

# BTS SIO 2025



**Administration des systèmes et des réseaux (E6 – SISR)**



**Conception et développement d'applications (E6 – SLAM)**

## PAGE DE PRÉSENTATION DU DOSSIER

N° d'inscription<sup>1</sup> : |0|2|0|4|6|6|9|7|3|3|2|

NOM : Chevrollier.....

PRÉNOM : Nathan.....

Date de passage<sup>1</sup> : ..12.... / ...06.. / 2025

Heure de passage<sup>1</sup> : ..13...h..30..

CATÉGORIE CANDIDAT <sup>2</sup> (UNE CASE À COCHER)	
<input type="checkbox"/> Scolaire <input type="checkbox"/> Apprenti <input checked="" type="checkbox"/> Formation professionnelle continue <input type="checkbox"/> Expérience professionnelle 3 ans	<input type="checkbox"/> Ex-scolaire <input type="checkbox"/> Ex-apprenti <input type="checkbox"/> Ex-formation professionnelle continue

<sup>1</sup> Informations communiquées sur votre convocation envoyée courant mars 2025 sur votre compte **Cyclades**

<sup>2</sup> Informations communiquées sur votre confirmation d'inscription.

Tampon de L'établissement

**SIEC – maison des examens**

7 rue Ernest Renan  
94749 ARQUEIL CEDEX  
Tél : 01 49 12 23 00  
[www.siec.education.fr](http://www.siec.education.fr)



## **Sommaire du dossier E6**

Page de garde du dossier

Annexe 10 A et B – B-Formation

Annexes projet 1 – B-Formation :

Annexe 0 – Cahier des charges

Annexe 1 – Dictionnaire de données

Annexe 2 – MCD

Annexe 3 – MLD

Annexe 4 – MPD

Annexe 5 – Diagramme de Gantt

Annexe 6 – Diagramme de contexte

Annexe 7 – Diagramme de package

Annexe 8 – Diagramme de classes

Annexe 9 – Cartographie technique

Annexe 10 A et B – ClicknEat

Annexes projet 2 – ClicknEat :

Annexe 0 – Cahier des charges

Annexe 1 – Dictionnaire de données

Annexe 2 – MCD

Annexe 3 – MLD

Annexe 4 – MPD

Annexe 5 – Diagramme de contexte

Annexe 6 – Diagramme de package

Annexe 7 – Diagramme de classes

Annexe 8 – Diagramme de Gantt

Annexe 9 – Cartographie technique

## ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle (recto)

## Épreuve E6 - Conception et développement d'applications (option SLAM)

DESCRIPTION D'UNE RÉALISATION PROFESSIONNELLE		N° réalisation :
Nom, prénom : Chevrollier Nathan		N° candidat : 02046697332
Épreuve ponctuelle <input checked="" type="checkbox"/>	Contrôle en cours de formation <input type="checkbox"/>	Date : ..24 / ..04 / .2025...
<b>Organisation support de la réalisation professionnelle</b> My Digital School Angers		
Intitulé de la réalisation professionnelle	B-Formation : Application PHP de gestion des classes et des présences	
Période de réalisation : Sept 2024 - mars 2025	Lieu : Angers	
Modalité : <input checked="" type="checkbox"/> Seul(e) <input type="checkbox"/> En équipe		
<b>Compétences travaillées</b>		
<input checked="" type="checkbox"/> Concevoir et développer une solution applicative <input checked="" type="checkbox"/> Assurer la maintenance corrective ou évolutive d'une solution applicative <input checked="" type="checkbox"/> Gérer les données		
<b>Conditions de réalisation<sup>5</sup> (ressources fournies, résultats attendus)</b>		
<p>Réalisée dans le cadre de la formation, ce projet repose sur un besoin fictif défini par moi-même : une application pour la gestion des classes et des présences d'un établissement d'enseignement.</p> <p>J'ai rédigé le cahier des charges en tenant compte des exigences pédagogiques, guidée par mon enseignant pour définir les objectifs.</p> <p>Résultats attendus : une application web fonctionnelle développée en PHP natif avec une base de données MySQL, des interfaces utilisateur par role, permettant la gestion des étudiants, des cours et des présences, ainsi qu'une documentation technique décrivant le fonctionnement de l'application.</p>		
<b>Description des ressources documentaires, matérielles et logicielles utilisées<sup>6</sup></b>		
<p>Documentaires : Cahier des charges , différentes documentations pour les langages utilisés</p> <p>Matérielles : PC sous Windows 11</p> <p>Logicielles :</p> <ul style="list-style-type: none"> <li>- VSCode / windsurf (environnement de développement)</li> <li>- XAMPP et Apache (serveur local)</li> <li>- VPS OVH debian et NGINX (serveur web)</li> <li>- phpMyAdmin (système de gestion de base de données (BDD))</li> <li>- Git et GitHub (solutions de versioning)</li> <li>- MySQLDump, Crontab(gestion et automatisation des sauvegardes BDD)</li> <li>- GoogleChrome/opera (navigateur web)</li> </ul> <p>Langages : HTML, CSS (Bootstrap), PHP, SQL, Bash</p>		
<b>Modalités d'accès aux productions<sup>7</sup> et à leur documentation<sup>8</sup></b>		
Application : <a href="https://bformation.chevrolliernathan.fr">https://bformation.chevrolliernathan.fr</a> Identifiants : -admin : nathan@orange.fr / mdp : blabla -élèves : gestiere@gmail.com /mdp : blabla -professeur : Rippoche@gmail.com /mdp blabla		code source et documentation: <a href="https://github.com/NathanChevrollier/B-formation">https://github.com/NathanChevrollier/B-formation</a>

<sup>5</sup> En référence aux *conditions de réalisation et ressources nécessaires* du bloc « Conception et développement d'applications » prévues dans le référentiel de certification du BTS SIO.

<sup>6</sup> Les réalisations professionnelles sont élaborées dans un environnement technologique conforme à l'annexe II.E du référentiel du BTS SIO.

<sup>7</sup> Conformément au référentiel du BTS SIO « *Dans tous les cas, les candidats doivent se munir des outils et ressources techniques nécessaires au déroulement de l'épreuve. Ils sont seuls responsables de la disponibilité et de la mise en œuvre de ces outils et ressources. La circulaire nationale d'organisation précise les conditions matérielles de déroulement des interrogations et les pénalités à appliquer aux candidats qui ne se seraient pas munis des éléments nécessaires au déroulement de l'épreuve.* ». Les éléments peuvent être un identifiant, un mot de passe, une adresse réticulaire (URL) d'un espace de stockage et de la présentation de l'organisation du stockage.

<sup>8</sup> Lien vers la documentation complète, précisant et décrivant, si cela n'a été fait au verso de la fiche, la réalisation professionnelle, par exemples service fourni par la réalisation, interfaces utilisateurs, description des classes ou de la base de données.

**ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle  
(verso, éventuellement pages suivantes)****Épreuve E6 - Conception et développement d'applications (option SLAM)****Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs**

Contexte :

Dans le cadre du BTS SIO SLAM, ce projet avait pour objectif de développer une application web permettant la gestion des utilisateurs, classes, des matières et des présences.

Les élèves ne peuvent valider leur présence que si l'enseignant l'a autorisée.

Contrainte principale : utilisation d'une base de données MySQL pour la gestion des différentes entités, programmation orientée objet.

Réalisé en autonomie avec la guidance pédagoque de Kévin Niel, intervenant "Solutions logicielles et Applications Métiers"  
Tâches :

- Rédaction du cahier des charges

- Modélisation UML :

- Diagramme de contexte
- Diagramme de paquetages
- Diagramme de classes
- Cartographie technique

- Modélisation de la base de données MySQL selon la méthode Merise :

• Rédaction d'un dictionnaire de données

• MCD (Modèle Conceptuel de Données) : Identification des entités (utilisateurs, classes, matières, présences) et relations

• MLD (Modèle Logique de Données) : Transformation en tables relationnelles avec clés primaires et étrangères

• MPD (Modèle Physique de Données) : Création des tables et contraintes en SQL

- Implémentation de l'authentification et gestion des rôles :

• Mise en place d'un système de connexion et d'authentification sécurisée, avec une gestion des sessions utilisateurs et un cryptage des mots de passe

• Ajout de la gestion des rôles pour restreindre les accès aux fonctionnalités spécifiques (administration, gestion des classes, consultation d'emploi du temps, etc.)

- Création des fonctionnalités principales :

• Gestion des utilisateurs : création, modification, suppression et consultation des comptes utilisateurs (administrateurs, enseignants, élèves)

• Gestion des classes et des matières : ajout, suppression, modification des classes et des matières associées

• Emploi du temps : gestion des horaires et affectation des matières aux classes

• Gestion de la présence : les enseignants peuvent marquer les élèves comme présents, et les élèves peuvent ensuite valider leur présence via un bouton visible uniquement si l'enseignant l'a autorisé

- Interface utilisateur avec Bootstrap :

• Création du tableau de bord administrateur : permet de gérer les utilisateurs, les classes et les matières

• Création du tableau de bord enseignant : permet de gérer les présences et de consulter l'emploi du temps

• Création du tableau de bord élève : permet de consulter son emploi du temps et de valider sa présence une fois autorisé

Sauvegarde :

• Création d'un script de sauvegarde automatique depuis le VPS (MySQLDump et Crontab)

Documentation disponible :

- Cahier des charges
- Diagramme de GANTT
- Diagrammes UML
- Modélisation de la BDD

**ANNEXE 10-B : Modèle d'attestation de respect de l'annexe II.E – Environnement technologique pour la certification du référentiel**

**Épreuve E6 - Conception et développement d'applications (option SLAM)**

**CONTRÔLE DE L'ENVIRONNEMENT TECHNOLOGIQUE**

**En référence à l'annexe II.E –« Environnement technologique pour la certification » du référentiel du BTS SIO**

<b>Identification<sup>2</sup></b>		<b>SLAM</b>
-----------------------------------	--	-------------

**1. Environnement commun aux deux options**

**1.1 L'environnement technologique supportant le système d'information de l'organisation cliente comporte au moins :**

<b>Éléments</b>	<b>Description de l'implantation dans le centre d'examen (nom du service ou de l'outil et caractéristiques techniques)</b>	<b>Remarques de la commission d'interrogation</b>
Un service d'authentification	Auth laravel + Auth PHP natif	
Un SGBD	MariaDB = MySQL - phpMyAdmin	
Un accès sécurisé à internet	certificats SSL (let'sencrypt) sur vos VPS	
Un environnement de travail collaboratif	git & github VSCode (outils / plugins, etc...)	
Deux serveurs, éventuellement virtualisés, basés sur des systèmes d'exploitation différents, dont l'un est un logiciel libre ( <i>open source</i> )	VPS sur Debian + serveur de développement sur Windows (mamp / wamp / xampp)	

<sup>2</sup> Nom et adresse du centre d'examen ou identification de la personne candidate individuelle (numéro, nom, prénom)

**ANNEXE 10-B (suite) : Modèle d'attestation de respect de l'annexe II.E – Environnement technologique pour la certification du référentiel**  
**Épreuve E6 - Conception et développement d'applications (option SLAM)**

<b>Éléments</b>	<b>Description de l'implantation dans le centre d'examen (nom du service ou de l'outil et caractéristiques techniques)</b>	<b>Remarques de la commission d'interrogation</b>
Une solution de sauvegarde	GIT - dump SQL	
Des ressources dont l'accès est sécurisé et soumis à habilitation	GIT(hub) - Gestion de rôle dans les projets de dev	
Deux types de terminaux dont un mobile (type <i>smartphone</i> ou encore tablette)	Web - Responsive (RWD) - electron desktop (calculatrice) - Script SHELL	

