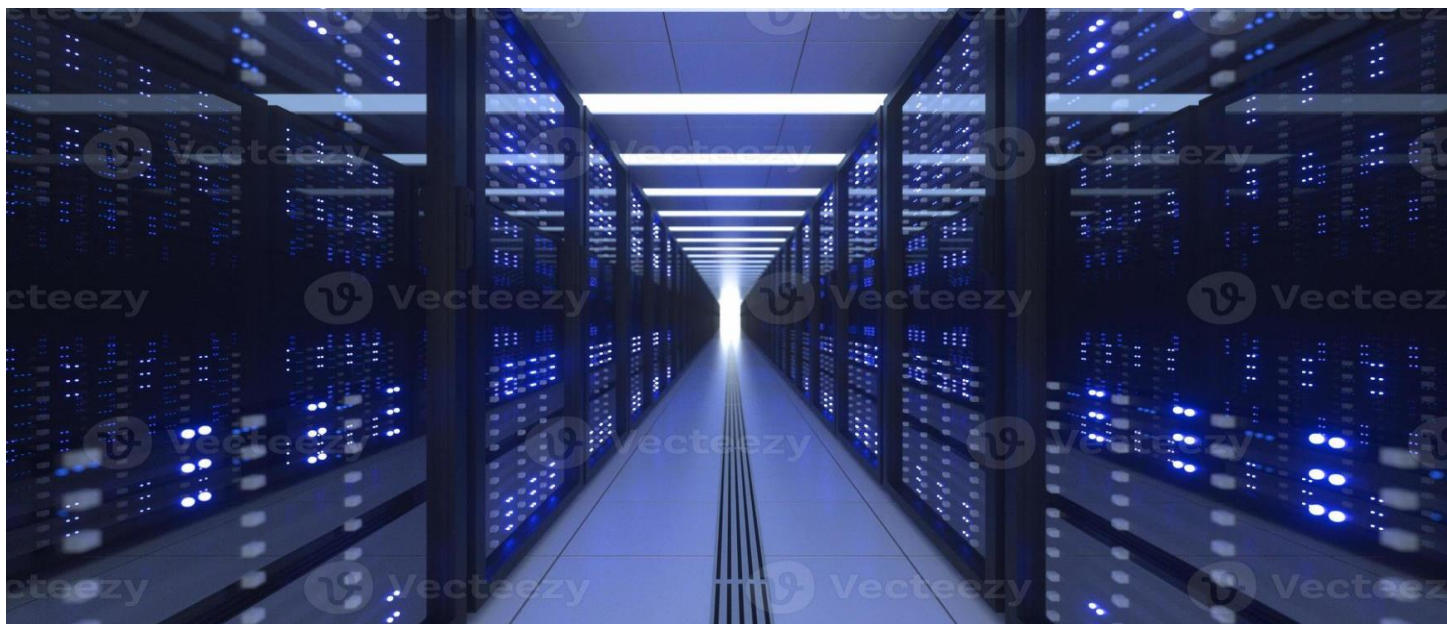


RAPPORT DE PROJET

Data Center Management System

Plateforme Web de Gestion des Ressources Informatiques



Réalisé par :

- CHAYMA OUALILI
- HALIMA ACHABBAK
- FATIMA BOUMEHAOUT
- FATIMA ZAHRAE EL BOUHSSINI
- OUARDA AIT EL FAKIH

Encadré par :

Mr. MHAMED AIT KBIR
Mr. YASYN EL YUSUFI

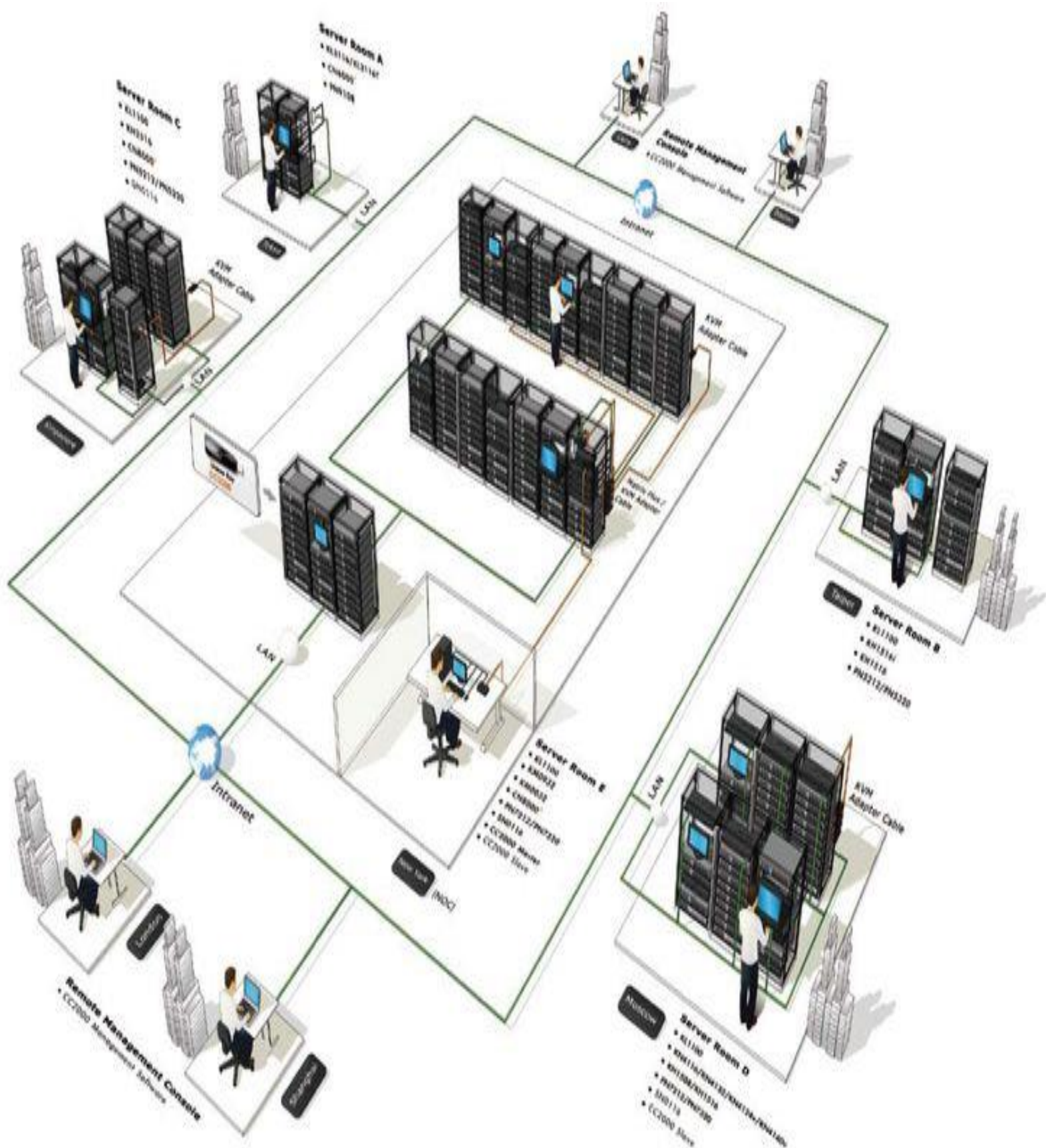


TABLE DES MATIÈRES

TABLE DES MATIÈRES	3
1. VUE D'ENSEMBLE DU PROJET	5
1.1 Objectifs du Projet	5
1.2 Diagramme ERD - Architecture de la Base de Données	5
1.3 Cahier des Charges	6
1.4 Technologies Utilisées	6
2. ARCHITECTURE GLOBALE DU SYSTÈME	7
2.1 Architecture en Couches	7
2.2 Modules du Système	7
3. MODULE DESIGN SYSTEM & UI	9
3.1 Rôle et Problématique	9
3.2 Palette Colorimétrique - Identité Elite Business	9
3.3 Principes de l'Interface	9
3.4 Composants Développés	9
3.5 Captures d'Écran	10
3.6 Compétences Développées	Error! Bookmark not defined.
3.7 Contact	Error! Bookmark not defined.
4. MODULE BASE DE DONNÉES & DASHBOARDS	11
4.1 Rôle et Problématique	11
4.2 Structure de la Base de Données	11
4.3 Fonctionnalités Implémentées	12
4.4 Dashboards par Rôle	13
4.5 Captures d'Écran des Dashboards	15
4.6 Structure des Fichiers	17
4.7 Métriques du Module	18
4.8 Commandes de Déploiement	18
4.9 Apprentissages et Défis	Error! Bookmark not defined.
5. MODULE AUTHENTIFICATION & SÉCURITÉ	19
5.1 Rôle et Problématique	19
5.2 Système d'Authentification Complet	19
5.3 Gestion Multi-Rôles	19
5.4 Matrice des Permissions Détaillée	19
5.5 Demandes de Compte	20
5.6 Sécurité Avancée Implémentée	20

