**CAHIER D’ANALYSE**

**Nom de l’entreprise : MadeInAkisch**

**Nom de projet : Application web de gestion de Cantine universitaire**

**Personne à contacter dans l’entreprise : AKPO K. Akisch**

**Adresse : Zanguera-Lomé-TOGO**

**Email : akpoakisch@emiaza.com**

**SOMMAIRE**

1. Modules et fonctionnalités
2. Acteurs / Utilisateurs
3. Objets du système
4. Diagramme de séquence
5. Règles de gestion
6. Diagramme de classe

**I. Modules et fonctionnalités**

Chaque **module** de l’application, aura les **fonctionnalités principales suivantes**:

| **Module** | **Fonctionnalité** |
| --- | --- |
| Authentification | - Inscription étudiant avec nom, prénom, institut, email et mot de passe |
|  | - Connexion étudiant (email ou nom utilisateur + mot de passe) |
|  | - Connexion administrateur |
|  | - Déconnexion sécurisée |
| Espace Étudiant | - Consultation du menu de la semaine (photo, nom, prix, type) |
|  | - Réservation de plats (quantité, suppléments, total) |
|  | - Suivi du statut des réservations (accepté, refusé, expiré) |
|  | - Historique des réservations |
|  | - Notation et avis sur les plats |
|  | - Consultation et modification du profil |
|  | - Consultation des notifications |
|  | - Recherche et filtres (type de plat, jour, etc.) |
| Espace Admin | - Ajout, modification, suppression de plats |
|  | - Planification hebdomadaire des plats (midi et soir par jour) |
|  | - Acceptation ou refus des réservations |
|  | - Gestion des avis étudiants |
|  | - Gestion de la liste des étudiants |
|  | - Envoi de notifications (ajout/modif plat, statut réservation) |
|  | - Consultation des statistiques (tableau de bord) |
|  | - Paramétrage global (durée de réservation, types de plats) |

**II. Acteurs / Utilisateurs**

Les **utilisateurs** du système (appelés aussi **acteurs**) et leurs interventions à chaque **modules/fonctionnalités**.

| **Acteurs** | **Modules accessibles** | **Fonctionnalités principales** |
| --- | --- | --- |
| **Étudiant** | Authentification | - Inscription, Connexion, Déconnexion |
|  |  | - Consulter le menu de la semaine |
|  |  | - Réserver un plat (avec suppléments) |
|  |  | - Suivre l’état de ses réservations |
|  | Espace Étudiant | - Consulter l’historique des réservations |
|  |  | - Noter les plats et laisser un avis |
|  |  | - Modifier son profil |
|  |  | - Recevoir des notifications |
|  |  | - Filtrer et rechercher des plats |
| **Administrateur** | Authentification | - Connexion Admin, Déconnexion |
|  |  | - Ajouter, modifier, supprimer un plat |
|  |  | - Planifier les repas (par jour et créneau) |
|  |  | - Valider ou refuser les réservations |
|  | Espace Admin | - Voir et gérer les avis des étudiants |
|  |  | - Gérer les comptes étudiants |
|  |  | - Envoyer des notifications |
|  |  | - Consulter les statistiques sur l’activité |
|  |  | - Gérer les paramètres globaux du système |

## ****III. Objets du système****

Les **objets du système** (ou entités) sont les éléments manipulés par l’application (utilisateurs, plats, réservations, etc.).  
On va décrire chaque objet sous forme de **tableau** avec ses **attributs**, **type** (texte, nombre, booléen, date, etc.), et **observations** (ce que l’attribut représente ou sa particularité).