**1.2 Des outils sont mobilisés pour la gestion de la sécurité :**

<b>Éléments</b>	<b>Description de l'implantation dans le centre d'examen (nom du service ou de l'outil et caractéristiques techniques)</b>	<b>Remarques de la commission d'interrogation</b>
Gestion des incidents	formulaire de contact support logs (error & access) VPS	
Détection et prévention des intrusions	Monitoring & services OVH (anti-DDoS automatique) - IPTABLE - Fail2ban (optionnel) - changement des ports par défaut - désactivation du root - LOGS	
Chiffrement	HTTPS (SSL) - hash password en BDD	
Analyse de trafic	Monitoring OVH - Google Analytics	

**Remarque : les logiciels de simulation ou d'émulation sont utilisés en réponse à des besoins de l'organisation. Ils ne peuvent se substituer complètement à des équipements réels dans l'environnement technologique d'apprentissage.**

**ANNEXE 10-B (suite) : Modèle d'attestation de respect de l'annexe II.E – Environnement technologique pour la certification du référentiel**  
**Épreuve E6 - Conception et développement d'applications (option SLAM)**

**2. Savoirs spécifiques à l'option « solutions logicielles et applications métiers » (SLAM)**

**2.1 L'environnement technologique supportant le système d'information de l'organisation cliente comporte au moins :**

<b>Éléments</b>	<b>Description de l'implantation dans le centre d'examen (nom du service ou de l'outil et caractéristiques techniques)</b>	<b>Remarques de la commission d'interrogation</b>
Un ou deux environnements de développement disposant d'outils de gestion de tests et supportant un cadre applicatif ( <i>framework</i> ) et au moins deux langages	HTML/CSS PHP SQL Javascript - PEST pour les tests, Laravel	
Une bibliothèque de composants logiciels	Bootstrap - calculette immo : JSPDF (un truc du genre) - JSCalendar / Fullcalendar	
Un SGBD avec langage de programmation associé	MariaDB MySQL	
Un logiciel de gestion de versions et de suivi de problèmes d'ordre logiciel	GIT & Github - tests unitaires - Github ACTION	
Une solution permettant de tester les comportements anormaux d'une application	Tests fonctionnels / unitaires @CSRF, failles de sécurité, protection injection SQL/XSS	

**2.2 Les activités de l'organisation cliente s'appuient sur aux moins deux solutions applicatives opérationnelles permettant d'offrir un accès sécurisé à des données hébergées sur un site distant. Au sein des architectures de ces solutions applicatives doivent figurer l'exploitation de mécanismes d'appel à des services applicatifs distants et au moins trois des situations ci-dessous :**

<b>Éléments</b>	<b>Description de l'implantation dans le centre d'examen (nom du service ou de l'outil et caractéristiques techniques)</b>	<b>Remarques de la commission d'interrogation</b>
Du code exécuté sur le système d'exploitation d'une solution technique d'accès fixe (type client lourd)	Calculette Electron	
Du code exécuté dans un navigateur Web (type client léger ou riche)	PHP + Laravel	
Du code exécuté sur le système d'exploitation d'une solution technique d'accès mobile		
Du code exécuté sur le système d'exploitation d'un serveur	Script Serveur shell	

**2.3 Une solution applicative peut être issue d'un développement spécifique ou de la modification du code d'un logiciel notamment open source.**

**2.4 Les solutions applicatives présentes dans le contexte sont opérationnelles et leur code source est accessible dans un environnement de développement opérationnel au moment de l'épreuve.**

# Annexes du projet B-Formation

---

Toute la documentation du projet est aussi disponible avec les fichiers originels sur le repo github correspondant au projet : [Cliquez ici](#)  
(<https://github.com/NathanChevrollier/B-formation>)

## Sommaire

Annexe 0 – Cahier des charges

Annexe 1 – Dictionnaire de données

Annexe 2 – MCD (Modèle Conceptuel de Données)

Annexe 3 – MLD (Modèle Logique de Données)

Annexe 4 – MPD (Modèle Physique de Données)

Annexe 5 – Diagramme de Gantt

Annexe 6 – Diagramme de contexte

Annexe 7 – Diagramme de package

Annexe 8 – Diagramme de classes

Annexe 9 – Cartographie technique

## Annexe 0 – Cahier des charges

### B-Formation Cahier des charges

#### Contexte

Le client nous a mandaté pour développer une application web de gestion des classes et des présences intégrant un système de validation des présences via un bouton de signature. L'application doit permettre de centraliser la gestion des utilisateurs (enseignants et élèves), des matières, des classes et des emplois du temps, ainsi que la validation des présences, avec une traçabilité précise grâce à l'enregistrement de l'horodatage.

#### Objectif

Fournir une solution complète qui :

Gère efficacement les utilisateurs (administrateurs, enseignants, élèves) et leurs droits d'accès.

Organise les classes, matières et emplois du temps de manière intuitive.

Permet la validation des présences par un bouton de signature visible uniquement lorsque l'enseignant a confirmé la présence de l'élève, et enregistre un horodatage dans la base de données.

#### Public cible

Établissements d'enseignement et responsables pédagogiques souhaitant moderniser et sécuriser la gestion des classes et des présences.

#### Expression des besoins

Le client souhaite disposer d'une application permettant de :

Assurer une gestion centralisée et optimisée des utilisateurs et des classes.

Offrir une interface conviviale et sécurisée pour la gestion des emplois du temps et des affectations.

Intégrer un mécanisme de validation des présences basé sur un bouton de signature (affiché uniquement après confirmation de l'enseignant) qui, lorsqu'il est cliqué, enregistre automatiquement l'horodatage dans la base de données.

#### Fonctionnalités MVP (Minimum Viable Product)

**Gestion des utilisateurs :**Création, modification et suppression des comptes pour administrateurs, enseignants et élèves.

Authentification sécurisée (email et mot de passe chiffré) avec gestion des rôles.

**Gestion des classes :**Création, modification et suppression des classes.

Affectation des enseignants et des élèves aux classes.

**Gestion des matières et emplois du temps :**Configuration des matières et définition des créneaux horaires par classe.

Consultation de l'emploi du temps via un espace personnalisé.

**Validation des présences :**Mise en place d'un bouton de signature, visible uniquement lorsque l'enseignant a confirmé la présence.

Enregistrement de l'horodatage du clic dans la base de données pour assurer la traçabilité.

### **Fonctionnalités MLP (Minimum Lovable Product)**

Visualisation en temps réel des présences et des signatures.

Interface responsive et design moderne grâce à Bootstrap.

Historique détaillé des signatures et possibilité de filtrer les données par classe ou par période.

### **Contraintes techniques**

Développement : PHP orienté objet (POO) afin d'assurer une structure modulaire et évolutive.

Base de données : MySQL avec une modélisation relationnelle robuste (MCD, MLD, MPD).

Environnements : **Local** : Développement sous XAMPP (Apache, PHP, MySQL, PhpMyAdmin).

**Production** : Déploiement sur un VPS OVH avec nginx et gestion via PhpMyAdmin.

Sécurité : Gestion sécurisée des sessions et des authentifications (cryptage avec password\_hash()).

Utilisation de HTTPS et mise en place d'un certificat SSL pour le déploiement.

Déploiement : Déploiement sur un nom de domaine avec intégration continue des mises à jour.

## **Template et Essais**

Interface utilisateur : Utilisation de Bootstrap pour garantir une interface responsive et ergonomique.

Validation des fonctionnalités : Réalisation d'essais manuels pour confirmer le bon fonctionnement de l'application (authentification, gestion des classes, validation des présences, etc.).

## Annexe 1 – Dictionnaire de données

### Dictionnaire de données

Table USER

Nom	Type	Taille	Description	Contrainte
id	INT	11	Identifiant unique de l'utilisateur	Clé primaire, Auto-increment
firstname	VARCHAR	255	Prénom de l'utilisateur	Non null
surname	VARCHAR	255	Nom de famille de l'utilisateur	Non null
email	VARCHAR	255	Adresse email de l'utilisateur	Non null
password	VARCHAR	255	Mot de passe hashé	Non null
role	VARCHAR	255	Rôle de l'utilisateur (admin, teacher, student)	Nullable
class_id	INT	11	Référence à la classe de l'étudiant	Clé étrangère, Nullable

Table CLASS

Nom	Type	Taille	Description	Contrainte
id	INT	11	Identifiant unique de la classe	Clé primaire, Auto-increment
name	VARCHAR	255	Nom de la classe	Non null

Table SUBJECT

Nom	Type	Taille	Description	Contrainte
id	INT	11	Identifiant unique de la matière	Clé primaire, Auto-increment
name	VARCHAR	255	Nom de la matière	Non null

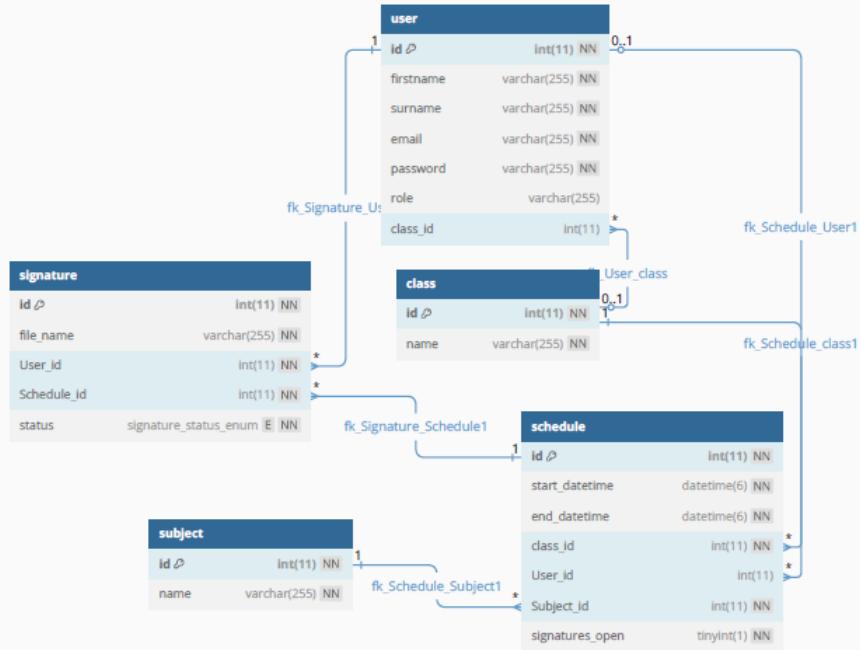
Table SCHEDULE

Nom	Type	Taille	Description	Contrainte
id	INT	11	Identifiant unique du créneau	Clé primaire, Auto-increment
start_datetime	DATETIME	6	Date et heure de début du cours	Non null
end_datetime	DATETIME	6	Date et heure de fin du cours	Non null
class_id	INT	11	Référence à la classe concernée	Clé étrangère, Non null
User_id	INT	11	Référence à l'enseignant	Clé étrangère, Nullable
Subject_id	INT	11	Référence à la matière enseignée	Clé étrangère, Non null
signatures_open	TINYINT	1	Indique si les signatures sont ouvertes (0/1)	Non null, Défaut 0

Table SIGNATURE

Nom	Type	Taille	Description	Contrainte
id	INT	11	Identifiant unique de la signature	Clé primaire, Auto-increment
file_name	VARCHAR	255	Nom du fichier de signature	Non null
User_id	INT	11	Référence à l'étudiant signataire	Clé étrangère, Non null
Schedule_id	INT	11	Référence au créneau concerné	Clé étrangère, Non null
status	ENUM	-	Statut de la signature (pending, validated, absent)	Non null, Défaut 'pending'