5.8 Statistiques du Module	21
5.9 Compétences Développées.....	Error! Bookmark not defined.
5.10 Contact.....	Error! Bookmark not defined.
6. MODULE GESTION RESSOURCES & ANALYTICS	23
6.1 Rôle et Problématique	23
6.2 Gestion du Cycle de Vie des Ressources	23
6.3 Fonctionnalités de Gestion des Ressources	23
6.4 Maintenance & Incidents	23
6.5 Analytics & Statistiques Avancées.....	24
6.6 Captures d'Écran du Module	25
6.7 Niveaux d'Accès par Rôle.....	28
6.8 Interfaces Développées.....	28
6.9 Structure du Code	28
6.10 Compétences Développées.....	Error! Bookmark not defined.
6.11 Contact.....	Error! Bookmark not defined.
7. MODULE RÉSERVATIONS.....	29
7.1 Aperçu du Module	Error! Bookmark not defined.
7.2 Fonctionnalités Prévués	Error! Bookmark not defined.
7.3 Intégration avec les Autres Modules	Error! Bookmark not defined.
8. INTÉGRATION ET WORKFLOW GLOBAL.....	32
8.1 Workflow de Collaboration	Error! Bookmark not defined.
8.2 Gestion de Version Git.....	34
9. DÉPLOIEMENT ET CONFIGURATION	34
9.1 Prérequis Système	34
9.2 Installation en 6 Étapes	34
9.3 Comptes de Test	35
10. TESTS ET VALIDATION.....	35
10.1 Validation du Cahier des Charges	35
11. STATISTIQUES GLOBALES DU PROJET	36
11.1 Métriques de Développement.....	36
12. CONCLUSION	37
12.1 Objectifs Atteints.....	Error! Bookmark not defined.
12.2 Perspectives.....	37

1. VUE D'ENSEMBLE DU PROJET

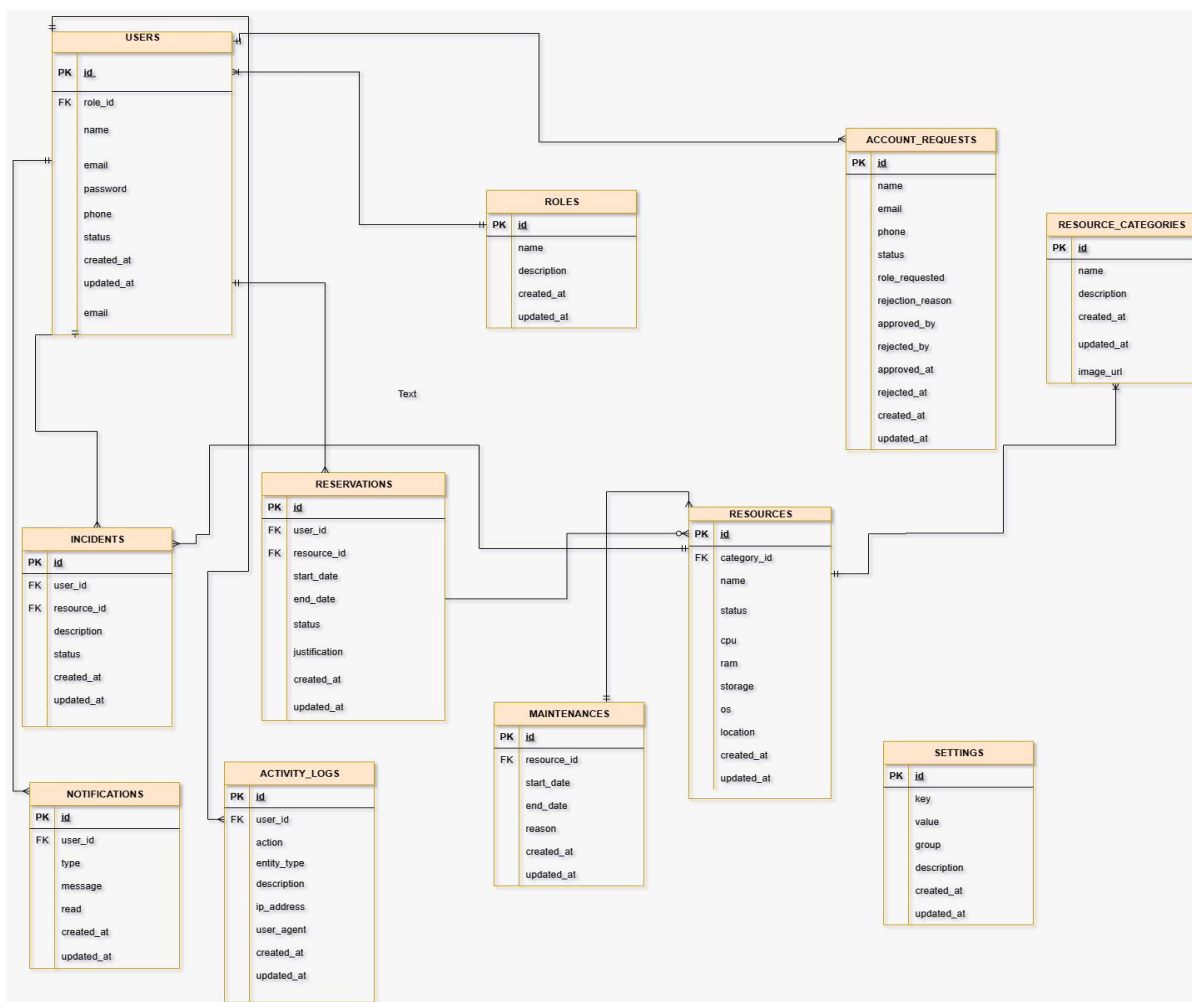
Le Data Center Management System est une application web complète développée avec Laravel et MySQL, conçue pour gérer efficacement les ressources informatiques d'un Data Center (serveurs, machines virtuelles, stockage, équipements réseau).

1.1 Objectifs du Projet

- Assurer une gestion transparente et efficace des ressources informatiques
- Permettre la réservation et l'allocation des équipements en temps réel
- Implémenter un système de permissions avec 4 profils utilisateurs distincts
- Fournir des statistiques et un suivi détaillé des réservations
- Garantir la sécurité et la traçabilité de toutes les opérations

1.2 Diagramme ERD - Architecture de la Base de Données

Le système repose sur une base de données relationnelle MySQL avec 13 tables interconnectées. Voici le diagramme entité-relation complet :



1.3 Cahier des Charges

L'application répond aux exigences suivantes :

- Base de données relationnelle MySQL avec modélisation complète
- Interface web ergonomique, sécurisée et responsive (sans frameworks CSS)
- Système d'authentification Laravel avec middleware
- Gestion multi-rôles : Invité, Utilisateur, Tech Manager, Administrateur
- Module de réservation avec gestion des conflits
- Système de notifications et de traçabilité
- Tableaux de bord avec statistiques graphiques (Chart.js)
- CSS/JavaScript personnalisé (Vanilla JS)

