

Cahier des charges – Site Universitaire

1. Introduction

1.1 Contexte du projet

Dans le cadre de l'amélioration de la communication et de la gestion des activités au sein de l'université, il est nécessaire de développer un site web permettant :

- La gestion des **fêtes et activités de fin d'année**, incluant les événements sportifs, culturels et sociaux.
- La consultation des **résultats académiques** des étudiants de manière sécurisée et rapide.

Actuellement, ces informations sont soit dispersées, soit gérées manuellement, ce qui entraîne des difficultés de suivi, de transparence et de consultation. Ce site vise à centraliser toutes ces fonctionnalités et à les rendre accessibles aux étudiants et à l'administration tout en conservant un historique annuel.

1.2 Objectifs généraux

Le site a pour principaux objectifs :

1. Permettre à l'administration de gérer facilement les **activités et événements** de l'université chaque année.
2. Permettre aux étudiants de **consulter et participer** aux activités et événements.
3. Permettre à l'administration de **saisir et publier les résultats académiques** par matière et semestre.
4. Permettre aux étudiants de **consulter leurs résultats** de manière sécurisée et historique.
5. Offrir une interface **simple, responsive et réutilisable** pour toutes les années universitaires.

1.3 Public cible

- **Étudiants :**
 - Accéder aux informations des activités et événements.
 - S'inscrire et suivre leur participation.
 - Consulter leurs résultats académiques actuels et historiques.
- **Administration / Personnel académique :**

- Gérer les activités, événements et conditions de participation.
- Saisir et publier les résultats académiques.
- Consulter les rapports et statistiques par activité, classe ou promotion.
- **Autres parties prenantes** (optionnel) :
 - Encadrants, professeurs ou responsables de clubs pouvant suivre les activités et évaluations.

2. Besoins et Exigences

2.1 Besoins fonctionnels

Les besoins fonctionnels décrivent **ce que le système doit faire** pour répondre aux attentes des utilisateurs.

Module Fêtes & activités

- L'administration peut :
 - Ajouter, modifier et supprimer des activités et événements (football, danse, musique, théâtre, sorties, soirées...).
 - Définir les conditions de participation pour chaque activité ou événement.
 - Gérer les tickets et prix de participation.
 - Suivre les inscriptions des étudiants et générer des rapports détaillés.
 - Consulter l'historique des activités des années précédentes.
 - Ajouter des photos ou vidéos des événements passés.
- Les étudiants peuvent :
 - Voir les activités et événements disponibles pour l'année en cours.
 - S'inscrire aux activités et événements.
 - Acheter ou réserver des tickets.
 - Suivre leurs participations.
 - Consulter l'historique des activités des années précédentes.
 - Laisser des commentaires ou avis sur les activités passées.

Module Résultats académiques

- L'administration peut :
 - Saisir ou modifier les notes des étudiants par matière et par semestre.
 - Importer des résultats depuis un fichier CSV.
 - Consulter des rapports et statistiques globales (par classe, semestre, promotion).
 - Gérer l'historique des années précédentes et exporter les données.
- Les étudiants peuvent :
 - Se connecter avec un identifiant unique (matricule + mot de passe).
 - Consulter les résultats par semestre et année.
 - Télécharger ou imprimer leurs relevés de notes.

- Accéder à l'historique complet des années précédentes.
- Visualiser des graphiques simples de l'évolution de leurs notes.
- Recevoir des alertes ou notifications lors de la publication des résultats.
- Calcul automatique de la moyenne générale par semestre et par année.

Fonctionnalités globales

- Interface utilisateur simple et responsive (mobile et PC).
- Gestion multi-niveaux pour l'administration : super-admin, admin activités, admin résultats.
- Recherche rapide pour trouver activités, événements ou étudiants.
- Notifications et alertes pour inscriptions et résultats.
- Export des données historiques pour archives (CSV/PDF).
- Profil étudiant complet (photo, promotion, activités suivies, résultats, tickets).
- Gamification optionnelle : points et classement pour les activités.

2.2 Besoins non fonctionnels

Les besoins non fonctionnels concernent **la qualité et les contraintes du système**.