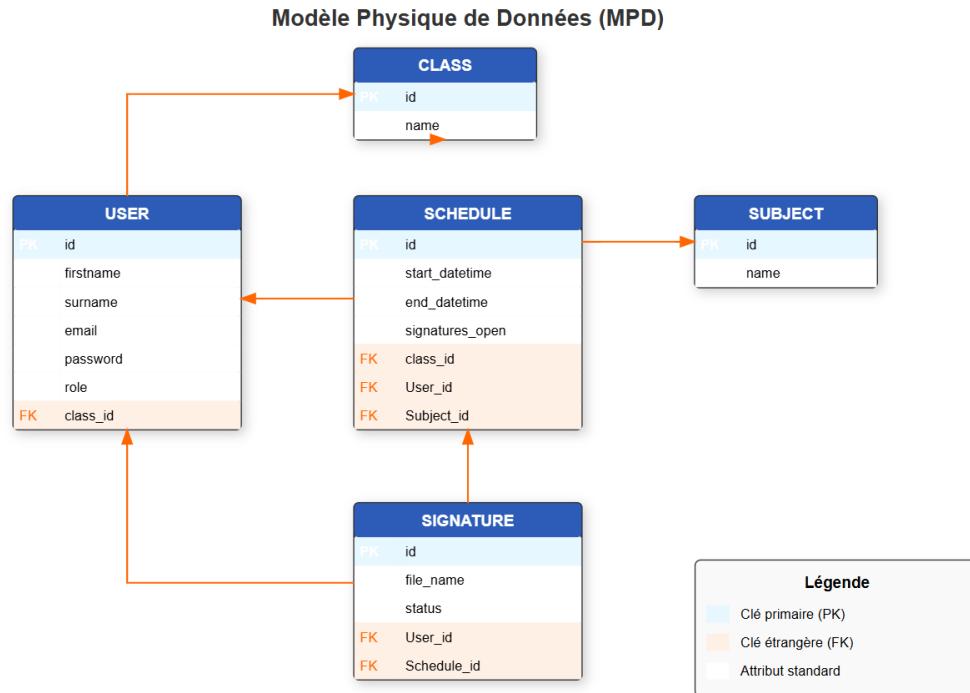
## Annexe 2 – MCD (Modèle Conceptuel de Données)



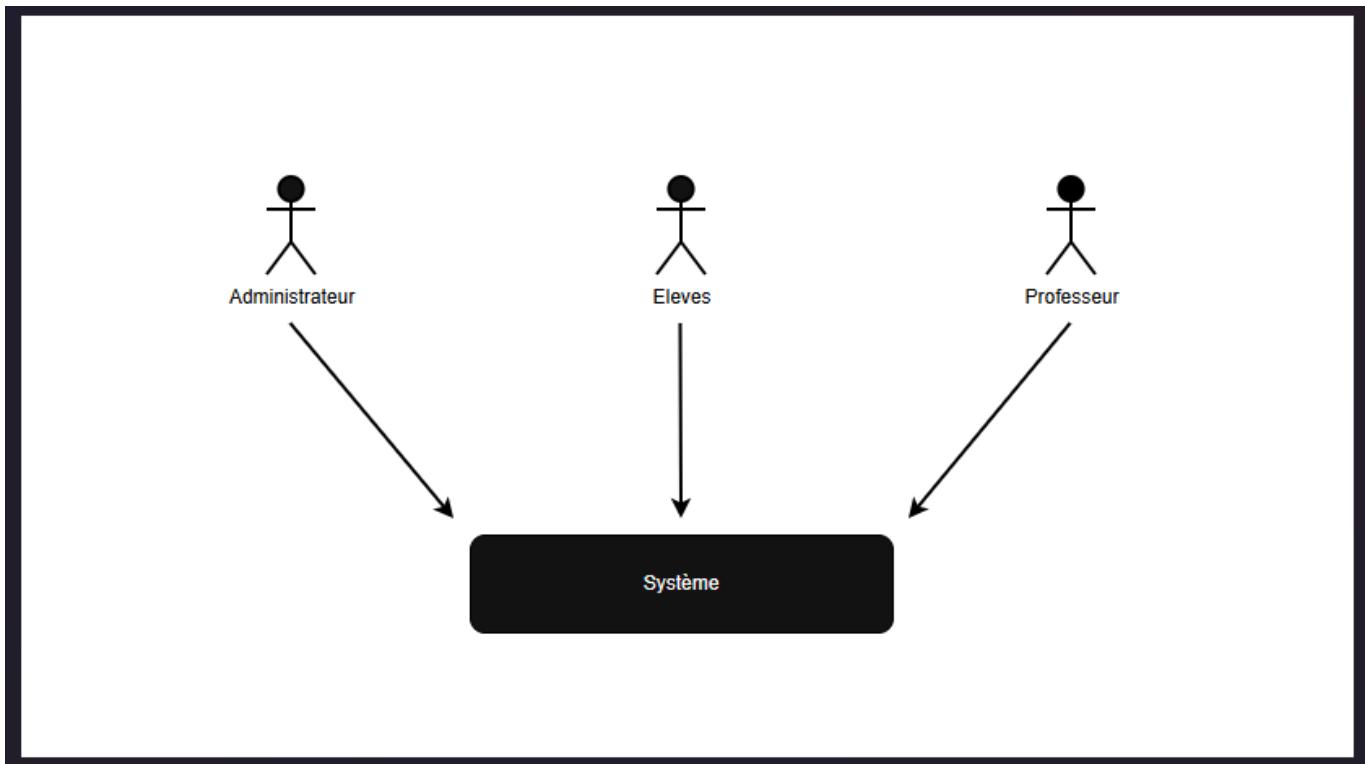
### Annexe 3 – MLD (Modèle Logique de Données)

```
CREATE TABLE CLASS (id INT NOT NULL,name VARCHAR(255) NOT NULL,PRIMARY KEY (id));  
  
CREATE TABLE SUBJECT (id INT NOT NULL,name VARCHAR(255) NOT NULL,PRIMARY KEY (id));  
  
CREATE TABLE USER (id INT NOT NULL,firstname VARCHAR(255) NOT NULL,surname  
VARCHAR(255) NOT NULL,email VARCHAR(255) NOT NULL,password VARCHAR(255)  
NOT NULL,role VARCHAR(255),class_id INT,FOREIGN KEY (class_id) REFERENCES  
CLASS(id),PRIMARY KEY (id));  
  
CREATE TABLE SCHEDULE (id INT NOT NULL,start_datetime DATETIME(6) NOT  
NULL,end_datetime DATETIME(6) NOT NULL,signatures_open TINYINT(1) NOT NULL  
DEFAULT 0,class_id INT NOT NULL,User_id INT,Subject_id INT NOT NULL,FOREIGN KEY  
(class_id) REFERENCES CLASS(id),FOREIGN KEY (User_id) REFERENCES  
USER(id),FOREIGN KEY (Subject_id) REFERENCES SUBJECT(id),PRIMARY KEY (id));  
  
CREATE TABLE SIGNATURE (id INT NOT NULL,file_name VARCHAR(255) NOT NULL,  
  
status ENUM('pending', 'validated', 'absent') NOT NULL DEFAULT 'pending',User_id INT  
NOT NULL,Schedule_id INT NOT NULL,FOREIGN KEY (User_id) REFERENCES  
USER(id),FOREIGN KEY (Schedule_id) REFERENCES SCHEDULE(id),PRIMARY KEY (id));
```