### **3.1. Étudiant**

| **Attribut** | **Type** | **Observation** |
| --- | --- | --- |
| id\_etudiant | Entier | Identifiant unique |
| nom | Texte | Nom de l’étudiant |
| prénom | Texte | Prénom de l’étudiant |
| institut | Texte | Institut d’appartenance |
| email | Texte | Utilisé pour la connexion |
| mot\_de\_passe | Texte (hashé) | Sécurisé |
| date\_inscription | Date | Date de création du compte |

### **3.2. Plat**

| **Attribut** | **Type** | **Observation** |
| --- | --- | --- |
| id\_plat | Entier | Identifiant unique |
| nom\_plat | Texte | Nom du plat |
| photo | Fichier/image | Image du plat |
| prix | Numérique | Prix du plat |
| type\_plat | Texte | Ex: standard, VIP |
| description | Texte | Détail du plat |

### **3.3. Emploi du temps des plats**

| **Attribut** | **Type** | **Observation** |
| --- | --- | --- |
| id\_emploi\_temps | Entier | Identifiant |
| id\_plat | Clé étrangère | Référence au plat |
| jour | Texte | Lundi, mardi, etc. |
| créneau | Texte | Midi ou soir |

### **3.4. Réservation**

| **Attribut** | **Type** | **Observation** |
| --- | --- | --- |
| id\_reservation | Entier | Identifiant |
| id\_etudiant | Clé étrangère | Référence à l’étudiant |
| id\_plat | Clé étrangère | Référence au plat |
| quantite | Entier | Nombre de plats réservés |
| suppléments | Texte | Liste ou JSON des suppléments |
| total\_prix | Numérique | Total à payer |
| statut | Texte | En attente, accepté, refusé, expiré |
| date\_reservation | DateTime | Date et heure de la réservation |

### **3.5. Notification**

| **Attribut** | **Type** | **Observation** |
| --- | --- | --- |
| id\_notification | Entier | Identifiant |
| titre | Texte | Sujet de la notification |
| contenu | Texte | Message |
| destinataire | Texte ou id\_etudiant | Étudiant ou tous |
| date\_envoi | DateTime | Date et heure |

### **3.6. Avis**

| **Attribut** | **Type** | **Observation** |
| --- | --- | --- |
| id\_avis | Entier | Identifiant |
| id\_etudiant | Clé étrangère | Auteur de l’avis |
| id\_plat | Clé étrangère | Plat concerné |
| note | Entier (1 à 5) | Note du plat |
| commentaire | Texte | Détail de l’avis |
| date | DateTime | Date de publication |

### **3.7. Administrateur**

| **Attribut** | **Type** | **Observation** |
| --- | --- | --- |
| id\_admin | Entier | Identifiant |
| nom\_utilisateur | Texte | Identifiant unique |
| mot\_de\_passe | Texte (hashé) | Pour connexion sécurisée |

### **3.8. Paramètres système**

| **Attribut** | **Type** | **Observation** |
| --- | --- | --- |
| id\_paramètre | Entier | Identifiant |
| nom\_paramètre | Texte | Clé du paramètre |
| valeur | Texte/Nombre | Valeur associée (ex : délai d’expiration, types de plats autorisés) |

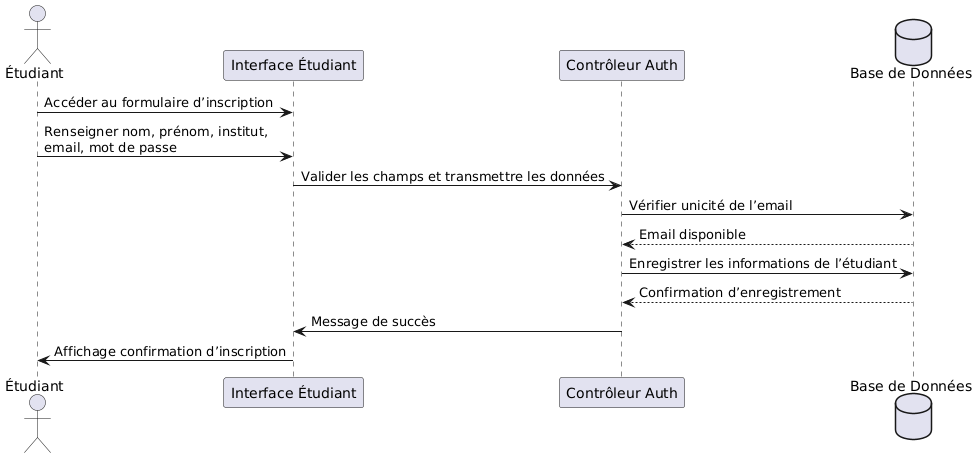
## ****IV. Scénarios / Diagrammes de séquence****

Ici, on va décrire plusieurs **scénarios d’interaction** entre les **acteurs** (Étudiant ou Admin) et le **système**.  
Ce sont des descriptions étape par étape qui montrent comment une fonctionnalité se déroule.