- **Performance :**
 - Le site doit être rapide et réactif, même avec un grand nombre d'étudiants et d'événements.
- **Sécurité :**
 - Authentification obligatoire pour étudiants et administration.
 - Protection des données personnelles et académiques.
 - Sauvegarde régulière de la base de données.
- **Fiabilité :**
 - Le système doit fonctionner de manière continue et stable.
 - Possibilité de restaurer les données en cas de problème.
- **Maintenabilité :**
 - Code clair et commenté pour faciliter les mises à jour.
 - Base de données structurée pour ajouter de nouvelles fonctionnalités facilement.
- **Portabilité :**
 - Le site doit être accessible sur différents navigateurs (Chrome, Firefox, Edge) et sur mobile.
- **Extensibilité :**
 - Le système doit pouvoir intégrer de nouvelles fonctionnalités (ex : paiement en ligne, nouvelles activités) sans modifier la structure principale.
- **Ergonomie et interface utilisateur :**
 - Interface intuitive et facile à utiliser pour les étudiants et l'administration.
 - Design responsive adapté aux écrans de différentes tailles.

4. Contraintes

4.1 Contraintes techniques

- **Technologie de développement** : le site sera développé en **PHP natif**, avec HTML, CSS et JavaScript pour le front-end.
- **Base de données** : utilisation de **MySQL** pour stocker toutes les informations (étudiants, activités, résultats, inscriptions).
- **Responsive design** : le site doit être accessible sur **ordinateurs, tablettes et smartphones**.
- **Sécurité** :
 - Authentification obligatoire pour étudiants et administrateurs.
 - Protection des données personnelles et académiques (mots de passe hachés, validation côté serveur).
- **Maintenance et évolutivité** : le code doit être **structuré et commenté** pour faciliter les mises à jour futures.
- **Compatibilité navigateur** : le site doit fonctionner sur **Chrome, Firefox, Edge et Safari**.
- **Gestion de l'historique** : toutes les activités et résultats doivent être liés à l'année universitaire pour permettre un suivi multi-années.
- **Export de données** : possibilité d'exporter les résultats et inscriptions en **CSV ou PDF**.

4.2 Contraintes temporelles

- **Durée de développement** : le projet doit être livré dans un délai défini, par exemple **3 à 4 mois** pour la version initiale.
- **Mises à jour annuelles** : au début de chaque année universitaire, l'administration doit pouvoir ajouter les nouvelles activités et résultats.
- **Disponibilité** : le site doit être opérationnel **pendant toute l'année universitaire** sans interruption prolongée.
- **Phases du projet** :
 - Analyse et conception : 2 à 3 semaines
 - Développement des modules de base : 6 à 8 semaines
 - Tests et validation : 2 à 3 semaines
 - Mise en production et formation : 1 semaine

4.3 Contraintes juridiques et réglementaires

- **Protection des données personnelles** : conformité aux règles locales sur la protection des données des étudiants (ex : RGPD ou législation nationale sur les données personnelles).
- **Propriété intellectuelle** : le contenu du site (photos, vidéos, documents) doit respecter les droits d'auteur et licences applicables.

- **Sécurité des informations académiques** : les résultats et informations sensibles doivent être **strictement accessibles** aux étudiants concernés et au personnel autorisé.
- **Archivage légal** : les résultats et inscriptions doivent pouvoir être conservés pendant la durée légale ou réglementaire définie par l'université.
- **Accès et confidentialité** : seuls les utilisateurs autorisés peuvent consulter ou modifier les données selon leur rôle (admin vs étudiant).

5. Environnement Technique

5.1 Langages et technologies utilisées

Le projet sera entièrement développé en **PHP natif**, avec un environnement web complet basé sur les standards actuels.

Back-end

- **Langage principal** : PHP (version ≥ 8.0)
- **Serveur web** : Apache (inclus dans WampServer ou XAMPP)
- **Base de données** : MySQL (ou MariaDB selon le serveur local)
- **Connexion BD** : PDO (PHP Data Objects) pour sécuriser les échanges et éviter les injections SQL.

Front-end

- **HTML5 / CSS3** : structure et présentation des pages.
- **JavaScript (ES6+)** : interactions dynamiques et validation côté client.
- **Framework CSS** : Bootstrap 5 ou Tailwind CSS (pour le design responsive et moderne).
- **Bibliothèque de graphiques** : Chart.js pour les statistiques et graphiques de résultats.