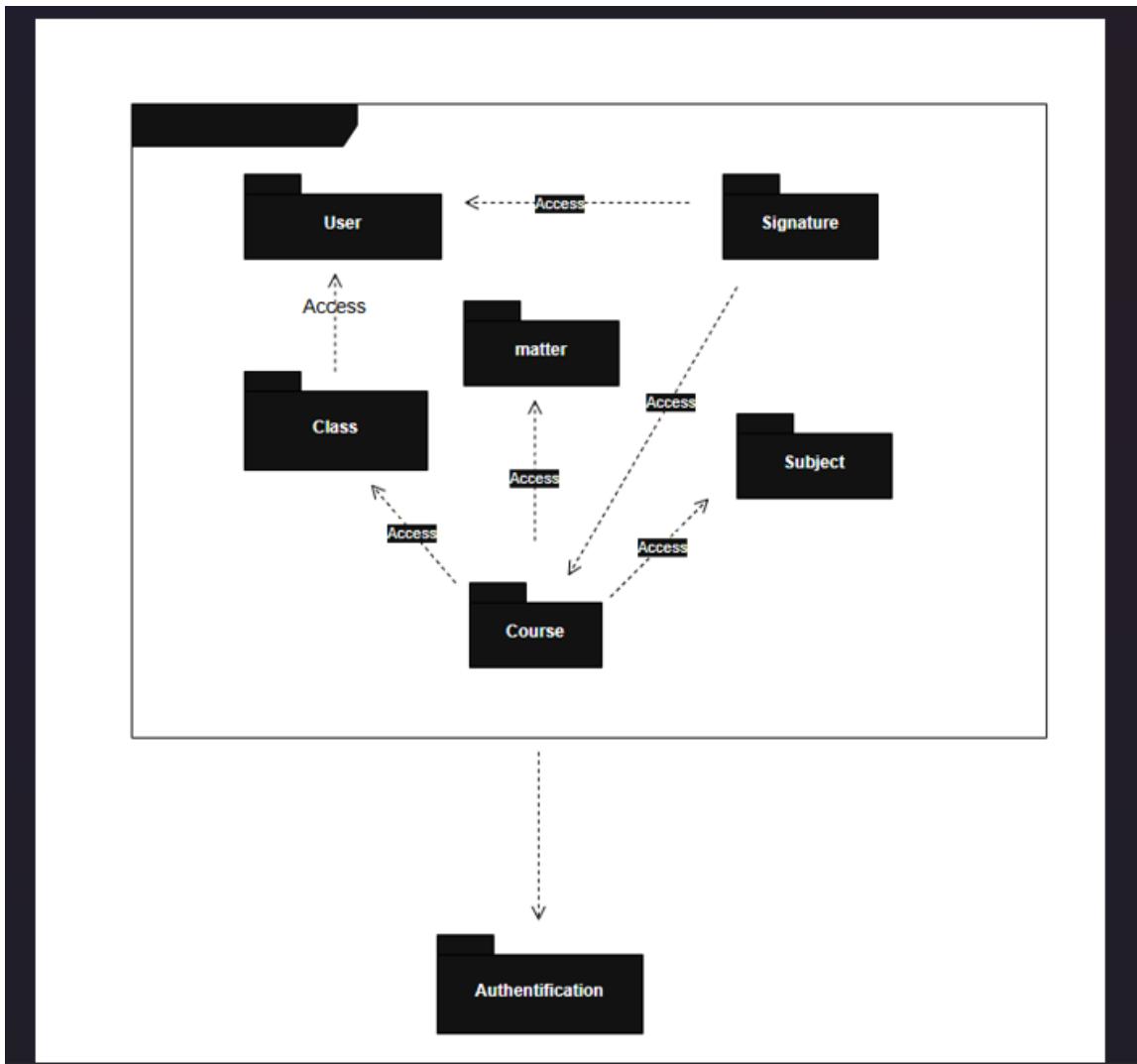
## Annexe 4 – MPD



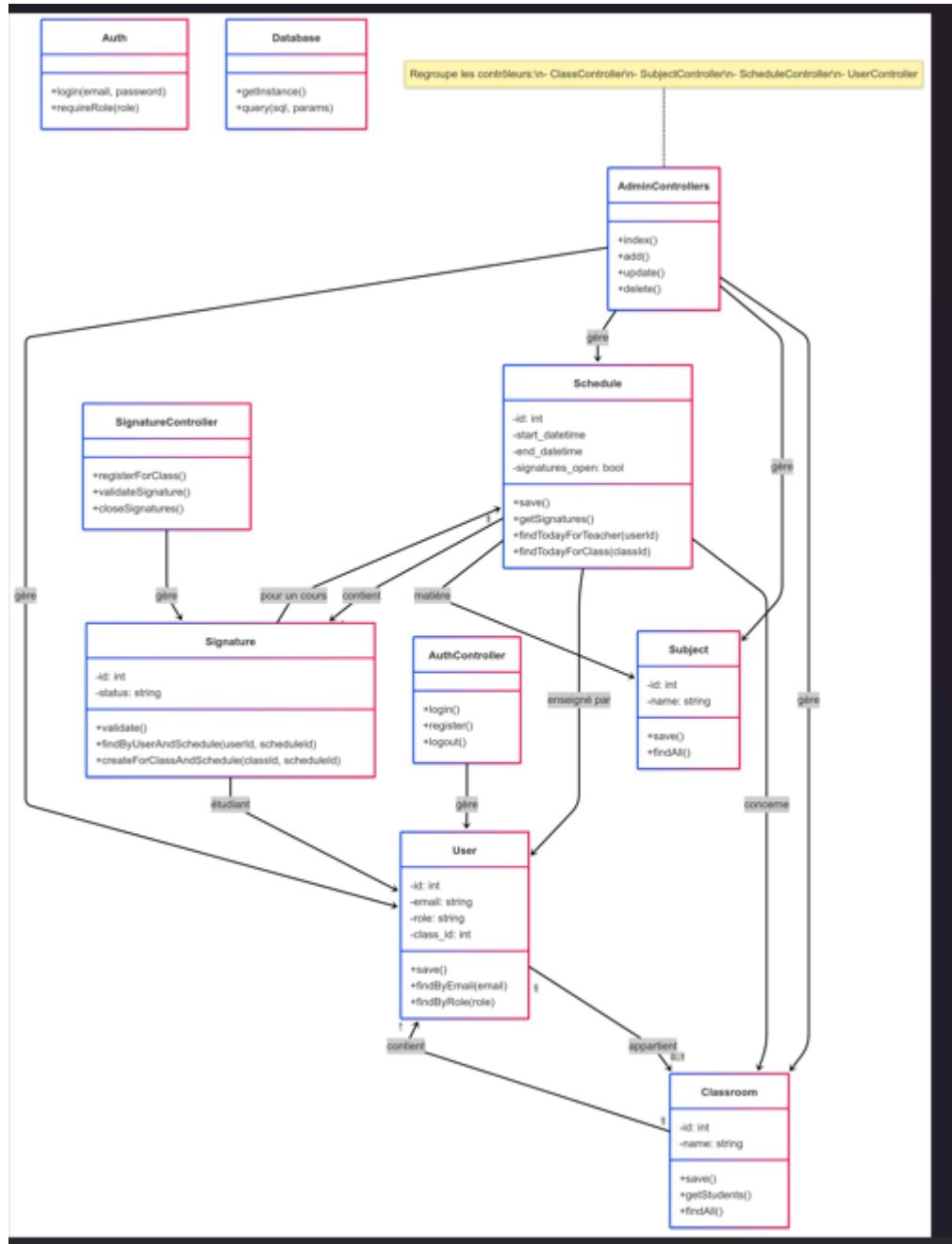
## Annexe 5 - Diagramme de context



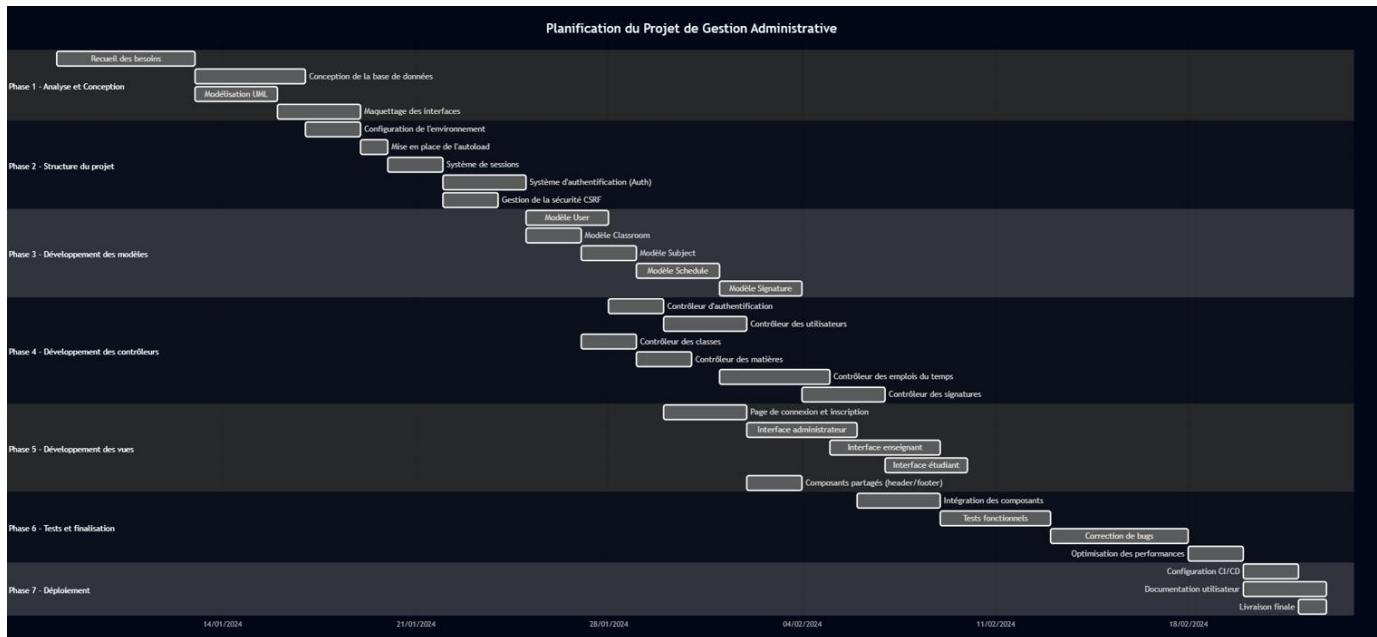
## Annexe 6 - Diagramme de package



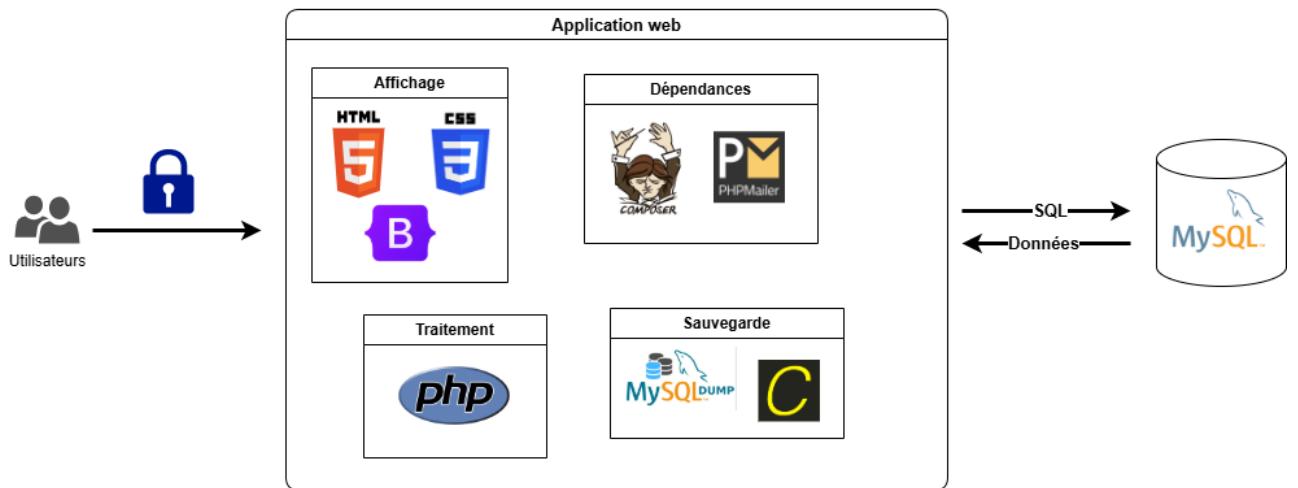
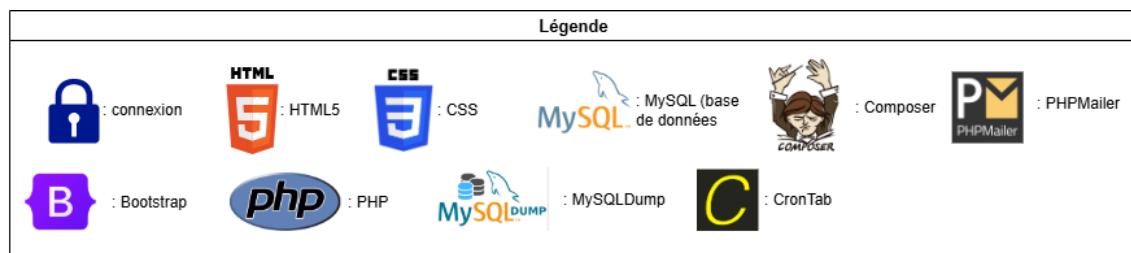
## Annexe 7 - Diagramme de classes



## Annexe 8 - Diagramme de Gantt



## Annexe 9 - Cartographie technique



## ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle (recto)

## Épreuve E6 - Conception et développement d'applications (option SLAM)

<b>DESCRIPTION D'UNE RÉALISATION PROFESSIONNELLE</b>		<b>N° réalisation :</b>
<b>Nom, prénom :</b> Chevrollier Nathan		<b>N° candidat :</b> 02046697332
<b>Épreuve ponctuelle</b> <input checked="" type="checkbox"/> <b>Contrôle en cours de formation</b> <input type="checkbox"/>		<b>Date :</b> ..24 / ..04 / .2025...
<b>Organisation support de la réalisation professionnelle</b> Projet fictif pédagogique encadré		
<b>Intitulé de la réalisation professionnelle</b> ClicknEat — Application de réservation et de commande en ligne pour restaurants		
<b>Période de réalisation :</b> Sept 2024 - avril 2025. <b>Lieu :</b> Angers		
<b>Modalité :</b> <input checked="" type="checkbox"/> <b>Seul(e)</b> <input type="checkbox"/> <b>En équipe</b>		
<b>Compétences travaillées</b> <ul style="list-style-type: none"> <li><input checked="" type="checkbox"/> Concevoir et développer une solution applicative</li> <li><input checked="" type="checkbox"/> Assurer la maintenance corrective ou évolutive d'une solution applicative</li> <li><input checked="" type="checkbox"/> Gérer les données</li> </ul>		
<b>Conditions de réalisation<sup>5</sup> (ressources fournies, résultats attendus)</b> <p>Réalisé dans le cadre de la formation, cette application repose sur un besoin fictif défini pour une startup : une solution de gestion des restaurants permettant la réservation et la commande anticipée en ligne, afin de réduire l'attente en salle. J'ai rédigé le cahier des charges en tenant compte des exigences pédagogiques, guidé par mon enseignant pour définir les objectifs. Résultats attendus : une application web fonctionnelle développée en PHP avec le framework Laravel, une base de données MySQL, une interface utilisateur par rôle (client, restaurateur), permettant la gestion des menus, des commandes, des réservations, ainsi qu'une documentation technique décrivant le fonctionnement de l'application.</p>		
<b>Description des ressources documentaires, matérielles et logicielles utilisées<sup>6</sup></b> <p>Documentaires : Cahier des charges, documentation Laravel            Matérielles : PC sous Windows 11</p> <p>Logiciels :</p> <ul style="list-style-type: none"> <li>- Laravel Breeze (template d'authentification)</li> <li>- Composer, Artisan, Faker, Blade (outils Laravel)</li> <li>- phpMyAdmin (gestionnaire de base de données)</li> <li>- Git et GitHub (versioning et CI/CD)</li> <li>- PEST (framework de tests unitaires)</li> </ul> <p>Langages : HTML, CSS (Bootstrap), PHP, SQL, Blade</p>		
<b>Modalités d'accès aux productions<sup>7</sup> et à leur documentation<sup>8</sup></b> <p>Application : <a href="https://clickneat.chevrolliernathan.fr">https://clickneat.chevrolliernathan.fr</a></p> <p>Identifiants :</p> <ul style="list-style-type: none"> <li>-admin : admin@clickneat.com / mdp : password</li> <li>-restaurateur : restau@clickneat.com / mdp : password</li> <li>-client : client@clickneat.com / mdp : password</li> </ul> <p>code source et documentation:  <a href="https://github.com/NathanChevrollier/ClicknEat">https://github.com/NathanChevrollier/ClicknEat</a></p>		

<sup>5</sup> En référence aux *conditions de réalisation et ressources nécessaires* du bloc « Conception et développement d'applications » prévues dans le référentiel de certification du BTS SIO.

