

BTS SIO 2026

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

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

PAGE DE PRÉSENTATION DU DOSSIER

N° d'inscription¹ : | 0 | 2 | 5 | 4 | 1 | 4 | 7 | 6 | 3 | 1 | 4 |

NOM : TARDY

PRÉNOM : Samuel

Date de passage² : / / 2026

Heure de passage² :h.....

CATÉGORIE CANDIDAT¹ (UNE CASE À COCHER)

☐ Scolaire

☒ Apprenti

☐ Formation professionnelle continue

☐ Expérience professionnelle

☐ Ex-scolaire

☐ Ex-apprenti

☐ Ex-formation professionnelle continue

¹ Informations communiquées sur votre confirmation d'inscription

² Informations à ne pas compléter

Tampon de L'établissement

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS SESSION 2026 ANNEXE
VII-1-B : Fiche descriptive de réalisation professionnelle (recto) Épreuve E6 -
Conception et développement d'applications (option SLAM)

DESCRIPTION D'UNE RÉALISATION PROFESSIONNELLE		N° réalisation : 1
Nom, prénom : TARDY Samuel		N° candidat : 02541476314
Épreuve ponctuelle ☒ Contrôle en cours de formation ☐		Date : 04/04/2026
Organisation support de la réalisation professionnelle Club de Tennis de Rambouillet		
Intitulé de la réalisation professionnelle Site web dédié au club de tennis de Rambouillet (Non mis en production)		
Période de réalisation : 2025 / 2026 Lieu : Lycée Louis Bascan Modalité : ☒ Seul(e) ☐ En équipe		
Compétences travaillées ☒ Concevoir et développer une solution applicative ☒ Assurer la maintenance corrective ou évolutive d'une solution applicative ☒ Gérer les données		
Conditions de réalisation (ressources fournies, résultats attendus) La réalisation professionnelle a été développée à partir de zéro. Résultat attendu : Site web complet et fonctionnel à destination du Club de Tennis de Rambouillet Authentification des membres (inscription, connexion sécurisée, espace personnel) Réservation de terrains en ligne avec gestion automatique des conflits de créneaux Boutique en ligne avec panier persistant et paiement via PayPal (mode démo) Gestion des événements et tournois avec système d'inscription/désinscription en ligne Panel d'administration complet (utilisateurs, terrains, événements, tournois, boutique, médias) Galerie médias protégée par mot de passe (photos locales + intégration vidéos YouTube)		
Description des ressources documentaires, matérielles et logicielles utilisées Hébergement en local sur ma machine (XAMPP / Apache) GitHub : https://github.com/Sasukisse/tennis-club-rambouillet Disponible visuellement sur samueltardy.fr Langages : HTML5, CSS3, JavaScript (vanilla), PHP (procédural) Accès base de données : PDO (PHP Data Objects) Serveur local : Apache via XAMPP Base de données : MySQL (encodage utf8mb4) Éditeur de code : Visual Studio Code		
Modalités d'accès aux productions et à leur documentation Code source de l'application : https://github.com/Sasukisse/tennis-club-rambouillet samueltardy.fr		

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS SESSION 2026

**ANNEXE VII-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

Le site web développé pour le Club de Tennis de Rambouillet est une application web complète permettant aux membres du club de gérer leur adhésion en ligne, de réserver des terrains, de s'inscrire à des événements et tournois, et d'acheter des articles sportifs via une boutique en ligne avec paiement PayPal. Il intègre également un espace d'administration complet permettant au personnel du club de gérer l'ensemble des fonctionnalités : utilisateurs, réservations de terrains, événements, tournois, catalogue boutique avec gestion multi-images, et galerie médias. Le code source est disponible sur GitHub.