Autres outils

- **Export de fichiers** : génération de rapports en **CSV ou PDF** avec les bibliothèques PHP natives (fputcsv, FPDF ou Dompdf).
- **Gestion des sessions** : PHP Sessions pour l'authentification et la gestion des rôles.
- **Sécurité** :
 - Hachage des mots de passe avec password_hash().
 - Validation et filtrage des entrées utilisateur.
 - Protection contre les failles XSS et CSRF.

5.2 Outils de développement

- **Environnement local** : WampServer ou XAMPP.
- **Éditeur de code** : Visual Studio Code (ou Sublime Text).
- **Système de gestion de versions** : Git / GitHub pour le suivi du code et le travail en équipe.
- **Navigateur recommandé pour les tests** : Google Chrome, Mozilla Firefox.

5.3 Hébergement et déploiement

- **Serveur de production** : hébergement mutualisé ou dédié compatible PHP/MySQL.
- **Nom de domaine** : défini par l'université (ex. : activites.universite-bj.com).
- **Déploiement** : transfert FTP ou via GitHub Actions / cPanel.
- **Sauvegarde automatique** : export régulier de la base de données pour éviter toute perte de données.

5.4 Configuration minimale requise

Côté serveur

- PHP ≥ 8.0
- MySQL ≥ 5.7
- Apache ≥ 2.4
- Extensions PHP activées : pdo_mysql, mbstring, json, openssl, gd, curl.

Côté client

- Navigateur moderne (Chrome, Firefox, Edge, Safari).
- Connexion Internet stable (au moins 1 Mbps).
- Résolution minimale recommandée : **1366×768 px**.

5.5 Environnement de test

- Base de données de test distincte pour les essais.
- Comptes de test pour les différents rôles (étudiant, admin, super-admin).
- Jeux de données simulés : activités, événements, étudiants, résultats.
- Tests effectués sur différents navigateurs et appareils (PC, mobile, tablette).

6. Livrables et Planning

6.1 Livrables

Les livrables correspondent à tous les éléments concrets remis à l'université ou à l'équipe projet à différentes étapes.

Livrables fonctionnels

1. **Site web complet** avec :
 - o Module **Fêtes & activités** (gestion des activités, événements, inscriptions, tickets, historique et galerie photo/vidéo).
 - o Module **Résultats académiques** (saisie, consultation, export, statistiques, historiques).
2. **Interface utilisateur responsive** (PC, tablette, mobile).
3. **Système de gestion multi-niveau** pour l'administration : super-admin, admin activités, admin résultats.
4. **Système d'authentification sécurisé** pour étudiants et administrateurs.
5. **Base de données MySQL** avec toutes les tables et relations.
6. **Export de données** en CSV ou PDF (résultats et inscriptions).
7. **Profil étudiant complet** : photo, activités, tickets, résultats.
8. **Graphiques statistiques** des résultats académiques.
9. **Historique annuel** pour toutes les activités et résultats.
10. **Documentation technique et utilisateur** :
 - o Guide d'installation et configuration.
 - o Guide utilisateur pour l'administration et les étudiants.
 - o Schéma de la base de données (MCD et MLD).

Livrables annexes

- Comptes de test pour étudiants et administrateurs.
- Jeux de données simulés pour tests (activités, événements, étudiants, résultats).
- Rapports de tests fonctionnels et de sécurité.

6.2 Planning prévisionnel

Le planning est basé sur une **durée totale de 3 à 4 mois** pour la version initiale du projet.

Phase	Durée estimée	Description
1. Analyse et conception	2 à 3 semaines	Définition des besoins, cahier des charges, schéma base de données, maquettes des pages.
2. Développement du module Fêtes & activités	3 à 4 semaines	Création des pages PHP, formulaires, inscriptions, gestion des tickets et historique.
3. Développement du module Résultats académiques	3 à 4 semaines	Saisie et consultation des résultats, import CSV, statistiques, graphiques et historique.
4. Développement des fonctionnalités globales	1 à 2 semaines	Authentification, profils étudiants, multi-niveau admin, notifications, export CSV/PDF.
5. Tests et validation	2 à 3 semaines	Tests fonctionnels, tests de sécurité, corrections des bugs et ajustements.
6. Mise en production et formation	1 semaine	Déploiement sur serveur réel, formation des utilisateurs (administration et étudiants).