<sup>6</sup> Les réalisations professionnelles sont élaborées dans un environnement technologique conforme à l'annexe II.E du référentiel du BTS SIO.

<sup>7</sup> Conformément au référentiel du BTS SIO « *Dans tous les cas, les candidats doivent se munir des outils et ressources techniques nécessaires au déroulement de l'épreuve. Ils sont seuls responsables de la disponibilité et de la mise en œuvre de ces outils et ressources. La circulaire nationale d'organisation précise les conditions matérielles de déroulement des interrogations et les pénalités à appliquer aux candidats qui ne se seraient pas munis des éléments nécessaires au déroulement de l'épreuve.* ». Les éléments peuvent être un identifiant, un mot de passe, une adresse réticulaire (URL) d'un espace de stockage et de la présentation de l'organisation du stockage.

<sup>8</sup> Lien vers la documentation complète, précisant et décrivant, si cela n'a été fait au verso de la fiche, la réalisation professionnelle, par exemples service fourni par la réalisation, interfaces utilisateurs, description des classes ou de la base de données.

**ANNEXE 9-1-B : Fiche descriptive de réalisation professionnelle  
(verso, éventuellement pages suivantes)****Épreuve E6 - Conception et développement d'applications (option SLAM)****Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs**

Contexte :

ClicknEat est une application web destinée aux restaurateurs afin d'optimiser la gestion des commandes en ligne, via réservation anticipée.

L'utilisateur commande à l'avance pour éviter toute attente au restaurant.

Le projet s'inscrit dans une logique de MVP avec évolutions prévues vers une version MLP.

Fonctionnalités développées :

- Authentification et gestion des rôles (clients / restaurateurs)
- Gestion de plusieurs restaurants (espace dédié)
- Création de menus personnalisés avec catégories d'articles
- Prise de commande en ligne et réservation de tables
- Application responsive
- Test unitaire du code via PEST
- Sauvegarde, migration et optimisation automatisée via Artisan

Contraintes techniques :

- Utilisation du framework Laravel avec Breeze
- Déploiement sur nom de domaine sécurisé (SSL)
- Intégration continue via GitHub Actions
- ORM : Eloquent (modèle relationnel)
- Tests via PEST
- Gestion du cache, des assets compilés, et des migrations via commandes artisan

Tâches réalisées :

- Rédaction du cahier des charges
- Conception et création de la base de données (modélisation Merise)
- Conception de la cartographie technique
- Développement du back-end Laravel + front-end Blade
- Intégration de Breeze, PEST
- Déploiement sur VPS avec certificat SSL
- Rédaction de la documentation technique

Documentation jointe :  
- Cahier des charges  
- Cartographie technique  
- Diagramme de GANTT  
- Modélisation de la BDD

**ANNEXE 10-B : Modèle d'attestation de respect de l'annexe II.E – Environnement technologique pour la certification du référentiel**

**Épreuve E6 - Conception et développement d'applications (option SLAM)**

**CONTRÔLE DE L'ENVIRONNEMENT TECHNOLOGIQUE**

**En référence à l'annexe II.E –« Environnement technologique pour la certification » du référentiel du BTS SIO**

<b>Identification<sup>2</sup></b>		<b>SLAM</b>
-----------------------------------	--	-------------

**1. Environnement commun aux deux options**

**1.1 L'environnement technologique supportant le système d'information de l'organisation cliente comporte au moins :**

<b>Éléments</b>	<b>Description de l'implantation dans le centre d'examen (nom du service ou de l'outil et caractéristiques techniques)</b>	<b>Remarques de la commission d'interrogation</b>
Un service d'authentification	Auth laravel + Auth PHP natif	
Un SGBD	MariaDB = MySQL - phpMyAdmin	
Un accès sécurisé à internet	certificats SSL (let'sencrypt) sur vos VPS	
Un environnement de travail collaboratif	git & github VSCode (outils / plugins, etc...)	
Deux serveurs, éventuellement virtualisés, basés sur des systèmes d'exploitation différents, dont l'un est un logiciel libre ( <i>open source</i> )	VPS sur Debian + serveur de développement sur Windows (mamp / wamp / xampp)	

<sup>2</sup> Nom et adresse du centre d'examen ou identification de la personne candidate individuelle (numéro, nom, prénom)

**ANNEXE 10-B (suite) : Modèle d'attestation de respect de l'annexe II.E – Environnement technologique pour la certification du référentiel**  
**Épreuve E6 - Conception et développement d'applications (option SLAM)**

<b>Éléments</b>	<b>Description de l'implantation dans le centre d'examen (nom du service ou de l'outil et caractéristiques techniques)</b>	<b>Remarques de la commission d'interrogation</b>
Une solution de sauvegarde	GIT - dump SQL	
Des ressources dont l'accès est sécurisé et soumis à habilitation	GIT(hub) - Gestion de rôle dans les projets de dev	
Deux types de terminaux dont un mobile (type <i>smartphone</i> ou encore tablette)	Web - Responsive (RWD) - electron desktop (calculatrice) - Script SHELL	

**1.2 Des outils sont mobilisés pour la gestion de la sécurité :**

<b>Éléments</b>	<b>Description de l'implantation dans le centre d'examen (nom du service ou de l'outil et caractéristiques techniques)</b>	<b>Remarques de la commission d'interrogation</b>
Gestion des incidents	formulaire de contact support logs (error & access) VPS	
Détection et prévention des intrusions	Monitoring & services OVH (anti-DDoS automatique) - IPTABLE - Fail2ban (optionnel) - changement des ports par défaut - désactivation du root - LOGS	
Chiffrement	HTTPS (SSL) - hash password en BDD	
Analyse de trafic	Monitoring OVH - Google Analytics	

**Remarque : les logiciels de simulation ou d'émulation sont utilisés en réponse à des besoins de l'organisation. Ils ne peuvent se substituer complètement à des équipements réels dans l'environnement technologique d'apprentissage.**

**ANNEXE 10-B (suite) : Modèle d'attestation de respect de l'annexe II.E – Environnement technologique pour la certification du référentiel**  
**Épreuve E6 - Conception et développement d'applications (option SLAM)**

**2. Savoirs spécifiques à l'option « solutions logicielles et applications métiers » (SLAM)**

**2.1 L'environnement technologique supportant le système d'information de l'organisation cliente comporte au moins :**

<b>Éléments</b>	<b>Description de l'implantation dans le centre d'examen (nom du service ou de l'outil et caractéristiques techniques)</b>	<b>Remarques de la commission d'interrogation</b>
Un ou deux environnements de développement disposant d'outils de gestion de tests et supportant un cadre applicatif ( <i>framework</i> ) et au moins deux langages	HTML/CSS PHP SQL Javascript - PEST pour les tests, Laravel	
Une bibliothèque de composants logiciels	Bootstrap - calculette immo : JSPDF (un truc du genre) - JSCalendar / Fullcalendar	
Un SGBD avec langage de programmation associé	MariaDB MySQL	
Un logiciel de gestion de versions et de suivi de problèmes d'ordre logiciel	GIT & Github - tests unitaires - Github ACTION	
Une solution permettant de tester les comportements anormaux d'une application	Tests fonctionnels / unitaires @CSRF, failles de sécurité, protection injection SQL/XSS	

**2.2 Les activités de l'organisation cliente s'appuient sur aux moins deux solutions applicatives opérationnelles permettant d'offrir un accès sécurisé à des données hébergées sur un site distant. Au sein des architectures de ces solutions applicatives doivent figurer l'exploitation de mécanismes d'appel à des services applicatifs distants et au moins trois des situations ci-dessous :**

<b>Éléments</b>	<b>Description de l'implantation dans le centre d'examen (nom du service ou de l'outil et caractéristiques techniques)</b>	<b>Remarques de la commission d'interrogation</b>
Du code exécuté sur le système d'exploitation d'une solution technique d'accès fixe (type client lourd)	Calculette Electron	
Du code exécuté dans un navigateur Web (type client léger ou riche)	PHP + Laravel	
Du code exécuté sur le système d'exploitation d'une solution technique d'accès mobile		
Du code exécuté sur le système d'exploitation d'un serveur	Script Serveur shell	

**2.3 Une solution applicative peut être issue d'un développement spécifique ou de la modification du code d'un logiciel notamment open source.**

**2.4 Les solutions applicatives présentes dans le contexte sont opérationnelles et leur code source est accessible dans un environnement de développement opérationnel au moment de l'épreuve.**

# Annexes du projet Click'n Eat

---

Toute la documentation du projet est aussi disponible avec les fichiers originels sur le repo github correspondant au projet : [Cliquez ici](#)

(<https://github.com/NathanChevrollier/ClicknEat>)

## Sommaire

- **Annexe 0** – Cahier des charges
- **Annexe 1** – Dictionnaire de données
- **Annexe 2** – MCD (Modèle Conceptuel de Données)
- **Annexe 3** -
- **Annexe 4** – MPD (Modèle Physique de Données)
- **Annexe 5** – Diagramme de contexte
- **Annexe 6** – Diagramme de package
- **Annexe 7** – Diagramme de classes
- **Annexe 8** – Diagramme de Gantt
- **Annexe 9** – Cartographie technique

## **Annexe 0 - Cahier des charges**

Click'n Eat - Laravel

Cahier des charges

Contexte

La startup Click'n Eat nous mandate pour réaliser une application de réservation de restaurant avec prise de commande au préalable, comme un "restaurant sans attente". Le repas est donc prêt dès notre arrivée.

Objectif

Proposer une solution optimisée pour plusieurs restaurants qui proposent des menus différents. Optimiser les échanges entre les différents acteurs du restaurant (administration, service, cuisine, clients).

Public cible

Restaurateurs.

Expression des besoins

Une gestion complète et optimisée des restaurants, permettant de suivre en temps réel les menus, les commandes et les réservations. L'ensemble des opérations est parfaitement synchronisé avec les horaires d'ouverture, assurant une expérience fluide et sur-mesure pour chaque client. L'application Click'n Eat permet de répondre à ces besoins en offrant les outils nécessaires à l'ajustement des offres, à la gestion des réservations et au suivi des commandes, accessibles en ligne pour un pilotage efficace des établissements à tout moment.