## ****4.1 : Inscription d’un étudiant****

### **Description**

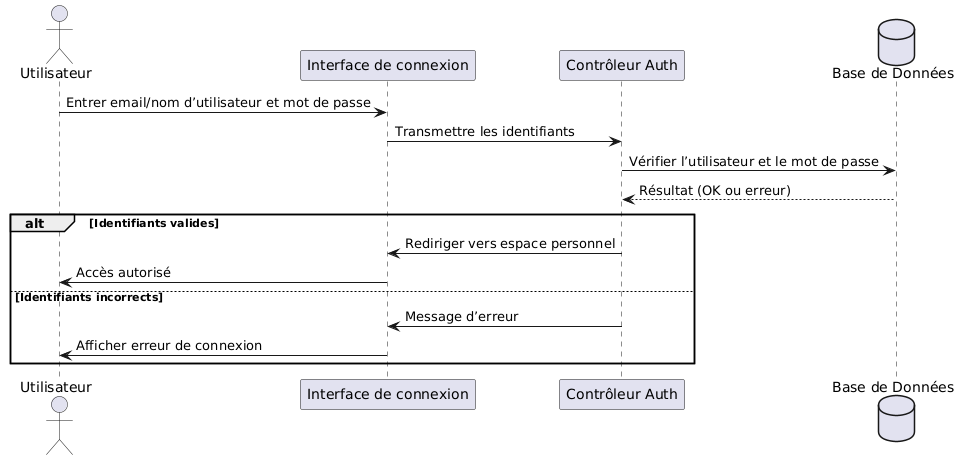
Ce diagramme illustre le processus d’inscription d’un nouvel étudiant sur la plateforme de gestion de cantine universitaire.  
L’étudiant remplit un formulaire d’inscription avec ses informations personnelles. L’interface transmet ces données au contrôleur, qui vérifie et enregistre les informations dans la base de données. Une confirmation est ensuite affichée à l’étudiant.



## ****4.2 : Connexion utilisateur****

### **Description**

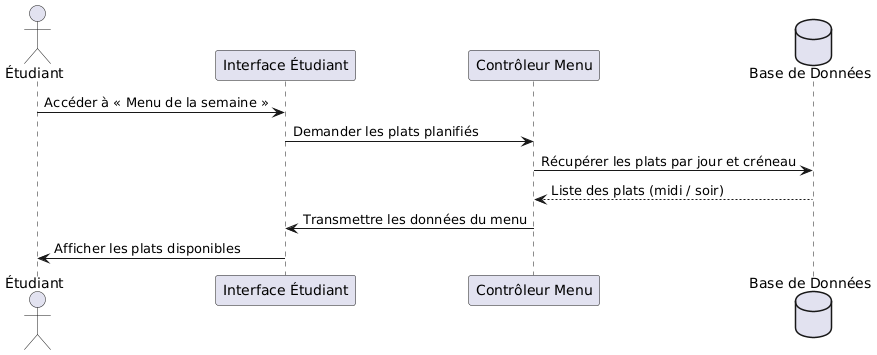
Ce diagramme représente le processus de connexion d’un utilisateur (étudiant ou administrateur).  
L’utilisateur saisit ses identifiants via l’interface. Le système vérifie ces données dans la base de données. En cas de succès, l’utilisateur est redirigé vers son espace. Sinon, un message d’erreur s’affiche.