1.4 Technologies Utilisées

Catégorie	Technologies
Framework Backend	Laravel 10
Langage	PHP 8.2
Base de données	MySQL 8.0
ORM	Eloquent
Frontend	Blade Templates, Vanilla JavaScript, CSS3
Visualisation	Chart.js 4.4
Gestion de versions	Git, GitHub
Outils	MySQL Workbench, Laravel Migrations, Seeders

2. ARCHITECTURE GLOBALE DU SYSTÈME

Le système est organisé en quatre modules principaux interconnectés, chacun développé par un membre de l'équipe :

2.1 Architecture en Couches

Couche Présentation (Frontend)

- Design System unifié avec palette colorimétrique Elite Business
- Composants Blade réutilisables (boutons, cartes, formulaires, modales)
- Interface responsive avec Flexbox et CSS Grid
- JavaScript Vanilla avec namespace global (window.App)

Couche Métier (Backend)

- Controllars Laravel organisés par fonctionnalité
- Services métier pour la logique complexe (Statistics, Activity Logs)
- Policies et Gates pour les autorisations
- Form Requests pour la validation

Couche Données

- Modèles Eloquent avec relations définies
- Migrations versionnées
- Seeders et Factories pour données de test
- Optimisation des requêtes (Eager Loading, indexation)

2.2 Modules du Système

Module	Responsable	Fonctionnalités Clés
Design System & UI	Chayma Oualili	Identité visuelle, composants réutilisables, responsive design, système de notifications
Base de Données & Dashboards	Fatima Boumehaout	Modélisation ERD (13 tables), modèles Eloquent, 4 dashboards par rôle
Authentification & Sécurité	Fatima Zahrae El Bouhssini	Login/Register, gestion multi-rôles, permissions, activity logs, CSRF protection
Gestion Ressources & Analytics	Ouarda ait el fakih	CRUD ressources, maintenance, incidents,

		statistiques, graphiques Chart.js
Réservations	Halima Achabbak	À compléter : système de réservation, gestion conflits, notifications

3. MODULE DESIGN SYSTEM & UI

Développé par : CHAYMA OUALILI

3.1 Rôle et Problématique

Rôle : UI/UX Designer & Frontend Developer

Problématique résolue : Comment harmoniser le travail de 5 développeurs différents pour que l'application ressemble à un produit unique, professionnel et Premium, tout en garantissant une réactivité parfaite sur mobile ?

3.2 Palette Colorimétrique - Identité Elite Business

Couleur	Code Hex	Usage
Bleu Nuit	#0f1e3f	Header, Sidebar, Boutons primaires
Doré Ocre	#997953	Hover, bordures actives, icônes
Crème	#e2d1b9	Fonds de cartes, bordures légères
Bleu Delft	#213a56	Footer, textes de titres

3.3 Principes de l'Interface

Principe	Implémentation
Atomic Design	Création de composants réutilisables (boutons, inputs, modales)
Namespace Global	Utilisation de window.App en Vanilla JS pour éviter les conflits
Mobile First	Interface entièrement responsive via Flexbox et Media Queries
Feedback Immédiat	Système de notifications (Toasts) pour chaque action utilisateur

3.4 Composants Développés

Composants Blade

- Layouts : admin.blade.php, app.blade.php, guest.blade.php
- Components : alert, badge, breadcrumbs, button, card, footer

- Formulaires : form-input, form-select, form-textarea
- Navigation : navigation.blade.php avec menu mobile

Composants JavaScript (Vanilla JS)

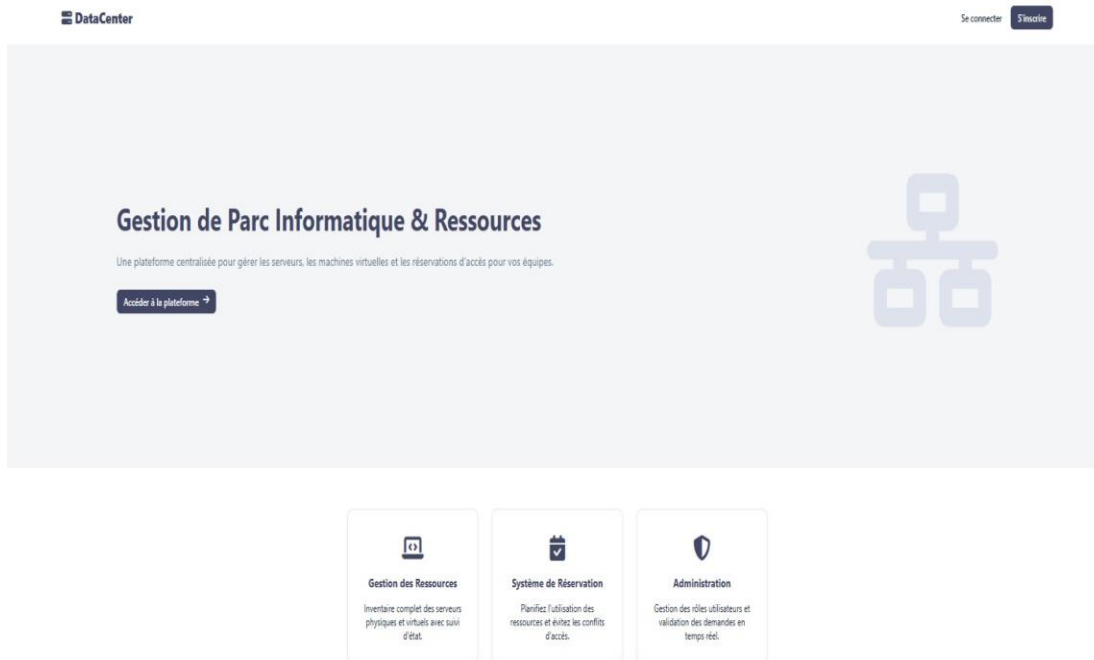
- Système de Notifications : Toasts animés avec 4 états (success, error, warning, info)
- Système de Tabs : Navigation par onglets pour dashboards complexes
- Modales Dynamiques : Fenêtres surgissantes pour confirmations
- Menu Mobile : Navigation optimisée pour écrans tactiles

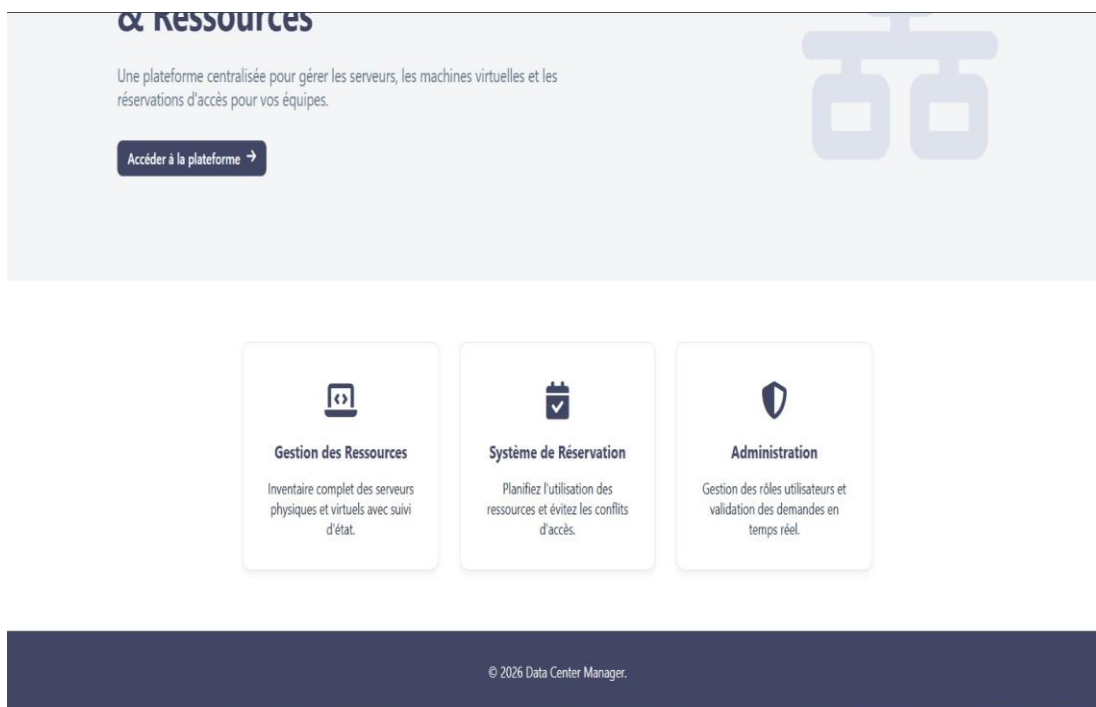
Fichiers CSS Créés

- admin.css : Variables globales, resets CSS
- app.css : Styles boutons, cartes, modales
- notifications.css : Styles système de notifications
- reservations.css, resources.css, statistics.css : Styles spécifiques modules