Fonctionnalités MVP (Minimum Viable Product)

Gestion des restaurants

Authentification des utilisateurs

Accès spécifique restaurant

Accès spécifique client

Gestion de la carte/des menus (classification des articles au menu par catégorie)

Gestion des commandes

Fonctionnalités MLP (Minimum Lovable Product)

Génération d'un lien d'accès à la commande de chaque carte de restaurant via un QR code

Responsive

Paramétrage de la charte graphique du restaurant pour sa carte

Paiement en ligne via API Stripe (avantages : package Laravel cashier, voir doc)

Contraintes techniques

Laravel

Déploiement sur un nom de domaine

Intégration continue

intégrer automatique les migrations de BDD

gérer le vide des caches de Laravel,

gérer l'optimisation des fichiers de production

Template

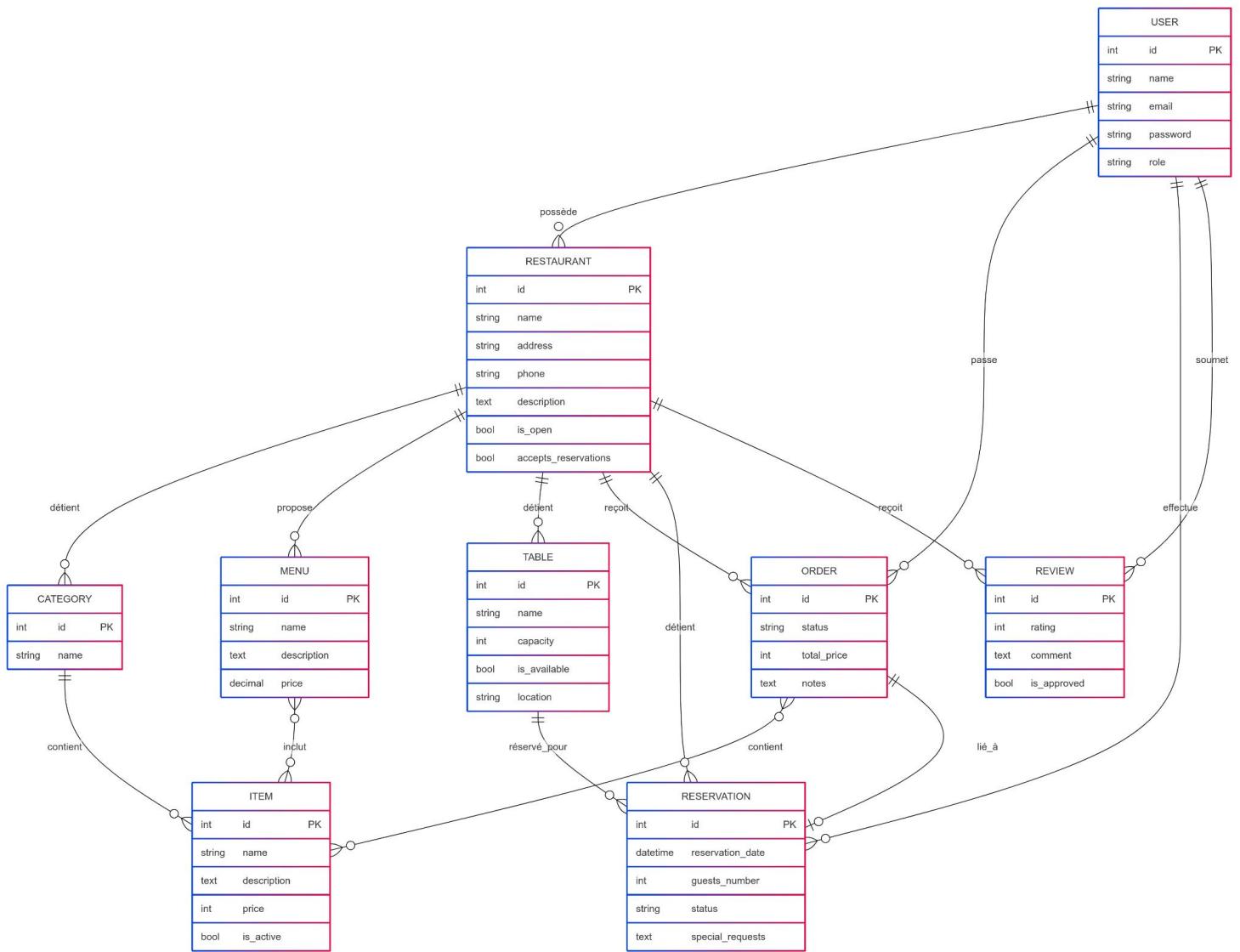
Certificat SSL (HTTPs)

Tests unitaires : code qui teste le code

## Annexe 1 – Dictionnaire de données

Column1	Column2	Column3	Column4	Column5
Table	Champ	Type	Description	Contrainte
Users	<code>id</code>	<code>BIGINT</code>	Identifiant unique	<code>PK AUTO_INCREMENT</code>
Users	<code>name</code>	<code>VARCHAR(255)</code>	Nom de l'utilisateur	<code>NOT NULL</code>
Users	<code>email</code>	<code>VARCHAR(255)</code>	Adresse email de l'utilisateur	<code>NOT NULL UNIQUE</code>
Users	<code>email_verified_at</code>	<code>TIMESTAMP</code>	Date de vérification de l'email	<code>NULL</code>
Users	<code>password</code>	<code>VARCHAR(255)</code>	Mot de passe hashé	<code>NOT NULL</code>
Users	<code>role</code>	<code>VARCHAR(255)</code>	Rôle de l'utilisateur (client restaurateur admin)	<code>NOT NULL DEFAULT 'client'</code>
Users	<code>remember_token</code>	<code>VARCHAR(100)</code>	Token pour la fonctionnalité "Se souvenir de moi"	<code>NULL</code>
Users	<code>created_at</code>	<code>TIMESTAMP</code>	Date de création	<code>NULL</code>
Users	<code>updated_at</code>	<code>TIMESTAMP</code>	Date de dernière modification	<code>NULL</code>
<hr/>				
Restaurants	<code>id</code>	<code>BIGINT</code>	Identifiant unique	<code>PK AUTO_INCREMENT</code>
Restaurants	<code>user_id</code>	<code>BIGINT</code>	Identifiant du propriétaire (restaurateur)	<code>FK NULL</code>
Restaurants	<code>name</code>	<code>VARCHAR(255)</code>	Nom du restaurant	<code>NOT NULL</code>
Restaurants	<code>address</code>	<code>VARCHAR(255)</code>	Adresse du restaurant	<code>NULL</code>
Restaurants	<code>phone</code>	<code>VARCHAR(20)</code>	Numéro de téléphone	<code>NULL</code>
Restaurants	<code>description</code>	<code>TEXT</code>	Description du restaurant	<code>NULL</code>
Restaurants	<code>is_open</code>	<code>BOOLEAN</code>	Indique si le restaurant est ouvert	<code>DEFAULT TRUE</code>
Restaurants	<code>accepts_reservations</code>	<code>BOOLEAN</code>	Indique si le restaurant accepte les réservations	<code>DEFAULT TRUE</code>
Restaurants	<code>created_at</code>	<code>TIMESTAMP</code>	Date de création	<code>NULL</code>
Restaurants	<code>updated_at</code>	<code>TIMESTAMP</code>	Date de dernière modification	<code>NULL</code>
<hr/>				
Categories	<code>id</code>	<code>BIGINT</code>	Identifiant unique	<code>PK AUTO_INCREMENT</code>
Categories	<code>restaurant_id</code>	<code>BIGINT</code>	Identifiant du restaurant	<code>FK NOT NULL</code>
Categories	<code>name</code>	<code>VARCHAR(255)</code>	Nom de la catégorie	<code>NOT NULL</code>
Categories	<code>created_at</code>	<code>TIMESTAMP</code>	Date de création	<code>NULL</code>
Categories	<code>updated_at</code>	<code>TIMESTAMP</code>	Date de dernière modification	<code>NULL</code>
<hr/>				
Items	<code>id</code>	<code>BIGINT</code>	Identifiant unique	<code>PK AUTO_INCREMENT</code>
Items	<code>category_id</code>	<code>BIGINT</code>	Identifiant de la catégorie	<code>FK NOT NULL</code>
Items	<code>name</code>	<code>VARCHAR(255)</code>	Nom du plat	<code>NOT NULL</code>
Items	<code>description</code>	<code>TEXT</code>	Description du plat	<code>NULL</code>
Items	<code>cost</code>	<code>INTEGER</code>	Coût de revient (en centimes)	<code>NULL</code>
Items	<code>price</code>	<code>INTEGER</code>	Prix de vente (en centimes)	<code>NOT NULL</code>
Items	<code>is_active</code>	<code>BOOLEAN</code>	Disponibilité du plat	<code>DEFAULT TRUE</code>
Items	<code>created_at</code>	<code>TIMESTAMP</code>	Date de création	<code>NULL</code>
Items	<code>updated_at</code>	<code>TIMESTAMP</code>	Date de dernière modification	<code>NULL</code>
<hr/>				
Menus	<code>id</code>	<code>BIGINT</code>	Identifiant unique	<code>PK AUTO_INCREMENT</code>
Menus	<code>restaurant_id</code>	<code>BIGINT</code>	Identifiant du restaurant	<code>FK NOT NULL</code>
Menus	<code>name</code>	<code>VARCHAR(255)</code>	Nom du menu	<code>NOT NULL</code>
Menus	<code>description</code>	<code>TEXT</code>	Description du menu	<code>NULL</code>
Menus	<code>price</code>	<code>DECIMAL(8,2)</code>	Prix du menu	<code>NOT NULL</code>
Menus	<code>created_at</code>	<code>TIMESTAMP</code>	Date de création	<code>NULL</code>
Menus	<code>updated_at</code>	<code>TIMESTAMP</code>	Date de dernière modification	<code>NULL</code>
<hr/>				
Menu_Items	<code>id</code>	<code>BIGINT</code>	Identifiant unique	<code>PK AUTO_INCREMENT</code>
Menu_Items	<code>menu_id</code>	<code>BIGINT</code>	Identifiant du menu	<code>FK NOT NULL</code>
Menu_Items	<code>item_id</code>	<code>BIGINT</code>	Identifiant du plat	<code>FK NOT NULL</code>
Menu_Items	<code>created_at</code>	<code>TIMESTAMP</code>	Date de création	<code>NULL</code>
Menu_Items	<code>updated_at</code>	<code>TIMESTAMP</code>	Date de dernière modification	<code>NULL</code>
<hr/>				
Tables	<code>id</code>	<code>BIGINT</code>	Identifiant unique	<code>PK AUTO_INCREMENT</code>
Tables	<code>restaurant_id</code>	<code>BIGINT</code>	Identifiant du restaurant	<code>FK NOT NULL</code>
Tables	<code>name</code>	<code>VARCHAR(255)</code>	Nom/numéro de la table	<code>NOT NULL</code>
Tables	<code>capacity</code>	<code>INTEGER</code>	Capacité en nombre de personnes	<code>NOT NULL</code>
Tables	<code>is_available</code>	<code>BOOLEAN</code>	Disponibilité de la table	<code>DEFAULT TRUE</code>
Tables	<code>location</code>	<code>VARCHAR(255)</code>	Emplacement dans le restaurant	<code>NULL</code>
Tables	<code>description</code>	<code>TEXT</code>	Description de la table	<code>NULL</code>
Tables	<code>created_at</code>	<code>TIMESTAMP</code>	Date de création	<code>NULL</code>
Tables	<code>updated_at</code>	<code>TIMESTAMP</code>	Date de dernière modification	<code>NULL</code>
<hr/>				
Orders	<code>id</code>	<code>BIGINT</code>	Identifiant unique	<code>PK AUTO_INCREMENT</code>
Orders	<code>user_id</code>	<code>BIGINT</code>	Identifiant du client	<code>FK NOT NULL</code>
Orders	<code>restaurant_id</code>	<code>BIGINT</code>	Identifiant du restaurant	<code>FK NOT NULL</code>
Orders	<code>status</code>	<code>VARCHAR(255)</code>	Statut de la commande (pending confirmed preparing ready completed cancelled)	<code>DEFAULT 'pending'</code>
Orders	<code>total_price</code>	<code>INTEGER</code>	Montant total (en centimes)	<code>DEFAULT 0</code>
Orders	<code>notes</code>	<code>TEXT</code>	Notes sur la commande	<code>NULL</code>
Orders	<code>created_at</code>	<code>TIMESTAMP</code>	Date de création	<code>NULL</code>
Orders	<code>updated_at</code>	<code>TIMESTAMP</code>	Date de dernière modification	<code>NULL</code>
<hr/>				
Order_Items	<code>id</code>	<code>BIGINT</code>	Identifiant unique	<code>PK AUTO_INCREMENT</code>
Order_Items	<code>order_id</code>	<code>BIGINT</code>	Identifiant de la commande	<code>FK NOT NULL</code>
Order_Items	<code>item_id</code>	<code>BIGINT</code>	Identifiant du plat	<code>FK NOT NULL</code>
Order_Items	<code>quantity</code>	<code>INTEGER</code>	Quantité	<code>DEFAULT 1</code>
Order_Items	<code>price</code>	<code>INTEGER</code>	Prix unitaire au moment de la commande (en centimes)	<code>NOT NULL</code>
Order_Items	<code>created_at</code>	<code>TIMESTAMP</code>	Date de création	<code>NULL</code>
Order_Items	<code>updated_at</code>	<code>TIMESTAMP</code>	Date de dernière modification	<code>NULL</code>
<hr/>				
Reservations	<code>id</code>	<code>BIGINT</code>	Identifiant unique	<code>PK AUTO_INCREMENT</code>
Reservations	<code>user_id</code>	<code>BIGINT</code>	Identifiant du client	<code>FK NOT NULL</code>
Reservations	<code>restaurant_id</code>	<code>BIGINT</code>	Identifiant du restaurant	<code>FK NOT NULL</code>
Reservations	<code>table_id</code>	<code>BIGINT</code>	Identifiant de la table	<code>FK NULL</code>
Reservations	<code>order_id</code>	<code>BIGINT</code>	Identifiant de la commande associée	<code>FK NULL</code>
Reservations	<code>reservation_date</code>	<code>DATETIME</code>	Date et heure de la réservation	<code>NOT NULL</code>
Reservations	<code>guests_number</code>	<code>INTEGER</code>	Nombre de personnes	<code>NOT NULL</code>
Reservations	<code>status</code>	<code>VARCHAR(255)</code>	Statut de la réservation (pending confirmed cancelled completed)	<code>DEFAULT 'pending'</code>
Reservations	<code>special_requests</code>	<code>TEXT</code>	Demandes spéciales	<code>NULL</code>
Reservations	<code>created_at</code>	<code>TIMESTAMP</code>	Date de création	<code>NULL</code>
Reservations	<code>updated_at</code>	<code>TIMESTAMP</code>	Date de dernière modification	<code>NULL</code>
<hr/>				
Reviews	<code>id</code>	<code>BIGINT</code>	Identifiant unique	<code>PK AUTO_INCREMENT</code>
Reviews	<code>user_id</code>	<code>BIGINT</code>	Identifiant du client	<code>FK NOT NULL</code>
Reviews	<code>restaurant_id</code>	<code>BIGINT</code>	Identifiant du restaurant	<code>FK NOT NULL</code>
Reviews	<code>rating</code>	<code>INTEGER</code>	Note (1-5)	<code>NOT NULL</code>
Reviews	<code>comment</code>	<code>TEXT</code>	Commentaire	<code>NULL</code>
Reviews	<code>is_approved</code>	<code>BOOLEAN</code>	Indique si l'avis est approuvé	<code>DEFAULT TRUE</code>
Reviews	<code>created_at</code>	<code>TIMESTAMP</code>	Date de création	<code>NULL</code>
Reviews	<code>updated_at</code>	<code>TIMESTAMP</code>	Date de dernière modification	<code>NULL</code>