## ****4.3 : Consultation du menu de la semaine****

### **Description**

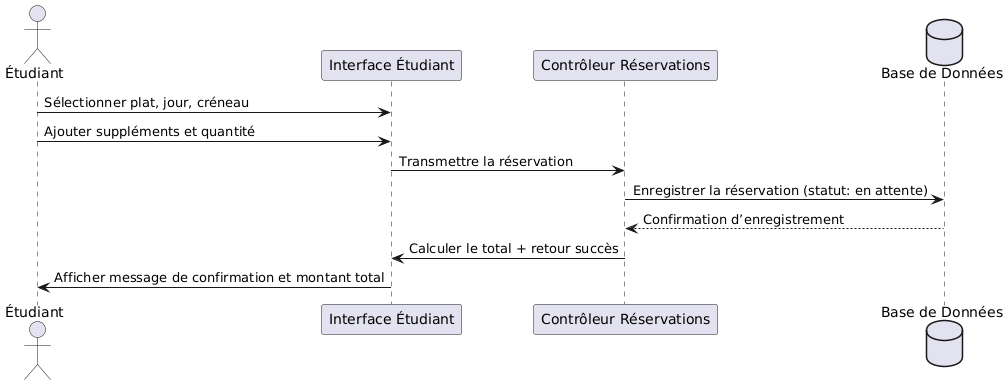
Ce diagramme montre le processus par lequel un étudiant consulte le menu des plats disponibles dans la semaine. L’interface interroge le système, qui récupère les plats planifiés dans la base de données, puis les retourne à l’interface pour affichage.



## ****4.4 : Réservation d’un plat****

### **Description**

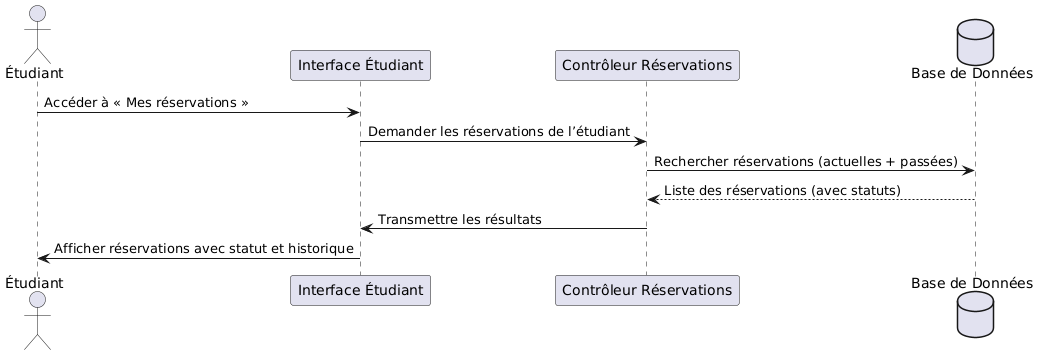
Ce diagramme décrit le processus complet de réservation d’un plat par un étudiant. Après avoir consulté le menu, l’étudiant sélectionne un plat, choisit un créneau (midi/soir), la quantité et les suppléments éventuels. Le système enregistre la réservation avec le statut "en attente", puis affiche une confirmation.



## ****4.5 : Suivi et historique des réservations****

### **Description**

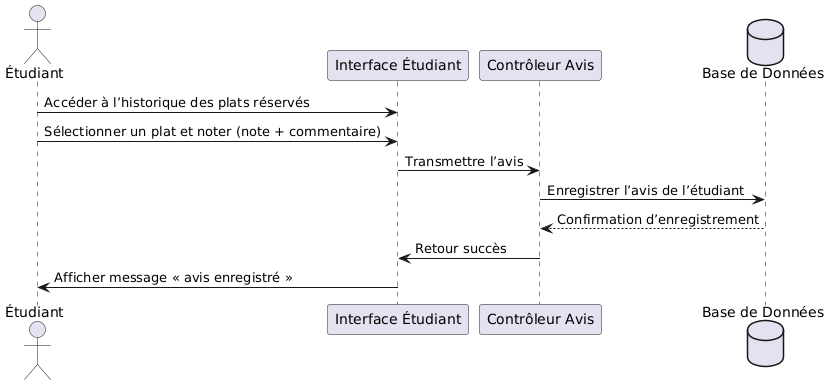
Ce diagramme montre comment un étudiant peut consulter l’état de ses réservations actuelles (acceptée, refusée, expirée) ainsi que son historique. L’interface interroge le système, qui récupère les données depuis la base de données, puis les affiche à l’utilisateur.