3.5 Captures d'Écran

Page d'Accueil





4. MODULE BASE DE DONNÉES & DASHBOARDS

Développé par : **FATIMA**

4.1 Rôle et Problématique

Rôle : Développeuse Backend & Base de Données

Problématique résolue : Comment structurer efficacement les données pour supporter les fonctionnalités de gestion des ressources, des réservations et des utilisateurs tout en maintenant des performances optimales ? Comment adapter l'interface dashboard aux besoins spécifiques de chaque rôle utilisateur ?

4.2 Structure de la Base de Données

Tables Principales (13 tables)

Table	Description
users	Gestion des comptes utilisateurs avec rôles (FK: role_id)
roles	Définition des 4 rôles système

resources	Inventaire complet équipements (FK: category_id)
resource_categories	Classification types de ressources
reservations	Planification réservations (FK: user_id, resource_id)
maintenances	Suivi interventions techniques (FK: resource_id)
incidents	Signalement problèmes (FK: user_id, resource_id)
activity_logs	Journal audit toutes activités (FK: user_id)
notifications	Système notifications (FK: user_id)
account_requests	Demandes création compte
settings	Paramètres système configurables

Relations Clés (11 relations)

- users → roles : belongsTo (role_id)
- resources → resource_categories : belongsTo (category_id)
- reservations → users : belongsTo (user_id)
- reservations → resources : belongsTo (resource_id)
- maintenances → resources : belongsTo (resource_id)
- incidents → users : belongsTo (user_id)
- incidents → resources : belongsTo (resource_id)
- activity_logs → users : belongsTo (user_id)
- notifications → users : belongsTo (user_id)
- account_requests → roles : belongsTo (role_requested)
- users → reservations, incidents, notifications : hasMany

4.3 Fonctionnalités Implémentées

1. Conception de la Base de Données

- Création des 13 tables avec relations optimisées
- Définition des contraintes d'intégrité référentielle (Foreign Keys)
- Indexation stratégique pour les requêtes fréquentes
- Normalisation jusqu'à la 3ème forme normale

2. Modèles Eloquent

- Relations définies : hasMany, belongsTo, belongsToMany
- Accessors et Mutators pour le formatage des données
- Événements et observateurs pour la logique métier automatique
- Scopes pour les requêtes récurrentes

3. Seeders et Factories

- Données de test réalistes avec Faker
- Peuplement initial de la base avec 13 tables
- Hiérarchie de seeders (DatabaseSeeder, UserSeeder, RoleSeeder, etc.)
- Factories pour génération automatique de données de démonstration

4. Services de Données

- StatisticsService pour agrégations complexes
- Gestion des transactions de base de données atomiques
- Optimisation des requêtes (Eager Loading, éviter N+1)
- Cache des requêtes fréquentes pour performances

5. Intégration Multi-Modules

- Liaison avec le système d'authentification (Users, Roles)
- Synchronisation avec le module de réservations
- Intégration avec le système de notifications
- Support du système d'activity logs pour traçabilité

4.4 Dashboards par Rôle

4 dashboards personnalisés ont été créés pour adapter l'interface aux besoins spécifiques de chaque profil utilisateur :

1. Dashboard Admin ([admin.blade.php](#))

Rôle : Administration complète du système

- Navigation complète : Dashboard, Ressources, Utilisateurs, Réservations, Logs, Settings
- Statistiques avancées : Total ressources, utilisateurs, réservations, incidents
- Actions rapides : Gestion ressources, validation réservations, planification maintenances
- Accès aux logs système et configurations globales

2. Dashboard Invité ([invite.blade.php](#))

Rôle : Vue publique et découverte

- Navigation limitée : Dashboard, Ressources, Disponibilités
- Statistiques publiques : Total ressources, disponibles, nombre utilisateurs
- Actions d'invitation : Voir ressources publiques, vérifier disponibilités
- Modal de contact fonctionnel pour demande de compte

3. Dashboard Tech Manager (tech.blade.php)

Rôle : Supervision technique et maintenance

- Navigation technique : Ressources, Maintenances, Incidents, Statistiques
- Monitoring système : Serveurs actifs, réservations en attente, maintenances planifiées
- Actions techniques : Gestion ressources, validation réservations, résolution incidents
- Planification et suivi des interventions

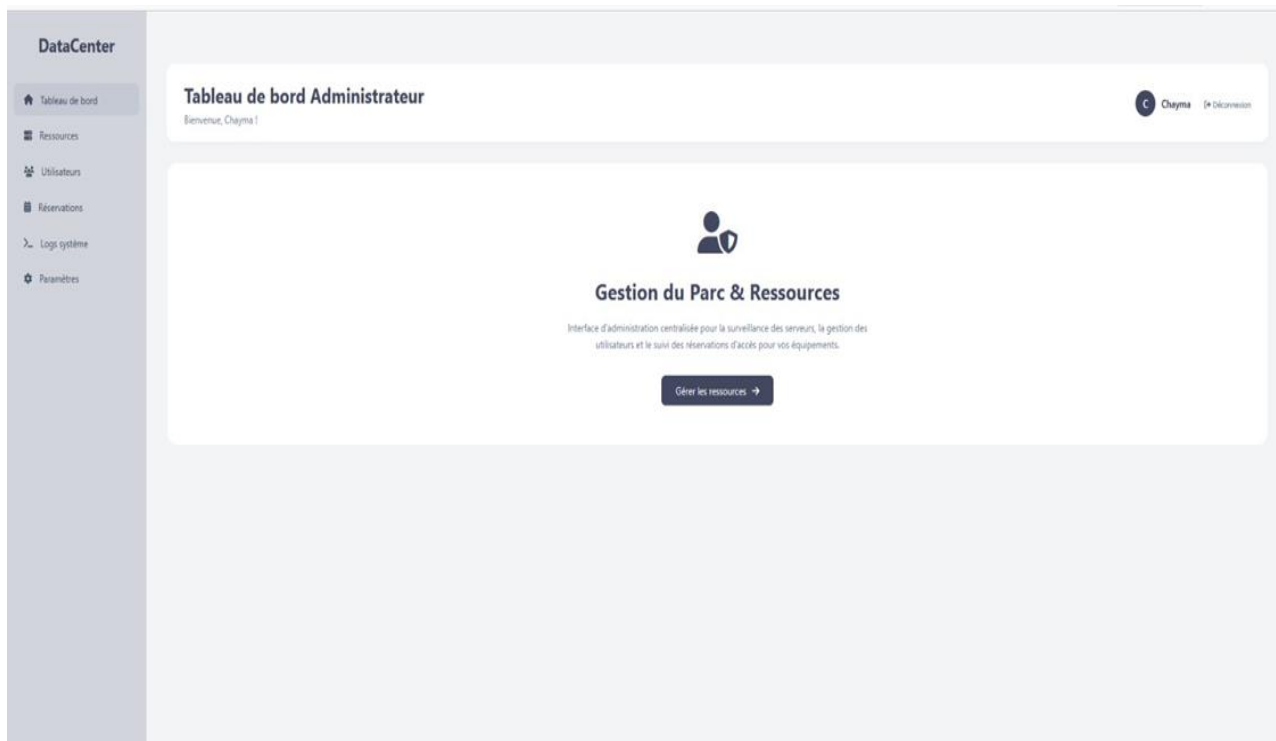
4. Dashboard User (user.blade.php)