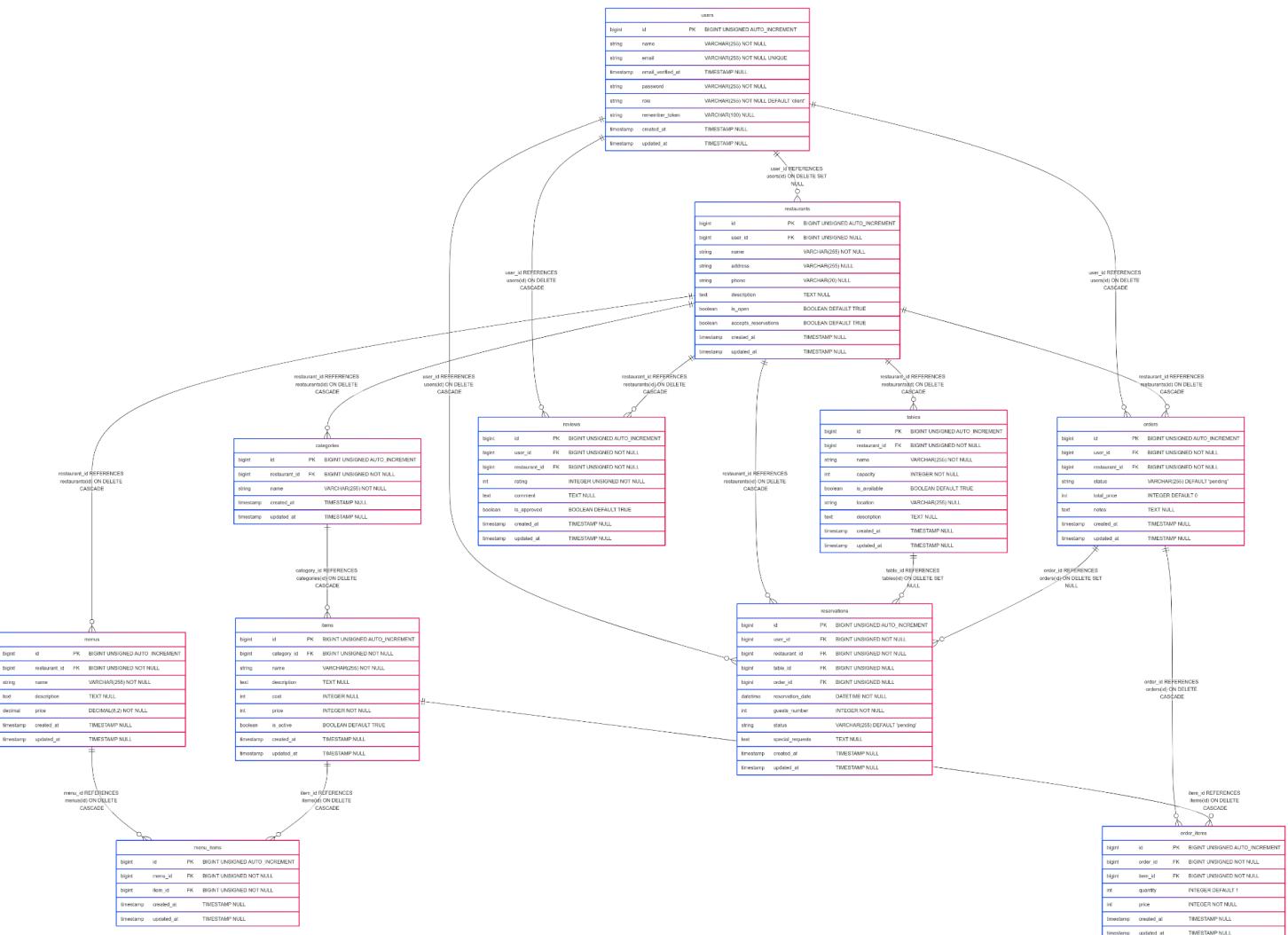
## Annexe 2 - MCD



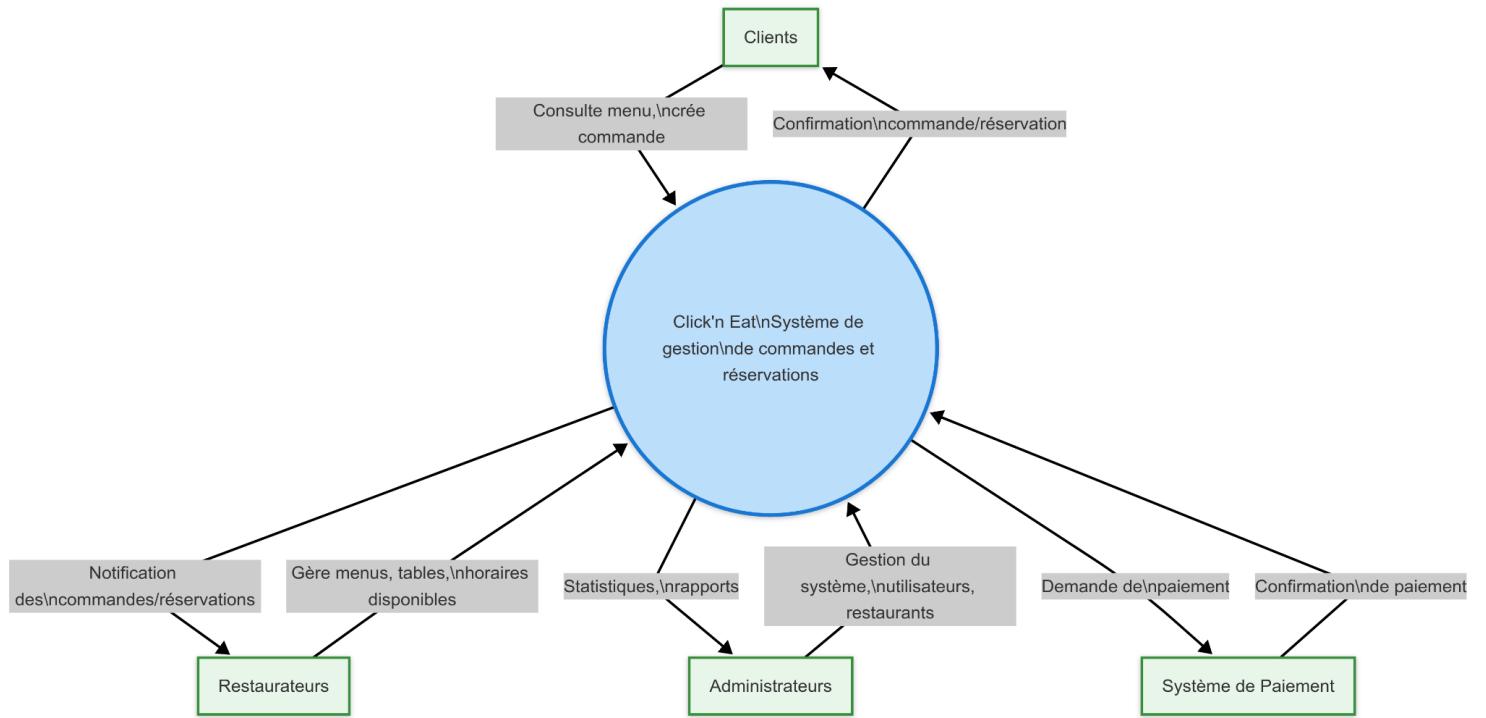
## Annexe 3 – MLD

Column1	Column2
Table	Attributs
users	id_PK name email password role email_verified_at remember_token created_at updated_at
restaurants	id_PK user_id_FK name address phone description is_open accepts_reservations created_at updated_at
categories	id_PK restaurant_id_FK name created_at updated_at
items	id_PK category_id_FK name description cost price is_active created_at updated_at
menus	id_PK restaurant_id_FK name description price created_at updated_at
menu_items	id_PK menu_id_FK item_id_FK created_at updated_at
tables	id_PK restaurant_id_FK name capacity is_available location description created_at updated_at
orders	id_PK user_id_FK restaurant_id_FK status total_price notes created_at updated_at
order_items	id_PK order_id_FK item_id_FK quantity price created_at updated_at
reservations	id_PK user_id_FK restaurant_id_FK table_id_FK order_id_FK reservation_date guests_number status special_requests created_at updated_at
reviews	id_PK user_id_FK restaurant_id_FK rating comment is_approved created_at updated_at