## ****4.6 : Notation d’un plat****

### **Description**

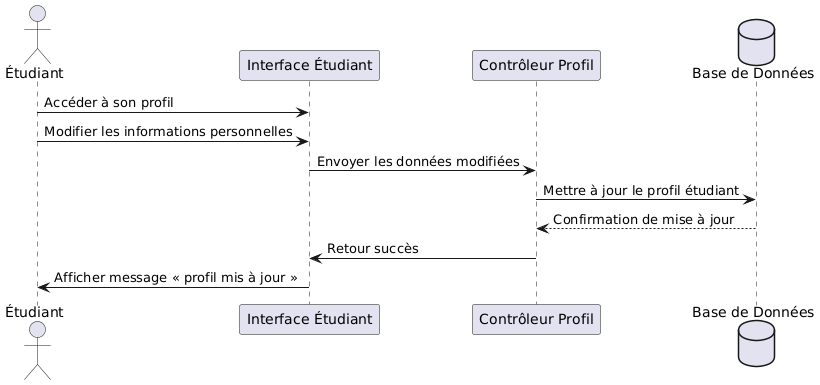
Ce diagramme décrit le processus par lequel un étudiant peut évaluer un plat qu’il a réservé et consommé. Il accède à son historique, sélectionne un plat, puis attribue une note sur 5 et un commentaire. Le système enregistre l’avis dans la base de données et affiche une confirmation.



## ****4.7 : Modification du profil étudiant****

### **Description**

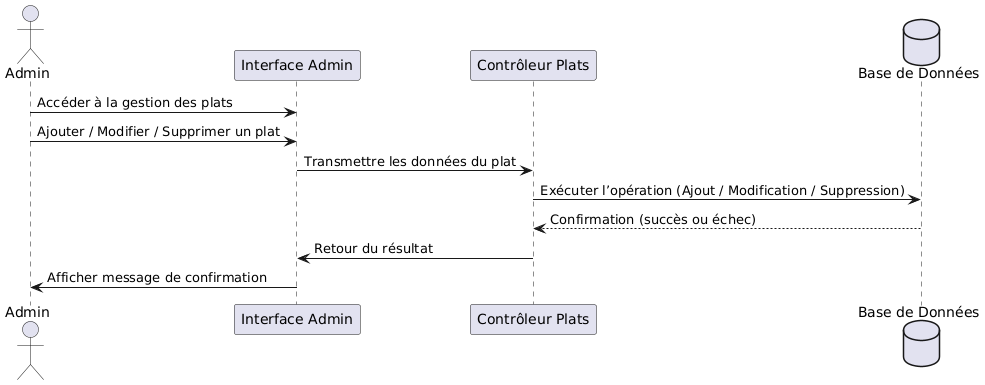
Ce diagramme illustre comment un étudiant peut modifier ses informations personnelles via son espace profil. Les nouvelles données sont envoyées au système, qui les met à jour dans la base de données. Une confirmation est ensuite retournée à l’utilisateur.



## ****4.8 : Gestion des plats (Ajout / Modification / Suppression)****

### **Description**

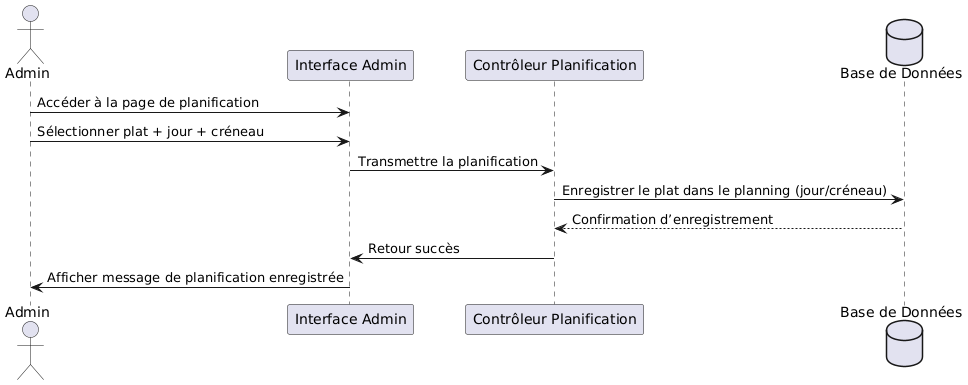
Ce diagramme illustre les opérations que peut effectuer l’administrateur sur les plats disponibles dans la cantine. L’admin peut ajouter un nouveau plat, modifier un plat existant, ou le supprimer. Le contrôleur envoie l’instruction à la base de données, qui effectue l’opération et renvoie un message de confirmation.