Rôle : Utilisateur interne (Ingénieur/Enseignant/Doctorant)

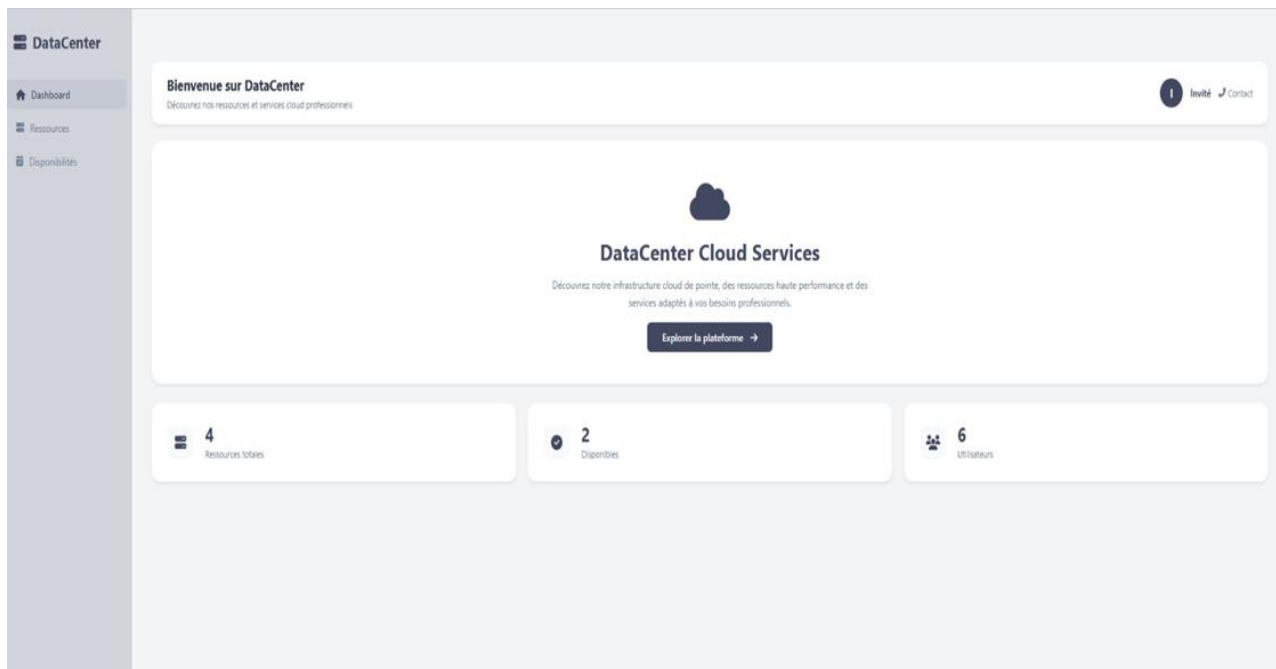
- Navigation : Dashboard, Mes Réservations, Ressources, Notifications
- Statistiques personnelles : Réservations actives, historique, notifications
- Actions utilisateur : Faire réservation, signaler incident, consulter disponibilités
- Suivi de ses demandes avec statuts : En attente, Approuvée, Refusée, Active

4.5 Captures d'Écran des Dashboards

Dashboard Admin



Dashboard Invité




Dashboard User


DataCenter

- Tableau de bord
- Ressources
- Catégories
- Mes réservations
- Incidents
- Maintenances
- Notifications3
- Paramètres

Tableau de bord

Bienvenue, Fatima Zahrae ! Voici un aperçu de votre activité.



 Fatima Zahrae
 [Déconnexion](#)



Bienvenue sur votre espace personnel


Gérez facilement vos réservations, consultez vos notifications et accédez à toutes les fonctionnalités de la plateforme depuis ce tableau de bord.

+ Nouvelle réservation




0

Réservations actives




0

En attente



0


Terminées



3

Nouvelles notifications


Actions rapides



Réserver une ressource

Consultez les disponibilités et réservez serveurs, VMs ou équipements


Nouvelle réservation



Mes réservations

Consultez l'état de vos réservations en cours, à venir ou passées


Voir mes réservations



Notifications

Consultez vos notifications et restez informé des mises à jour

Voir les notifications



Signaler un incident

Signalez un problème technique ou une panne d'équipement

Nouveau signalement

Activité récente

4.6 Structure des Fichiers

Modèles Eloquent (7 modèles)

- User.php : Gestion utilisateurs avec relations roles
- Resource.php : Modèle ressources avec catégories
- Reservation.php : Réservations avec users et ressources
- Category.php : Classification des ressources

- Maintenance.php : Suivi des interventions
- ActivityLog.php : Journal d'audit
- Notification.php : Système de notifications

Dashboards (4 vues)

- resources/views/dashboard/admin.blade.php
- resources/views/dashboard/invite.blade.php
- resources/views/dashboard/tech.blade.php
- resources/views/dashboard/user.blade.php

4.7 Métriques du Module

Métrique	Valeur
Nombre de tables	13
Nombre de relations	11
Modèles Eloquent	7
Dashboards créés	4
Migrations	13
Seeders	8

4.8 Commandes de Déploiement

php artisan migrate --seed

Exécute toutes les migrations et peuple la base avec les données initiales

php artisan migrate:status

Vérifie l'état des migrations

php artisan db:show

Affiche les informations de la base de données

php artisan optimize

Optimise les performances (cache config, routes, views)

5. MODULE AUTHENTIFICATION & SÉCURITÉ

Développé par : FATIMA ZAHRAE

5.1 Rôle et Problématique

Rôle : Security & Authentication Specialist

Problématique résolue : Comment permettre à 4 types d'utilisateurs différents (Invité, User, Tech Manager, Admin) d'accéder aux ressources appropriées tout en maintenant une sécurité maximale et une traçabilité complète ?

5.2 Système d'Authentification Complet

Fonctionnalités d'Authentification

- Login/Logout : Connexion sécurisée avec option 'Se souvenir de moi'
- Inscription : Création de compte avec validation email
- Password Reset : Récupération mot de passe par email avec token temporaire
- Protection CSRF : Tokens automatiques sur tous les formulaires
- Rate Limiting : Protection contre attaques brute force (5 tentatives/min)

5.3 Gestion Multi-Rôles

Rôle	Permissions Principales
Invité	Lecture seule ressources publiques, demande de compte
Utilisateur Interne	Réservations, historique personnel, notifications
Tech Manager	Gestion ressources, approbation réservations, maintenance
Administrateur	Contrôle total, gestion utilisateurs, statistiques globales

5.4 Matrice des Permissions Détaillée

Fonctionnalité	Invité	User	Tech Mgr	Admin
Voir ressources publiques	✓	✓	✓	✓
Faire réservation	✗	✓	✓	✓

Approuver réservation	✗	✗	✓	✓
Créer utilisateur	✗	✗	✗	✓
Voir logs système	✗	✗	✗	✓
Gérer maintenance	✗	✗	✓	✓

5.5 Demandes de Compte

- Formulaire Public : Les invités peuvent demander un compte avec justification
- Validation Admin : L'administrateur approuve ou refuse avec justification
- Notifications Email : Email automatique lors de la décision
- Suivi Complet : Historique de toutes les demandes avec statuts

5.6 Sécurité Avancée Implémentée

Mesure de Sécurité	Implémentation
CSRF Tokens	Protection automatique sur tous les formulaires POST
Password Hashing	Bcrypt avec coût adapté (12 rounds)
Rate Limiting	5 tentatives max/minute sur login
XSS Protection	Échappement automatique Blade {{ }}
SQL Injection	Protection via Eloquent avec bindings
Session Security	Configuration sécurisée cookies HttpOnly
Activity Logging	Traçabilité complète actions sensibles

5.7 Interfaces Développées

Pages d'Authentification (5 vues)

- login.blade.php : Connexion principale avec CSRF
- register.blade.php : Inscription utilisateurs
- forgot-password.blade.php : Demande réinitialisation
- reset-password.blade.php : Formulaire nouveau mot de passe
- verify-email.blade.php : Vérification email

Pages Administration (8 vues)

- users/index.blade.php : Liste utilisateurs avec filtres
- users/create.blade.php : Création utilisateur
- users/edit.blade.php : Modification profil

- users/show.blade.php : Détails utilisateur
- account-requests/index.blade.php : Demandes en attente
- account-requests/approve.blade.php : Approbation/refus
- logs.blade.php : Journal d'activité complet

5.8 Statistiques du Module

Métrique	Valeur
Routes sécurisées	23
Middleware de protection	4
Contrôleurs développés	4
Vues Blade créées	13
Tests unitaires	8
Temps de développement	6 jours
Lignes de code (PHP + Blade)	~3,500

5.9 Captures d'écran

Data Center Ressources Catégories Réservations Incidents Maintenances Notifications Dashboard

Connexion Inscription

Connexion

Email

Mot de passe

Se connecter

Pas encore de compte? [S'inscrire](#)
Mot de passe oublié?