Modèle conceptuel des données :

La base de données MySQL nafi7014_tennis est composée de 12 tables :

Table	Attributs principaux	Rôle
users	<i>id, email, password_hash, full_name, role</i>	Comptes membres et admins
user_profiles	<i>user_id, address_line, zip, city</i>	Adresse de livraison
user_carts	<i>user_id, items (JSON), updated_at</i>	Panier persistant par utilisateur
bookings	<i>id, user_id, date, court, start_time, end_time</i>	Réservations de terrains
events	<i>id, title, event_date, event_time, location</i>	Événements du club
event_participants	<i>user_id, event_id</i>	Inscriptions aux événements
tournaments	<i>id, title, start_date, end_date, max_participants</i>	Tournois
tournament_participants	<i>user_id, tournament_id</i>	Inscriptions aux tournois
products	<i>id, sku, name, price, variants, stock, active</i>	Catalogue boutique
product_images	<i>product_id, image_path, display_order, is_primary</i>	Images produits (multi)
youtube_videos	<i>id, youtube_url, title</i>	Vidéos galerie médias
settings	<i>key, value</i>	Paramètres dynamiques (ex : MDP médias)

Arborescence du projet :

tennis-club-rambouillet/

index.html ← Page d'accueil (présentation du club)

le-club.html ← Histoire et galerie photos

inscriptions.html ← Informations et tarifs d'adhésion

boutique.html ← Catalogue produits et panier

checkout.html ← Validation de commande

success.html ← Confirmation de commande

contact.html ← Formulaire de contact

medias.html ← Galerie médias (version statique)

terrains.php ← Réservation de terrain (PHP)

css/ ← Feuilles de style (une par page)

js/ ← Scripts JS (panier, boutique, login...)

img/ ← Images produits et médias

pdf/

schema.sql ← Schéma complet de la base de données

php/

config.php ← Connexion PDO à MySQL

login.php / register.php / logout.php

dashboard.php ← Espace membre

competitions.php ← Liste des tournois à venir

medias.php ← Galerie protégée par mot de passe

bookings-api.php ← API : réservations terrains

events-api.php ← API : événements

tournaments-api.php ← API : tournois

products-api.php ← API : catalogue boutique

cart.php ← API : panier serveur

me.php ← API : session courante

event-participe.php / tournament-participe.php

admin-*.php ← Panel d'administration complet

Fonctionnement général et workflows :

Fonctionnalité	Description
Authentification	Inscription avec hash bcrypt → connexion via password_verify → session PHP + cookie Remember Me signé HMAC SHA-256 (httponly, SameSite=Lax, 30 jours)
Réservation terrain	Sélection date/court/créneau → vérification anti-conflit SQL (bookings-api.php) → INSERT en base. Créneaux disponibles de 8h à 22h.
Boutique en ligne	Catalogue chargé via products-api.php → panier persistant (user_carts / localStorage) → checkout.html → paiement PayPal SDK (démon) → success.html
Événements & Tournois	Affichage liste + statut de participation → clic inscription → event-participate.php / tournament-participate.php → INSERT/DELETE table pivot
Galerie médias	Accès libre pour admins. Pour membres/visiteurs : saisie d'un mot de passe global vérifié via hash_equals en session. Affiche photos locales + vidéos YouTube.
Panel administration	CRUD complet : utilisateurs (promouvoir/rétrograder), événements, tournois, produits boutique (variantes taille/couleur, multi-images), réservations terrains, galerie médias. Toutes les pages admin sont protégées par require_admin().

Sécurité appliquée :

Mesure	Détail
Mots de passe	PASSWORD_BCRYPT + password_verify (aucun mot de passe stocké en clair)
Requêtes SQL	PDO avec prepare/execute sur toutes les requêtes (protection injection SQL)
Sorties HTML	htmlspecialchars() systématique (protection XSS)
En-têtes HTTP	hash_hmac SHA-256, httponly, SameSite=Lax (protection CSRF/vol de cookie)
Contrôle accès	Fonctions require_login() et require_admin() en tête de chaque fichier sensible

Ressources :

Dashboard membre (dashboard.php) :

Bonjour, Samuel TARDY

Bienvenue dans votre espace.