## ****4.9 : Planification des plats****

### **Description**

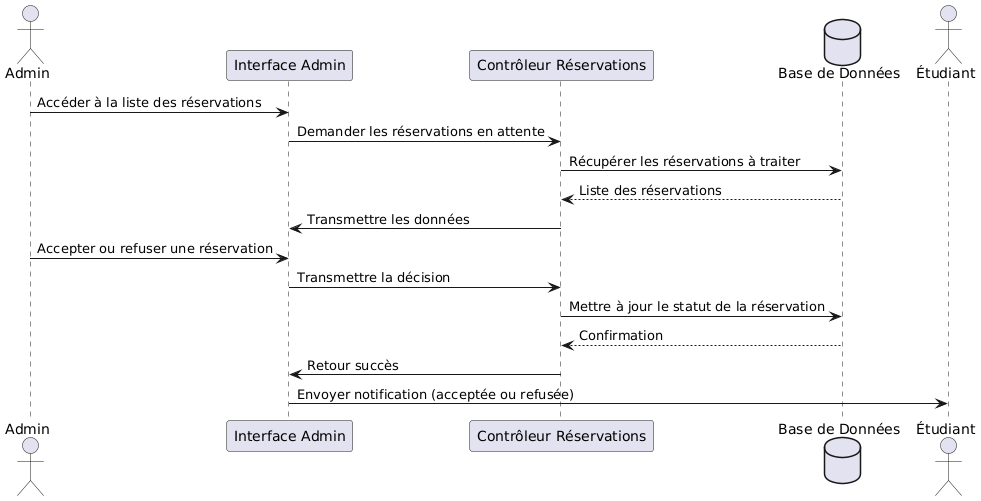
Ce diagramme montre comment l’administrateur planifie les plats à servir pour chaque jour de la semaine et chaque créneau (midi ou soir). Il sélectionne les plats et les associe à un jour/créneau via l’interface. Le contrôleur enregistre cette planification dans la base de données et retourne une confirmation.



## ****4.10 : Traitement des réservations (acceptation ou refus)****

### **Description**

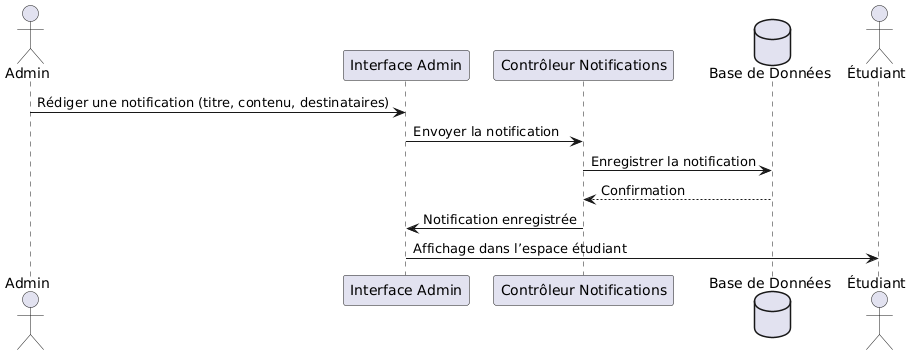
Ce diagramme décrit le processus par lequel l’administrateur traite les réservations effectuées par les étudiants. Il peut consulter les demandes, puis choisir de les accepter ou de les refuser. Le système met à jour le statut dans la base de données et notifie l’étudiant du résultat.



