## Cahier des Charges : Département Finance

#### **1. Introduction**

**Contexte du projet :** Le département Finance a pour mission de garantir une gestion rigoureuse des paiements, des facturations et des interactions avec les autres départements (Training, Marketing, Scholar, Family). Le système de gestion des paiements doit être automatisé, sécurisé et fluide afin d'assurer une expérience utilisateur optimale et d'intégrer efficacement les processus financiers avec les autres départements.

**Objectifs globaux du département :** Créer un système intégré de gestion des paiements et de facturation qui offre une fluidité maximale dans le processus de paiement des formations, assure une gestion rigoureuse des transactions, et facilite les interactions avec les départements Training, Marketing, Scholar, et Family.

#### **2. Description du projet**

**Fonctionnalités principales à développer :**

* **Intégration des méthodes de paiement en ligne :** Mettre en place une infrastructure de paiement sécurisée et efficace, permettant l'achat de formations et l'automatisation des processus post-paiement (génération de codes d'accès).
* **Système de facturation :** Automatiser la création et la gestion des factures pour chaque transaction.
* **Gestion des interactions avec les autres départements :** Assurer la transmission des informations financières et la coordination avec les départements Training, Marketing, Scholar, et Familly.

**Cas d'utilisation spécifiques :**

1. **Un utilisateur effectue un paiement pour une formation :** Le paiement est validé par une passerelle de paiement et un code d'accès à la formation est généré et envoyé à l'utilisateur.
2. **Création d'une facture après paiement :** Le système génère automatiquement une facture conforme aux normes fiscales et l'envoie à l'utilisateur par e-mail.
3. **Consultation des paiements et factures :** L'utilisateur peut consulter l'historique de ses paiements et télécharger ses factures via un tableau de bord.
4. **Coordination avec le département Training :** Le système permet d'envoyer des informations sur les paiements validés au département Training pour la gestion de l'accès aux formations.

#### **3. Exigences fonctionnelles**

**Liste détaillée des fonctionnalités :**

* **Sélection et achat de formations :**
  + L'utilisateur choisit une formation depuis le site web et initie l'achat.
  + L'interface doit présenter des informations claires sur les formations et le montant à payer.
* **Passerelles de paiement :**
  + Intégration avec des passerelles comme Stripe, PayPal ou Payoneer pour accepter les paiements par carte bancaire, PayPal, etc.
  + Le système doit valider le paiement en temps réel et enregistrer les transactions.
* **Validation et génération de codes d'accès :**
  + Une fois le paiement validé, un code d'accès unique est généré et envoyé à l'utilisateur par e-mail.
  + Le code est également disponible dans le tableau de bord de l'utilisateur.
* **Système de facturation :**
  + Génération automatique de factures après chaque paiement réussi.
  + Les factures doivent être envoyées par e-mail et stockées dans la base de données sécurisée de l'entreprise.
* **Consultation de l'historique des paiements :**
  + Les utilisateurs doivent pouvoir consulter leurs paiements passés via un tableau de bord dans leur compte.
  + Les paiements doivent être consultables par date, montant et formation/service.
* **Suivi des interactions avec les départements :**
  + Le système doit synchroniser les paiements validés avec les autres départements, notamment le département Training pour l'accès aux formations et le département Marketing pour la gestion des promotions.

#### **4. Exigences techniques**

**Technologies à utiliser :**

* **Frontend :** React.js (ou framework similaire pour l’interface utilisateur).
* **Backend :** FastAPI (pour les API de gestion des paiements, des factures, et des interactions).
* **Bases de données :** MySQL (ou équivalent) pour la gestion des paiements, des utilisateurs et des factures.
* **Passerelles de paiement :** Stripe, PayPal, Payoneer (et/ou équivalent) (APIs pour intégrer ces services de paiement).
* **Sécurisation :** SSL/TLS pour sécuriser les transactions et protéger les données des utilisateurs.

**Contraintes spécifiques :**

* Les passerelles de paiement doivent être intégrées de manière à respecter les standards de sécurité PCI-DSS.
* Le système doit permettre une gestion fluide des transactions et être capable de gérer un grand nombre de paiements simultanés.

**Intégrations avec d'autres systèmes :**

* **Département Training :** Synchronisation des paiements pour l'accès aux formations.
* **Département Marketing :** Suivi des paiements réalisés à travers des campagnes marketing ou promotions.
* **Département Scholar :** Validation des paiements des bourses pour les étudiants bénéficiaires (et/ou autres).
* **Département Familly :** Gestion des paiements (*et des remboursements*) pour les formations distribuées par les membres.

#### **5. Dépendances et interactions**

**Relations avec les autres départements :**

* **Département Training :** Validation des paiements pour l'accès aux formations.
* **Département Marketing :** Suivi des promotions et ajustement des prix en fonction des campagnes.
* **Département Scholar :** Suivi des bourses et financements pour les formations.
* **Département Familly :** Gestion des paiements des membres financés pour les formations.

**Dépendances fonctionnelles ou techniques :**

* L'intégration avec les passerelles de paiement dépend des API de Stripe, PayPal ,Payoneer et/ou autres.
* Le backend devra communiquer avec les bases de données de chaque département pour garantir une gestion fluide des données.
* Le système de facturation doit être conforme aux exigences fiscales locales.