Événements à venir

Soirée barbecue Annuler

Bonjour à tous,

Nous organisons une soirée barbecue le vendredi 10 mars 2026 au club afin de passer un moment convivial tous ensemble.

Le club

vendredi 10 avril 2026 20:00 Club house

5 participant(s)

Mes informations

Email: samuel.tardy78@gmail.com

Rôle: Admin

Mon adresse

Adresse

Code postal **Ville**

Enregistrer

Tournois à venir

Tournoi du printemps Annuler

Bonjour à tous,

...

Mes réservations de terrains

Réservations en cours

Aucune réservation en cours.

...

Extrait du code :

```
// Prépare la persistance des informations personnelles (adresse)
$profileMsg = '';
$profileErr = '';
try{
    $pdo = db();
    $pdo->exec("CREATE TABLE IF NOT EXISTS user_profiles (
        user_id INT PRIMARY KEY,
        address_line VARCHAR(255),
        zip VARCHAR(10),
        city VARCHAR(128),
        CONSTRAINT fk_user_profiles_user FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE CASCADE
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4");
    $profile = ['address_line'=>'', 'zip'=>'', 'city'=>''];
    $st = $pdo->prepare('SELECT address_line, zip, city FROM user_profiles WHERE user_id=? LIMIT 1');
    $st->execute([intval($u['id'])]);
    $row = $st->fetch();
    if($row){ $profile = $row; }
    if($_SERVER['REQUEST_METHOD']=='POST' && isset($_POST['action']) && $_POST['action']=='update_address'){
        $address = trim($_POST['address_line'] ?? '');
        $zip = trim($_POST['zip'] ?? '');
        $city = trim($_POST['city'] ?? '');
        if(mb_strlen($address) > 255) $address = mb_substr($address,0,255);
        if(mb_strlen($zip) > 10) $zip = mb_substr($zip,0,10);
        if(mb_strlen($city) > 128) $city = mb_substr($city,0,128);
        $zipValid = preg_match('/^\d{5}$/', $zip) === 1;
        $cityValid = preg_match('/^[A-Za-zÀ-Öö-ÿ' . "\\\' " . ' \-]{2,128}$/u', $city) === 1;
        if(!$zipValid || !$cityValid){
            $profileErr = 'Code postal: 5 chiffres · Ville: lettres/espaces uniquement.';
            $profile = ['address_line'=>$address, 'zip'=>$zip, 'city'=>$city];
        } else {
            $st = $pdo->prepare('INSERT INTO user_profiles (user_id,address_line,zip,city) VALUES (?,?,,?) ON DUPLICATE KEY UPDATE address_line=VALUES(ad
            $st->execute([intval($u['id']), $address, $zip, $city]);
            $profileMsg = 'Informations mises à jour.';
            $profile = ['address_line'=>$address, 'zip'=>$zip, 'city'=>$city];
        }
    }
}
}catch(Throwable $e){ /* silence côté UI */ }
```

Boutique (boutique.html) :

[Accueil](#) [Le Club](#) [Inscriptions](#) [Terrains](#) [Médias](#) [Boutique](#) [Contact](#) [Mon espace](#)

Réserver un terrain

Choisissez une date, un terrain et un créneau pour jouer.