© 2026 Data Center Manager. Tous droits réservés.
Développé par l'équipe Projet (Fatima, Zahrae, Ouada, Halima, Chaymae).

Data Center
Ressources
Catégories
Réservations
Incidents
Maintenance
Notifications
Dashboard
Connexion
Inscription

Inscription

Nom complet

Email

Téléphone (optionnel)

Département

Rôle demandé

Sélectionner un rôle

Motivation

Mot de passe

Confirmer le mot de passe

S'inscrire

Déjà un compte? Se connecter

© 2020 Data Center Manager. Tous droits réservés.

Réinitialiser le mot de passe

Choisissez votre nouveau mot de passe

Token généré! Vous pouvez maintenant réinitialiser votre mot de passe.

Adresse Email

nadin@gmail.com

Nouveau mot de passe

Confirmer le mot de passe

Réinitialiser le mot de passe

Retour à la connexion S'inscrire

Mot de passe oublié

Entrez votre adresse email pour réinitialiser votre mot de passe (mode développement)

Adresse email

nadin@gmail.com

Générer le lien de réinitialisation

Retour à la connexion S'inscrire

6. MODULE GESTION RESSOURCES & ANALYTICS

Développé par : OUARDA AIT EL FAKIH

6.1 Rôle et Problématique

Rôle : Backend Developer & Data Analyst

Problématique résolue : Comment gérer efficacement des centaines de ressources matérielles tout en maintenant une visibilité complète sur leur état, leur utilisation et leur maintenance ?

6.2 Gestion du Cycle de Vie des Ressources

Chaque ressource suit un flux d'états défini :

Disponible → Peut être réservée immédiatement

Réservée → En cours d'utilisation par un utilisateur

Maintenance → Intervention technique en cours

Hors service → Nécessite remplacement

6.3 Fonctionnalités de Gestion des Ressources

CRUD Complet

- Création : Ajout de nouvelles ressources avec formulaire validé
- Lecture : Liste paginée avec filtres avancés (nom, catégorie, statut)
- Modification : Édition des caractéristiques techniques
- Suppression : Suppression sécurisée (soft delete) avec vérifications

Catégorisation Flexible

- Serveurs : CPU, RAM, stockage, OS
- Machines Virtuelles : Ressources allouées, hyperviseur
- Stockage : Capacité, type (SSD/HDD), RAID
- Équipements Réseau : Switches, routeurs, bande passante

Vue Publique

- Catalogue consultable sans authentification
- Filtres par catégorie et disponibilité
- Fiches détaillées avec caractéristiques techniques

6.4 Maintenance & Incidents

Signalement d'Incidents

- Reporting rapide par les utilisateurs
- Niveaux de gravité : Critique, Élevé, Moyen, Faible
- Assignation automatique au Tech Manager

- Suivi du statut : Ouvert, En cours, Résolu, Fermé

Planification de Maintenance

- Calendrier des interventions programmées
- Détection automatique des conflits avec réservations
- Notifications aux utilisateurs affectés
- Historique complet des maintenances

6.5 Analytics & Statistiques Avancées

Le module fournit des analyses en temps réel avec visualisations Chart.js :

Type de Graphique	Données Visualisées
Taux d'Occupation	Pourcentage ressources utilisées vs disponibles
Répartition par Statut	Graphique circulaire des états (Dispo, Réservé, Maintenance)
Top Ressources	Classement par taux d'utilisation
Tendances Temporelles	Évolution utilisation sur 6 mois
KPIs Personnalisés	Métriques par utilisateur ou département

6.6 Captures d'Écran du Module

- Liste des Ressources

Data CenterRessourcesCatégoriesRéservationsIncidentsMaintenancesNotificationsStatistiquesDashboard

chayma oualiliDéconnexion

Gestion de l'InventaireAjouter

Rechercher...

Catégorie

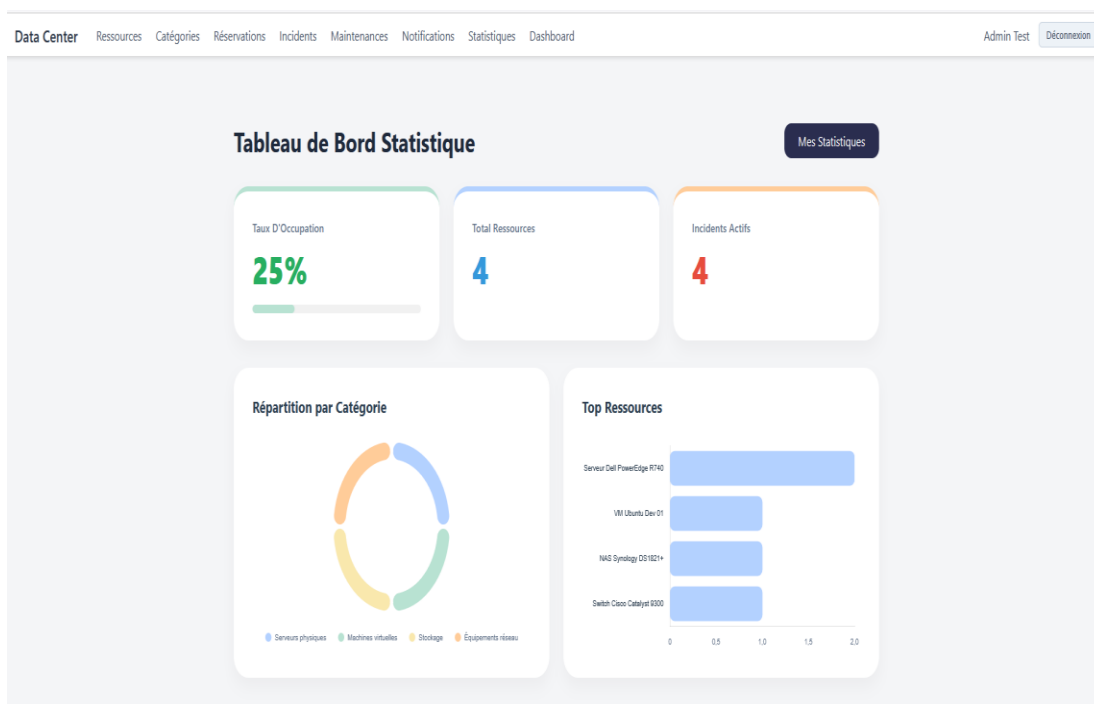
Statut

Filtrer

# ID	NOM	CPU / RAM	STATUT	ACTIONS
#1	Serveur Dell PowerEdge R740	Intel Xeon Silver 4210 / 64GB	MAINTENANCE	DétailsEditer
#2	VM Ubuntu Dev 01	4 vCPU / 16GB	BUSY	DétailsEditer
#3	NAS Synology DS1821+	/	AVAILABLE	DétailsEditer
#4	Switch Cisco Catalyst 9300	/	MAINTENANCE	DétailsEditer
#6	chayma oualili	12 / 16	BUSY	DétailsEditer

© 2026 Data Center Manager. Tous droits réservés.