### **6. Échéancier Ajusté pour les Phases de Développement**

| **Phase** | **Tâche** | **Sous-tâches** | **Durée estimée** | **Dépendances** |
| --- | --- | --- | --- | --- |
| **Janvier (S1-S2)** | **Audit des passerelles de paiement** | - Analyse des passerelles disponibles (Stripe, PayPal, etc.) | 1 semaine | Aucune |
|  |  | - Rapport d’audit détaillé sur la faisabilité et l’intégration | 1 semaine |  |
| **Janvier (S3-S4)** | **Maquette du système de paiement et de facturation** | - Conception de l'interface utilisateur (UI/UX) | 1 semaine | Audit des passerelles terminé |
|  |  | - Création des maquettes fonctionnelles du système | 1 semaine |  |
| **Février (S1-S2)** | **Développement de l’interface du système de paiement** | - Implémentation de l’UI de paiement (front-end) | 2 semaines | Maquettes fonctionnelles terminées |
| **Février (S3-S4)** | **Développement du système de paiement** | - Intégration des passerelles de paiement (Stripe, PayPal) | 2 semaines | Interface prête |
| **Mars (S1-S2)** | **Intégration complète du système de paiement** | - Mise en place des processus de validation des paiements | 2 semaines | Passerelles intégrées, système de paiement testé |
| **Mars (S3-S4)** | **Développement du système de facturation** | - Conception du module de facturation automatique | 2 semaines | Système de paiement opérationnel |
|  |  | - Développement de l’interface de consultation des factures | 1 semaine | Module de facturation développé |
| **Avril (S1-S2)** | **Production des APIs adéquates pour l’intégration** | - Conception des APIs pour la gestion des paiements et facturation | 2 semaines | Système de facturation prêt |
|  |  | - Développement et tests des APIs pour chaque département (Training, Marketing, Scholar, Family) | 2 semaines |  |
| **Avril (S3-S4)** | **Tests de sécurité, validation et ajustements** | - Tests de sécurité sur les transactions et les données financières | 2 semaines | APIs et systèmes développés |
|  |  | - Ajustements après retours des tests utilisateurs et de sécurité | 1 semaine |  |
| **Mai (S1)** | **Préparation du déploiement final** | - Préparation du système pour la mise en production | 1 semaine | Tests et ajustements terminés |
|  |  | - Formation des administrateurs et des responsables de départements | 1 semaine | Système prêt pour production |
| **Mai (S2)** | **Mise en production** | - Déploiement final du système de paiement et de facturation | 1 semaine | Déploiement validé, formation effectuée |
|  |  | - Documentation utilisateur et administrateur | 1 semaine |  |

### **7. Livrables Attendus par Phase**

| **Phase** | **Livrables Attendus** |
| --- | --- |
| **Janvier (S2)** | - Rapport d’audit des passerelles de paiement. |
|  | - Recommandations sur les passerelles les plus adaptées (Stripe, PayPal, etc.). |
| **Janvier (S4)** | - Maquettes fonctionnelles du système de paiement et de facturation. |
| **Février (S2)** | - Interface utilisateur pour le système de paiement (front-end). |
| **Février (S4)** | - Intégration complète des passerelles de paiement (Stripe, PayPal). |
| **Mars (S2)** | - Processus de validation des paiements opérationnel. |
| **Mars (S4)** | - Module de facturation automatique fonctionnel. |
|  | - Tableau de bord utilisateur pour consulter paiements et factures. |
| **Avril (S2)** | - APIs pour l’intégration avec les départements (Training, Marketing, Scholar, Family). |
| **Avril (S4)** | - Rapport des tests de sécurité et ajustements effectués. |
| **Mai (S1)** | - Formation des administrateurs et responsables de départements. |
| **Mai (S2)** | - Système de paiement et facturation en production. |
|  | - Documentation complète pour les utilisateurs et administrateurs. |

### **Explications**

1. **Audit des passerelles de paiement** : Phase préliminaire essentielle pour valider les solutions de paiement à intégrer dans le système. Cela inclut une analyse des options (Stripe, PayPal) en fonction des besoins spécifiques.
2. **Développement des interfaces et intégration** : Une fois les maquettes validées, le développement des interfaces utilisateur est réalisé en priorité, suivi par l’intégration technique des passerelles de paiement.
3. **Production des APIs adéquates** : Plutôt que de réaliser une intégration directe avec les départements à cette phase, il est plus efficace de développer des APIs flexibles permettant à chaque département (Training, Marketing, Scholar, Family) d'intégrer le système de manière fluide. Cela permet un découplage propre et un système plus extensible.
4. **Sécurisation et validation** : Les tests de sécurité doivent être faits après l’intégration et la mise en production des modules principaux, suivis de correctifs si nécessaire.
5. **Déploiement en mai** : Le système est mis en production avec une phase finale de documentation et de formation des administrateurs. La préparation du déploiement inclut également une phase de préproduction où tout est validé et testé.

#### **8. Critères de validation**

**Tests attendus :**

* Test des transactions de paiement, avec validation des processus de paiement et de génération des codes d’accès.
* Test de la création et de l’envoi des factures après paiement.
* Test de l'intégration avec les autres départements (Training, Marketing, Scholar, Family).
* Test de la sécurité du système de paiement (SSL/TLS).

**Conditions pour considérer le projet comme réussi :**

* Les paiements sont traités et validés correctement, avec une génération rapide des codes d’accès.
* Les factures sont générées, envoyées et stockées sans erreurs.
* Le système est fluide et sans erreur pour les utilisateurs.
* Les interactions entre les départements (Training, Marketing, Scholar, Familly) sont bien synchronisées.

**suggestions:**

* *à compléter…*