indicateurs clés : Total, taux approbation, temps moyen
- Boutons export : Excel, PDF

Utilise Chart.js pour les graphiques (comme Dev 3).

## 5 Tâche 4 : Exports Excel et PDF

### 5.1 Export Excel

#### 💡 Concept

Pour générer des fichiers Excel en PHP, utilise la librairie **PhpSpreadsheet**.

Installation : `composer require phpoffice/phpspreadsheet`

Recherche : "PhpSpreadsheet tutorial", "PHP export Excel"

#### ☰ Tâche

##### Contenu du fichier Excel :

- Une ligne par demande
- Colonnes : Numéro, Professeur, Type, Date séance, Statut, Date soumission, Date traitement
- En-têtes en gras
- Filtres sur les colonnes

### 5.2 Export PDF

#### 💡 Concept

Pour générer des PDF, utilise **TCPDF** ou **Dompdf**.

**Dompdf** est plus simple : tu écris du HTML et il le convertit en PDF.

Installation : `composer require dompdf/dompdf`

#### ☰ Tâche

##### Contenu du rapport PDF :

- Titre : "Rapport des demandes - [Période]"
- Logo de l'établissement
- Résumé statistique (totaux, taux)
- Tableau des demandes
- Graphiques (en images)
- Date de génération

## 6 Tâche 5 : Créer les données de test (seed.sql)

### ⚠️ Important

Ce fichier est **très important** ! Il permet à toute l'équipe de tester l'application avec des données réalistes.

### ☰ Tâche

#### Données à insérer :

##### Utilisateurs (13 minimum)

- 10 professeurs avec des noms réalistes
- 2 assistantes
- 1 directeur
- Mots de passe : tous "password123" (hashés avec bcrypt !)

##### Matières (5-8)

- Exemples : Algorithmique, Base de données, Réseaux, Sécurité, etc.
- Avec codes (ALG101, BDD201, etc.)

##### Salles (10)

- Exemples : Amphi A, Salle 101, Labo Info 1, etc.
- Avec capacités variées (30, 50, 100, 200)

##### Séances (50 minimum)

- Réparties sur plusieurs semaines
- Pour différents professeurs
- Différents types (cours, TD, TP)

##### Demandes (30 minimum)

- Dans tous les statuts (brouillon, en\_attente, validée, approuvée, rejetée)
- Les deux types (changement et annulation)
- Avec des motifs réalistes

##### Validations

- Historique pour les demandes validées/approuvées/rejetées

##### Notifications

- Quelques notifications pour tester l'affichage

### ℹ️ À savoir

#### Astuce pour les mots de passe :

Génère le hash bcrypt de "password123" une fois avec PHP, puis utilise ce hash pour tous les utilisateurs dans le SQL.

## 7 Tâche 6 : Documentation

C'est une grosse partie de ton travail ! La documentation est un livrable noté.

### 7.1 Diagrammes UML à produire

#### ☰ Tâche

##### 1. Diagramme de cas d'utilisation

- Les 3 acteurs : Professeur, Assistante, Directeur
- Tous les cas d'utilisation (créer demande, valider, approuver, etc.)
- Les relations entre acteurs et cas d'utilisation

##### 2. Diagramme de classes

- Toutes les classes du projet (User, Demande, Seance, etc.)
- Les attributs et méthodes principales
- Les relations (association, composition, héritage)
- Les cardinalités (1..\*, 0..1, etc.)

##### 3. Diagrammes de séquence (3-4)

- Scénario 1 : Professeur crée et soumet une demande
- Scénario 2 : Assistante valide une demande
- Scénario 3 : Directeur approuve une demande
- Scénario 4 : Connexion d'un utilisateur

##### 4. Diagramme d'états (optionnel mais recommandé)

- Cycle de vie d'une demande
- États : brouillon, en\_attente, validée, approuvée, rejetée, annulée
- Transitions entre les états

#### ⓘ À savoir

##### Outils pour créer les diagrammes :

- **draw.io** (gratuit, en ligne) — <https://draw.io>
- **StarUML** (logiciel)
- **Lucidchart** (en ligne)
- **PlantUML** (si tu aimes coder les diagrammes)

## 7.2 Documentation technique

### ☰ Tâche

**Contenu du document :**

#### **1. Introduction**

- Contexte et objectifs du projet
- Technologies utilisées

#### **2. Architecture**

- Structure MVC expliquée
- Organisation des dossiers
- Schéma de l'architecture

#### **3. Base de données**

- Schéma relationnel (image)
- Description de chaque table
- Relations entre les tables

#### **4. Modules fonctionnels**

- Authentification
- Gestion des demandes
- Planning
- Administration
- Notifications

#### **5. Sécurité**

- Mesures implémentées
- Protection CSRF, XSS, SQL Injection, etc.

#### **6. Installation**

- Prérequis
- Étapes d'installation
- Configuration

## 7.3 Guide utilisateur

### ☰ Tâche

**Un guide pour chaque rôle :**

#### **Guide Professeur**

- Comment se connecter
- Comment créer une demande (avec captures d'écran)
- Comment suivre ses demandes
- Comment modifier son profil

#### **Guide Assistante**

- Comment traiter les demandes
- Comment utiliser le planning
- Comment valider ou rejeter

#### **Guide Directeur**

- Comment approuver les demandes
- Comment gérer les utilisateurs
- Comment consulter les statistiques
- Comment configurer les paramètres

**Conseil :** Fais beaucoup de captures d'écran avec des flèches/annotations !

## 8 Planning suggéré

---

### Tâches

---

Fichier seed.sql (données de test)  
Diagramme de cas d'utilisation  
Diagramme de classes  
Diagrammes de séquence  
AdminController (gestion utilisateurs)  
Vues gestion utilisateurs  
Page paramètres  
Module statistiques  
Exports Excel/PDF  
Documentation technique  
Guide utilisateur

---

## 9 Checklist finale

### 9.1 Code

Le directeur peut voir la liste des utilisateurs  
Le directeur peut créer un utilisateur  
Le directeur peut modifier un utilisateur  
Le directeur peut activer/désactiver un compte  
La page paramètres fonctionne  
Les statistiques s'affichent avec graphiques  
L'export Excel génère un fichier correct  
L'export PDF génère un rapport lisible  
Le fichier seed.sql s'exécute sans erreur

### 9.2 Documentation

Diagramme de cas d'utilisation complet  
Diagramme de classes complet  
3-4 diagrammes de séquence  
Documentation technique rédigée  
Guide utilisateur avec captures d'écran

---

**Bon courage Aya ! 🍀**

Tu gères l'admin et la doc — deux rôles importants !