- Dashboard Statistiques



- Liste des Incidents

Data Center Ressources Catégories Réservations Incidents Maintenances Notifications Statistiques Dashboard Admin Test Déconnexion

Liste des Incidents

DATE	RESSOURCE	DESCRIPTION	GRAVITÉ	UTILISATEUR	STATUT	ACTIONS
27/01/2026 15:02	Serveur Dell PowerEdge R740	serveur injoignable	MOYENNE	Responsable Technique	OUVERT	Détails
27/01/2026 14:27	Serveur Dell PowerEdge R740	serveur injoignable	MOYENNE	Fatima Zahrae	RÉSOLU	Détails
15/01/2026 09:07	VM Ubuntu Dev 01	La VM Ubuntu Dev 01 ne répond pas au démarrage.	MOYENNE	Fatima Zahrae	RÉSOLU	Détails
15/01/2026 03:07	Serveur Dell PowerEdge R740	Le serveur émet un bruit suspect dans le rack 3.	MOYENNE	Ouarda	OUVERT	Détails
15/01/2026 03:07	Ressource supprimée	Problème de connexion réseau intermittent dans L.	MOYENNE	Halima	OUVERT	Détails
15/01/2026 03:07	Switch Cisco Catalyst 9300	Port 10 du switch ne fonctionne plus correctement.	MOYENNE	Fatima	OUVERT	Détails

© 2026 Data Center Manager. Tous droits réservés.
Développé par l'équipe Projet (Fatima Zahrae, Ouarda, Halima, Chayma)

• Planification Maintenance

Data Center
Ressources
Catégories
Réservations
Incidents
Maintenances
Notifications
Statistiques
Dashboard

chayma oualili Déconnexion

Planning des Maintenances

+ Planifier une intervention

PÉRIODE	RESSOURCE	RAISON / DÉTAILS	ACTIONS
Du 29/01/2026 02:18 Au 30/01/2026 04:18	Switch Cisco Catalyst 9300 Équipements réseau	recherche	Annuler
Du 29/01/2026 02:20 Au 30/01/2026 04:20	Serveur Dell PowerEdge R740 Serveurs physiques	recherche	Annuler

© 2026 Data Center Manager. Tous droits réservés.
Développé par l'équipe Projet (Fatima, Zahraa, Ouarda, Halima, Chaymae).


• Catégories

Data Center
Ressources
Catégories
Réservations
Incidents
Maintenances
Notifications
Statistiques
Dashboard

chayma oualili Déconnexion

Gestion du Catalogue de Ressources


Nouvelle Catégorie


1 ressources

Serveurs physiques

Serveurs dédiés hébergés dans le Data Center


Voir Catalogue


1 ressources

Machines virtuelles

VMs créées sur les hyperviseurs pour différents services

Voir Catalogue


2 ressources

Stockage

Équipements de stockage comme NAS ou SAN

Voir Catalogue


1 ressources

Équipements réseau

Switches, routeurs et autres équipements réseau

Voir Catalogue

6.7 Niveaux d'Accès par Rôle

Rôle	Consultation	Création	Modification	Suppression
Invité	✓Public	✗	✗	✗
User	✓Toutes	✗	✗	✗
Tech Manager	✓Toutes	✓	✓	✗
Admin	✓Toutes	✓	✓	✓

6.8 Interfaces Développées

Pages Ressources (5 vues)

- public-index.blade.php : Catalogue public
- index.blade.php : Liste admin avec filtres avancés
- create.blade.php : Formulaire création
- edit.blade.php : Formulaire édition
- show.blade.php : Fiche détaillée

Autres Pages (8 vues)

- Catégories : index, create, edit (3 vues)
- Incidents : index, create, show (3 vues)
- Statistiques : index, my-resources (2 vues)

6.9 Structure du Code

Controllers (5 fichiers)

- ResourceController.php : CRUD + Filtres
- CategoryController.php : Gestion catégories
- IncidentController.php : Signalement + Résolution
- MaintenanceController.php : Planification
- StatisticsController.php : Analytics

Services

- ResourceStatisticsService.php : Calculs métiers complexes

Assets Frontend

- resources.css : Styles composants
- statistics.css : Styles graphiques
- statistics.js : Génération charts Chart.js

7. MODULE RÉSERVATIONS

Développé par : Halima Achabbak

7.1 Rôle et Problématique

Rôle : Backend Developer & Notification System Specialist

Problématique résolue : Comment centraliser et automatiser la gestion des réservations de ressources informatiques dans un Data Center, en assurant la détection des conflits, la validation des demandes et la notification en temps réel des utilisateurs et gestionnaires ?

Contexte Avant le Module

- Gestion manuelle des réservations par emails et appels téléphoniques
- Conflits de réservations non détectés automatiquement
- Absence de suivi centralisé des disponibilités
- Pas d'historique ni de traçabilité des utilisations

7.2 Architecture Backend

Composant	Technologie	Rôle
Framework	Laravel 10	Architecture MVC et gestion des requêtes
Base de données	MySQL	Stockage persistant des réservations
ORM	Eloquent	Relations entre modèles (User, Resource, Reservation)
Scheduler	Laravel Scheduler	Tâches planifiées (activation, fin, rappels)
Services	Classes PHP	Logique métier (validation, notifications)

7.3 Fonctionnalités Développées

Système de Réservation Complet

Workflow de réservation :

- Création de demande → statut pending
- Validation Tech Manager → statut approved
- Début de réservation → statut active
- Fin de réservation → statut completed

Validation intelligente :

- ✓ Détection automatique des conflits de dates
- ✓ Vérification croisée avec les maintenances planifiées
- ✓ Alertes en temps réel lors de la saisie

- ✓ Interface de validation pour Tech Managers

Système de Notifications

Notifications multi-niveaux :

- Utilisateurs : Confirmations, rappels avant début/fin
- Tech Managers : Demandes en attente, validations requises
- Admins : Rapports d'utilisation, alertes système

Rappels automatisés :

- ✓ 60 minutes avant le début de la réservation
- ✓ Notification de fin de réservation
- ✓ Délais personnalisables par l'administrateur

Centre de notifications :

- ✓ Interface unifiée pour toutes les notifications
- ✓ Marquage lu/non-lu avec compteur
- ✓ Filtrage par type et date
- ✓ Historique conservé

Tableau de Bord Utilisateur

Statistiques personnelles :

- Total des réservations (toutes périodes)
- Réservations en cours
- Réservations à venir
- Historique des réservations terminées

7.4 Principes de Développement Backend

Principe	Implémentation
Traçabilité	Logs automatiques de toutes actions (création, modification, annulation)
Performance	Eager loading et pagination pour optimiser les requêtes
Évolutivité	Architecture modulaire basée sur des services extensibles
Sécurité	Middleware d'authentification et contrôle d'autorisation
Disponibilité	Notifications internes en temps réel

7.5 Fichiers Principaux Développés

Contrôleurs :

- app/Http/Controllers/ReservationController.php
- app/Http/Controllers/TechReservationController.php

Services :

- app/Services/ReservationValidationService.php
- app/Services/NotificationService.php

Modèles :