7. Risques et Solutions

7.1 Risques identifiés

Lors du développement et de l'exploitation du site universitaire, plusieurs risques peuvent survenir :

Risques techniques

- Pannes serveur ou base de données** : le site pourrait devenir indisponible si le serveur ou la base tombe en panne.
- Failles de sécurité** : risque d'accès non autorisé aux données sensibles des étudiants et de l'administration.
- Incompatibilités navigateurs** : certaines fonctionnalités pourraient ne pas fonctionner correctement sur tous les navigateurs ou appareils mobiles.
- Problèmes de performance** : le site pourrait devenir lent si le nombre d'étudiants ou d'événements augmente fortement.

Risques organisationnels

- Retard dans la saisie des données** : l'administration pourrait tarder à entrer les activités ou les résultats, retardant l'accès aux informations.

6. **Mauvaise utilisation du système** : erreurs de saisie, oubli ou suppression accidentelle de données importantes.

Risques humains

7. **Disponibilité des développeurs** : le retard des équipes de développement peut affecter le respect du planning.
8. **Résistance au changement** : certains utilisateurs peuvent être réticents à utiliser le nouveau système.

7.2 Plan d'atténuation

Pour réduire l'impact des risques identifiés, les mesures suivantes seront mises en place :

Mesures techniques

- **Sauvegarde régulière de la base de données** : automatisée pour garantir la récupération des données.
- **Sécurité renforcée** : utilisation de mots de passe hachés, validation côté serveur, protection contre les failles XSS/CSRF et gestion des rôles d'accès.
- **Tests multiplateformes** : vérifier la compatibilité du site sur différents navigateurs et appareils mobiles.
- **Optimisation des performances** : indexation des tables, requêtes SQL optimisées, mise en cache si nécessaire.

Mesures organisationnelles

- **Planification stricte des saisies** : définir un calendrier annuel pour l'entrée des activités et résultats.
- **Formation des utilisateurs** : sessions pour l'administration et les étudiants pour assurer une bonne utilisation du système.
- **Validation et double contrôle des données** : l'administration vérifiera les informations saisies avant publication.

Mesures humaines

- **Suivi du planning de développement** : mise en place de jalons hebdomadaires et suivi par le chef de projet.
- **Sensibilisation des utilisateurs** : communiquer les avantages du site et fournir un guide utilisateur clair pour encourager son adoption.

8. Annexes

8.1 Glossaire

Terme	Définition
Administrateur (Admin)	Personne responsable de la gestion des activités, événements et résultats sur le site. Il peut être super-admin ou admin spécialisé (activités / résultats).
Étudiant	Utilisateur principal du site qui peut consulter ses résultats, s'inscrire aux activités et participer aux événements.
Activité	Toute activité organisée par l'université : football, danse, musique, théâtre, club d'anglais, etc.
Événement	Evénements spéciaux liés aux activités : sorties, soirées, concerts, compétitions.
Ticket / Inscription	Preuve de participation à une activité ou un événement, pouvant être payant ou gratuit.
Résultats académiques	Notes et relevés de chaque étudiant par matière et semestre.
Année universitaire	Période scolaire définie par l'université, généralement de septembre à juin.
Historique	Ensemble des données passées (activités, événements, résultats) accessibles pour consultation.
Responsive Design	Technique permettant au site de s'adapter automatiquement à différents écrans (PC, tablette, mobile).
CSV	Format de fichier texte contenant des données séparées par des virgules, utilisé pour importer ou exporter des données.
PDF	Format de fichier permettant de partager des documents de manière fiable et imprimable.
Gamification	Utilisation de points, classements ou récompenses pour encourager la participation des étudiants.
MCD (Modèle Conceptuel de Données)	Schéma représentant les entités et leurs relations dans la base de données.
MLD (Modèle Logique de Données)	Traduction du MCD en tables et relations dans la base de données relationnelle.

8.2 Références

- Documentation officielle PHP : <https://www.php.net/manual/fr/>
- Documentation MySQL : <https://dev.mysql.com/doc/>
- Documentation Bootstrap : <https://getbootstrap.com/docs/5.0/getting-started/introduction/>
- Documentation Tailwind CSS : <https://tailwindcss.com/docs>
- Documentation Chart.js : <https://www.chartjs.org/docs/latest/>
- Guide RGPD / protection des données personnelles : <https://gdpr-info.eu/>