## ****4.11 : Envoi de notifications****

### **Description**

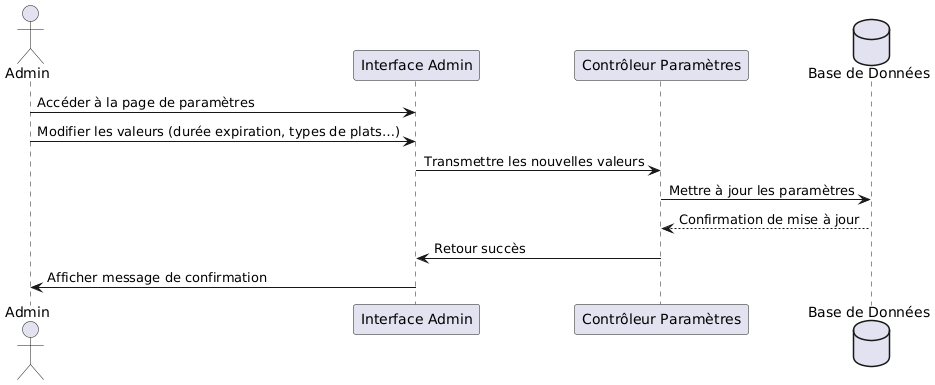
Ce diagramme représente le processus par lequel l’administrateur envoie une notification à un ou plusieurs étudiants. Cela peut concerner l’ajout ou modification d’un plat, le traitement d’une réservation, ou une information générale. Le message est transmis au contrôleur, enregistré dans la base, puis distribué aux destinataires.



## ****4.12 : Paramétrage global du système****

### **Description**

Ce diagramme illustre la gestion des paramètres globaux par l’administrateur. Cela peut inclure la durée d’expiration d’une réservation, les types de plats, ou d’autres règles de fonctionnement. Les paramètres sont modifiés via l’interface, enregistrés dans la base de données, puis une confirmation est retournée.



## ****V. Règles de gestion****

Les règles de gestion définissent les **contraintes métier** du système, les **relations entre les objets**, et les **conditions d’utilisation**.  
Elles garantissent que le fonctionnement de l’application respecte la logique métier de la cantine universitaire.

**1. Règles générales d'accès**

* Un étudiant ne peut accéder à son espace qu'après inscription et connexion réussie.
* Un seul compte administrateur peut exister dans le système.
* Un étudiant ne peut consulter que ses propres réservations, avis, et informations personnelles.

### **2. Règles sur les plats**

* Un plat doit avoir un nom, une description, un type (standard ou VIP), un prix et une photo.
* Un plat ne peut apparaître dans le menu que s’il a été planifié pour un jour et un créneau spécifique (midi ou soir).
* Un plat peut être modifié ou supprimé uniquement par l’administrateur.

### **3. Règles sur la planification**

* L’administrateur peut planifier plusieurs plats par jour et par créneau (midi / soir).
* Chaque plat peut être planifié pour plusieurs jours ou créneaux.
* La suppression d’un plat entraîne sa déplanification pour tous les créneaux concernés.

### **4. Règles sur les réservations**

* Un étudiant peut effectuer plusieurs réservations, mais une seule par plat par créneau.
* Une réservation a initialement le statut « en attente ».
* Une réservation peut être « acceptée », « refusée » ou « expirée » après dépassement du créneau horaire.
* Les réservations expirées sont automatiquement désactivées après un délai défini par l’administrateur.
* Les réservations doivent être validées ou refusées par l’administrateur.

### **5. Règles sur les avis**

* Un étudiant ne peut laisser un avis que sur un plat qu’il a réservé et qui a été accepté.
* Un avis comprend une note sur 5 et un commentaire.
* L’administrateur peut consulter et modérer les avis.

### **6. Règles sur les notifications**

* L’administrateur peut envoyer une notification aux étudiants pour :
  + L’ajout ou la modification d’un plat
  + L’acceptation ou le refus d’une réservation
  + Toute autre annonce administrative
* Les notifications sont automatiquement associées à leur destinataire et affichées dans l’espace étudiant.

### **7. Règles sur les paramètres du système**

* L’administrateur peut définir la durée maximale de validité d’une réservation.
* L’administrateur peut modifier la liste des types de plats disponibles.
* Toute modification de paramètre prend effet immédiatement après enregistrement.

## ****VI. Diagramme de classes****

Le **diagramme de classes UML** représente les **entités principales** du système (étudiant, plat, réservation, etc.) ainsi que :

* Leurs **attributs**,
* Les **relations entre elles** (associations, cardinalités),
* Et parfois les **méthodes principales** (si besoin).