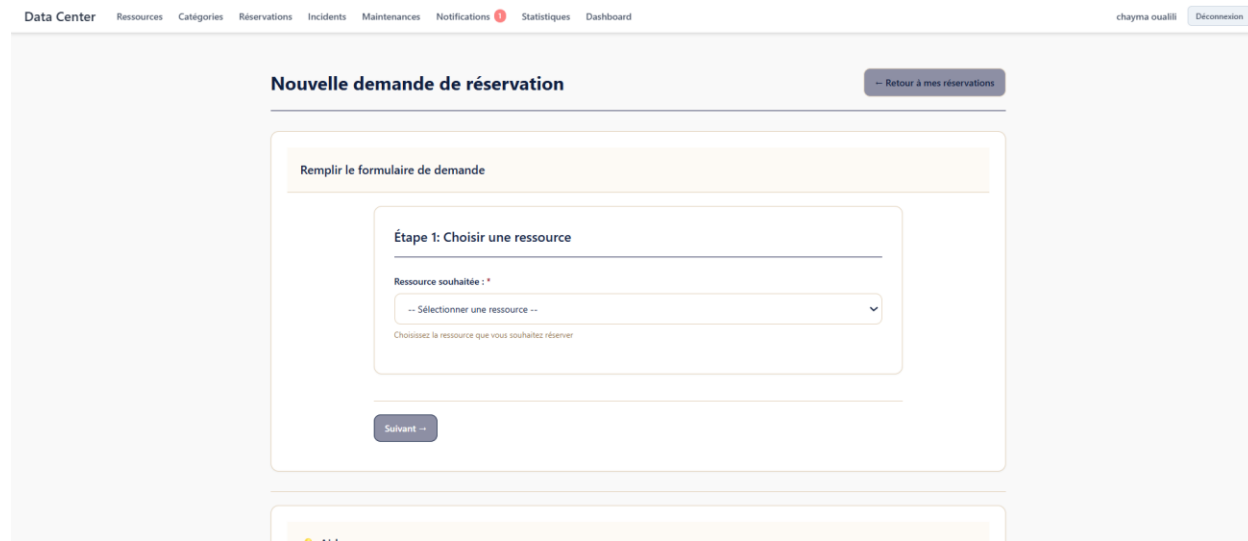
- app/Models/Reservation.php
- app/Models/Notification.php

Vues Blade :

- resources/views/reservations/ (create, index, history)
- resources/views/dashboard/user.blade.php
- resources/views/notifications/index.blade.php

7.6 Captures d'écran

- Demande des réservations



The screenshot shows a web application interface for creating a new reservation. At the top, there is a navigation bar with links: Data Center, Ressources, Catégories, Réservations, Incidents, Maintenances, Notifications (with a red badge), Statistiques, and Dashboard. On the right of the navigation bar, the user's name 'chayma oualili' and a 'Déconnexion' button are visible. The main heading is 'Nouvelle demande de réservation', with a 'Retour à mes réservations' button on the right. Below the heading is a section titled 'Remplir le formulaire de demande'. Inside this section, there is a sub-section 'Étape 1: Choisir une ressource'. It contains a label 'Ressource souhaitée : *' and a dropdown menu with the text '-- Sélectionner une ressource --'. Below the dropdown, there is a small instruction: 'Choisissez la ressource que vous souhaitez réserver'. At the bottom of the form, there is a 'Suivant -->' button. At the very bottom of the page, there is an 'Aide' link with a yellow star icon.

- Listes des notifications

Data Center
Réervations
Notifications 1
chayma oualili
Déconnexion

Mes Notifications

✓ Tout marquer comme lu
Mes réservations

Type: Tous les types
Statut: Tous les statuts
Filtrer Réinitialiser

1 Total des notifications
1 Non lues
0 Déjà lues

☐
Réservation soumise : Votre réservation pour NAS Synology DS1821+ est en attente d'approbation.
28/01/2026 20:49 NON LUE
☒ ☐

- Listes des réservations

Data Center
Ressources
Catégories
Réservations
Incidents
Maintenances
Notifications 1
Statistiques
Dashboard
chayma oualili
Déconnexion

Mes Réservations

+ Nouvelle Réservation

Statut: Tous
Période: JJ/MM/AAAA - JJ/MM/AAAA
Filtrer Réinitialiser

Liste de vos réservations

RESSOURCE	DATES	JUSTIFICATION	STATUT	ACTIONS
NAS Synology DS1821+ Stockage	Début: 06/03/2026 21:48 Fin: 29/03/2026 21:48	Recherche - Expériences sur machines virtuelles	EN ATTENTE	<input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>

- Historique des réservations passés

jj/mm/aaaa

Filtrer

ID	UTILISATEUR	RESSOURCE	DATE DÉBUT	DATE FIN	STATUT	ACTIONS
7	Fatima Zahrae	VM Ubuntu Dev 01	28/01/2026 19:40	28/01/2026 19:45	✓ Terminée	
6	Fatima Zahrae	Switch Cisco Catalyst 9300	28/01/2026 18:45	28/01/2026 18:50		
5	Fatima Zahrae	Serveur Dell PowerEdge R740	28/01/2026 14:30	28/01/2026 15:31	⌚ En attente	
1	Fatima Zahrae	Serveur Dell PowerEdge R740	29/01/2026 10:49	31/01/2026 10:49	✓ Approuvée	
2	Ouarda	VM Ubuntu Dev 01	30/01/2026 10:49	02/02/2026 10:49	✓ Approuvée	
3	Halima	NAS Synology DS1821+	28/01/2026 10:49	30/01/2026 10:49	🔒 Active	
4	Fatima	Switch Cisco Catalyst 9300	29/01/2026 10:49	01/02/2026 10:49	⌚ En attente	

Cette page affiche toutes les réservations. Pour approuver ou refuser les demandes en attente, utilisez la page Demandes en attente.

8. INTÉGRATION ET WORKFLOW GLOBAL

8.2 Gestion de Version Git

- Branches principales : main (production)
- Branche commune : dev
- Branches par module :
 - feature/frontend,
 - feature/backend/reservations,
 - feature/backend/resources,
 - feature/backend/-database,
 - feature/backend/authentication-securite

9. DÉPLOIEMENT ET CONFIGURATION

9.1 Prérequis Système

Composant	Version Requise
PHP	≥ 8.2
MySQL	≥ 8.0
Composer	≥ 2.0
Extensions PHP	OpenSSL, PDO, Mbstring, Tokenizer, XML, Ctype, JSON

9.2 Installation en 6 Étapes

1. `git clone [URL] && cd datacenter-management-system`
2. `composer install`
3. `cp .env.example .env && php artisan key:generate`
4. Configurer `.env` (`DB_DATABASE`, `DB_USERNAME`, `DB_PASSWORD`)
5. `php artisan migrate --seed`
6. `php artisan serve (http://localhost:8000)`

9.3 Comptes de Test

Rôle	Email	Mot de passe
Admin	admin@datacenter.com	password
Tech Manager	techmanager@datacenter.com	password
User	user@datacenter.com	password

10. TESTS ET VALIDATION

10.1 Validation du Cahier des Charges

Exigence	Statut
Base de données relationnelle MySQL	✓ Complété - 13 tables, 11 relations
Interface web ergonomique et responsive	✓ Complété - Mobile First, 4 dashboards
Authentification Laravel (middleware)	✓ Complété - 4 rôles, 23 routes sécurisées
Gestion multi-rôles	✓ Complété - Matrice permissions complète
Système de réservation	❑ En cours - Membre 5
Gestion conflits (overlapping)	❑ En cours - Membre 5
Notifications messagerie interne	❑ En cours - Membre 5
Statistiques graphiques	✓ Complété - Chart.js avec 5 types graphiques
CSS/JS personnalisé (sans frameworks)	✓ Complété - Vanilla JS, CSS3 moderne
Journalisation actions importantes	✓ Complété - Activity Logs complets

11. STATISTIQUES GLOBALES DU PROJET

11.1 Métriques de Développement

Métrique	Valeur
Durée totale projet	~8 jours
Nombre de développeurs	5 (4 rapports soumis)
Tables de base de données	13
Relations définies	11
Modèles Eloquent	7
Controllers	13+
Vues Blade créées	45+
Routes définies	60+
Composants réutilisables	15+
Dashboards	4 (par rôle)

12. CONCLUSION

Le Data Center Management System représente une solution complète et professionnelle pour la gestion des ressources informatiques d'un Data Center. Développé en équipe sur une période de 8 jours, le projet démontre une maîtrise approfondie de Laravel, PHP, MySQL et des technologies web modernes.

12.1 Perspectives

Le projet a posé des fondations solides pour un système de gestion de Data Center professionnel. La base de données relationnelle avec 13 tables interconnectées, les 4 dashboards adaptés par rôle, et l'architecture modulaire permettent une évolutivité aisée.

Avec la finalisation du module Réservations et les améliorations futures planifiées (API REST, notifications temps réel, export statistiques), l'application sera prête pour un déploiement en environnement de production.

ÉQUIPE DATA CENTER MANAGEMENT

———— FIN DU RAPPORT ————