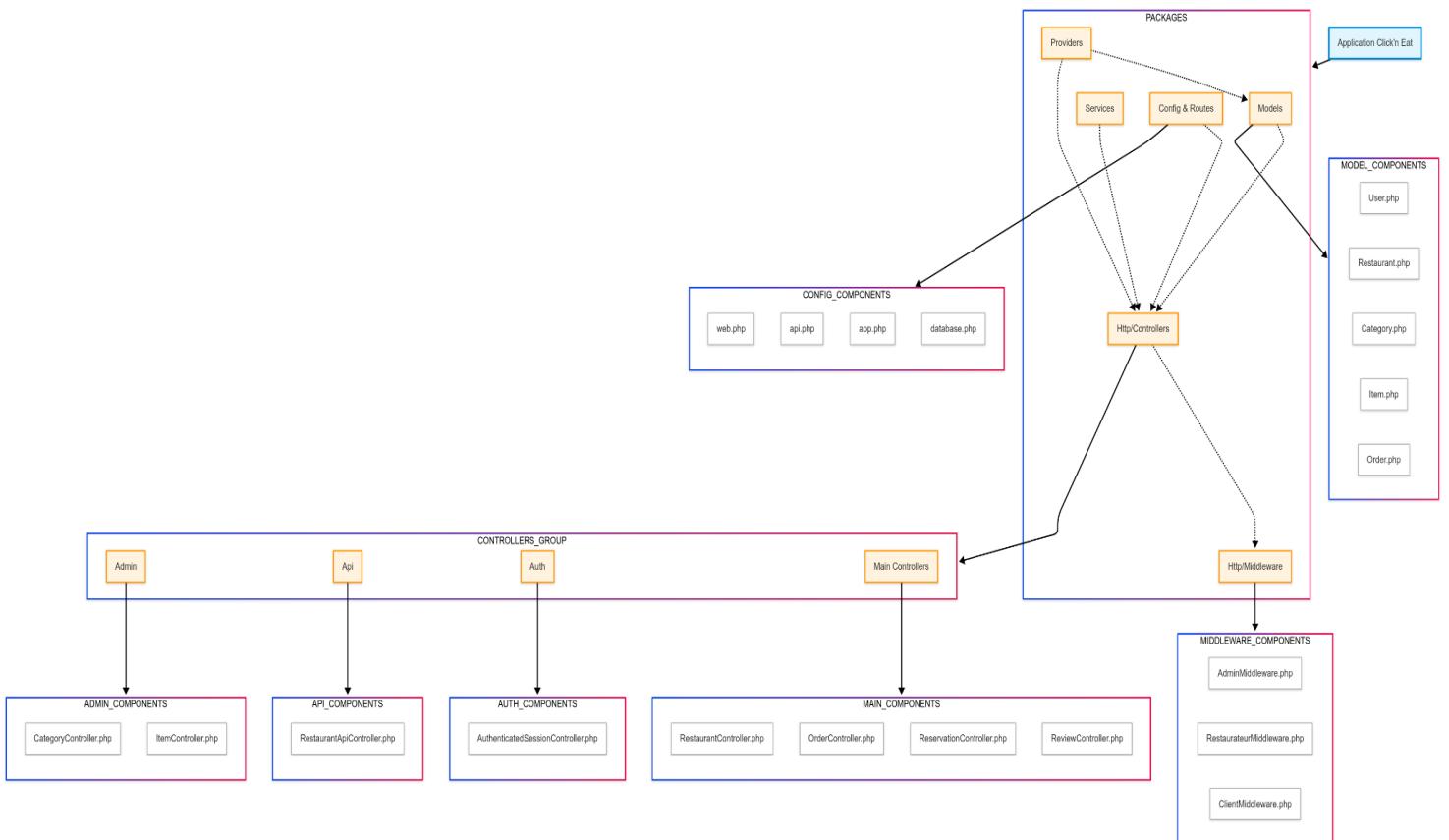
## Annexe 4 - MPD



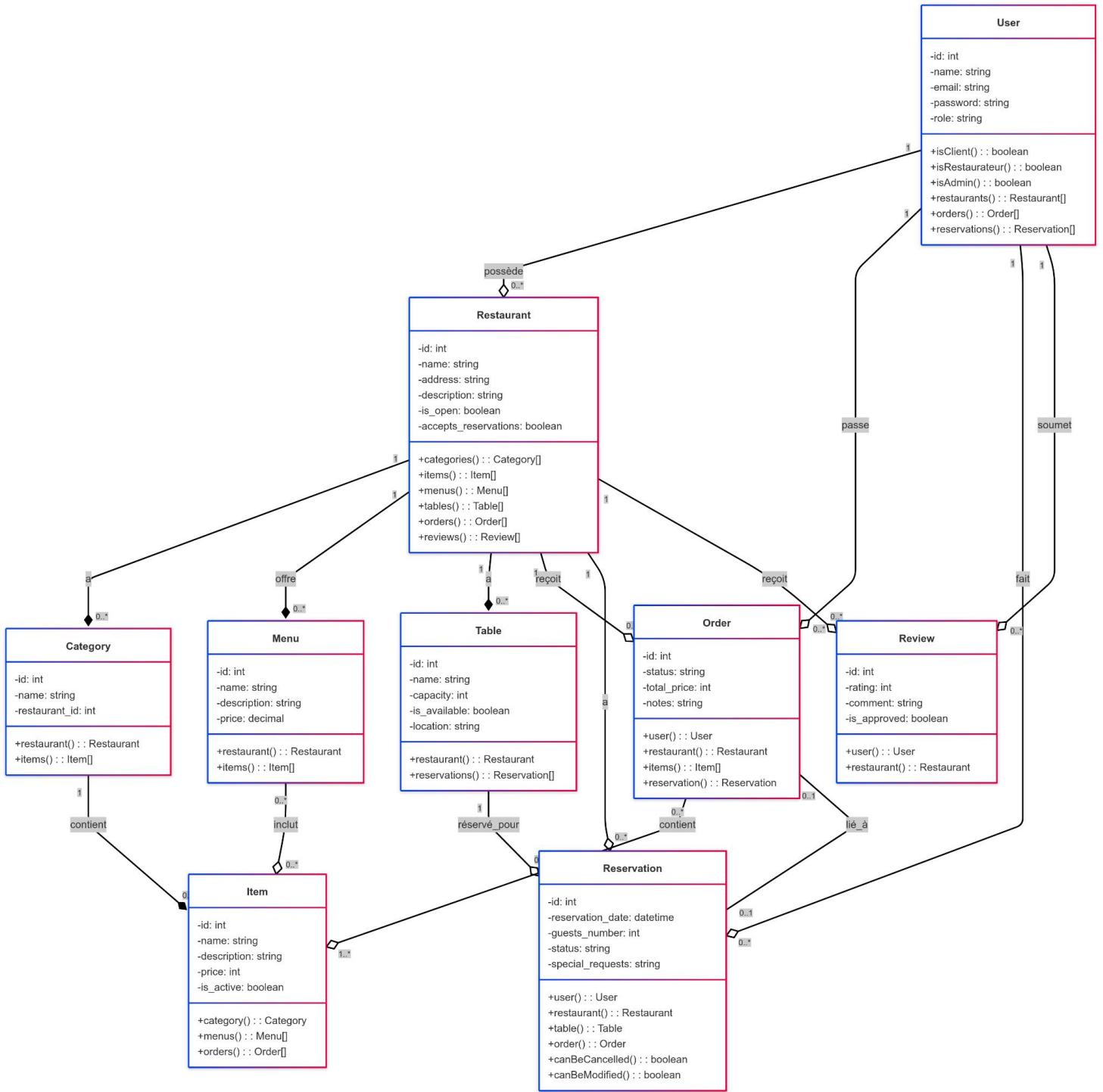
## Annexe 5 - Diagramme de contexte



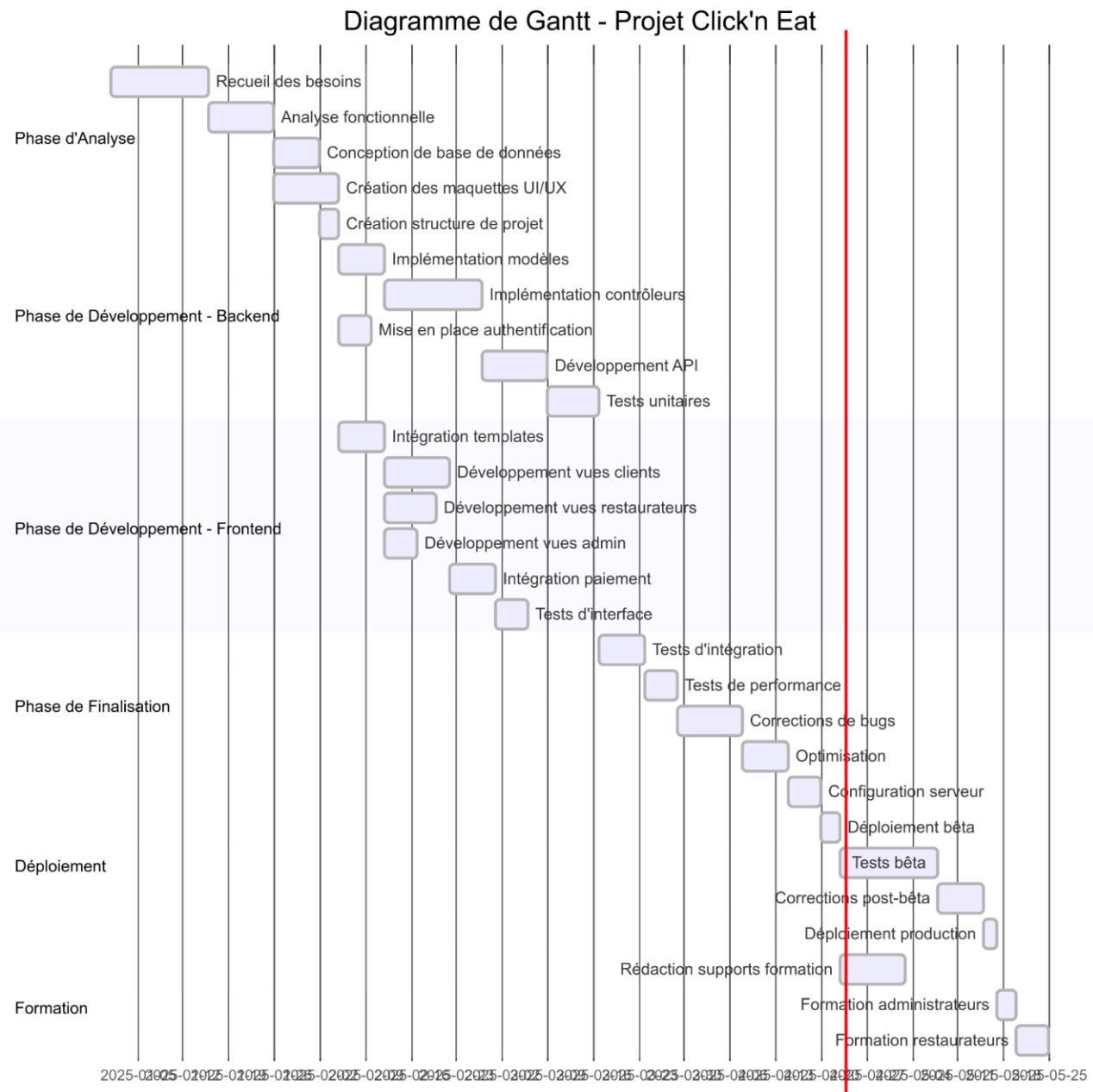
## Annexe 6 - Diagramme de package



## Annexe 7 - Diagramme de classes



## Annexe 8 - Diagramme de Gantt



## Annexe 9 - Cartographie technique