Nouvelle réservation

Date

Terrain

Début

Durée

Réserver

Effacer la journée

13/05/2026

Salle 1

18:00

2 h

Mes réservations

Salle 1

2026-05-13

18:00 – 20:00

Annuler

© 2025 Tennis Club de Rambouillet — Tous droits réservés

Extrait du code :

```
<main class="container shop-layout">
  <!-- PRODUITS: chaque article contient image, titre, description, prix et contrôles -->
  <section class="products" aria-label="Liste de produits" id="products-container">
    <p style="text-align:center;color:#666;padding:40px">Chargement des produits...</p>
  </section>

  <!-- PANIER: liste d'items ajoutés + récapitulatif + paiement -->
  <aside class="cart" aria-label="Panier">
    <h2>Panier</h2>
    <p class="cart-empty">Votre panier est vide.</p>
    <ul class="cart-list" hidden></ul>

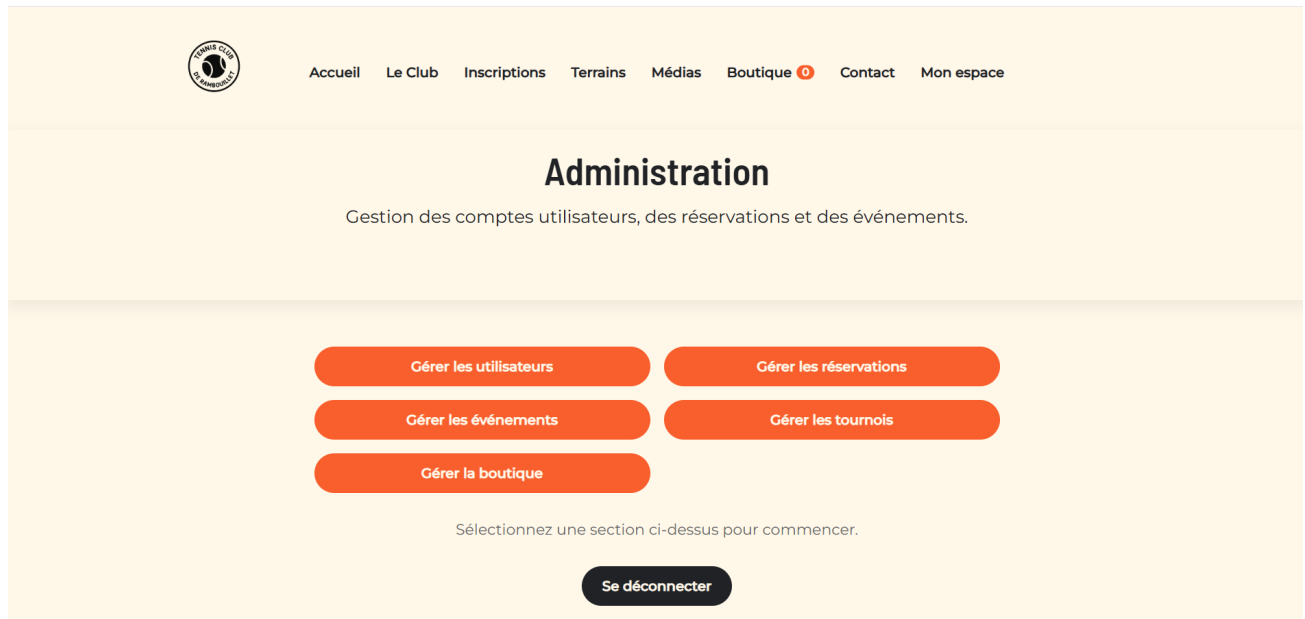
    <div class="cart-summary" hidden>
      <div class="summary-row"><span>Sous-total</span><span id="sum-subtotal">0,00 €</span></div>
      <div class="summary-row"><span>Livraison <small>(forfait)</small></span><span id="sum-shipping">0,00 €</span></div>
      <div class="summary-row total"><span>Total</span><span id="sum-total">0,00 €</span></div>
      <!-- Validation: redirection vers la page de paiement (démonstration carte) -->
      <a class="btn-demo" href="checkout.html" id="btn-checkout">Valider le panier</a>
    </div>
  </aside>
</main>

<footer>
  <div class="container">
    <p>&copy; 2025 Tennis Club de Rambouillet – Tous droits réservés</p>
  </div>
</footer>

<!-- Paiement: démonstration via page checkout (pas de PayPal en mode école) -->

<script src="js/nav-badge.js"></script>
<script src="js/boutique.js"></script>
</body>
</html>
```

Panel d'administration :



Extrait du code :

```
// Gestion de la création d'utilisateur depuis l'admin
$createError = '';
$eventError = '';
$eventSuccess = '';

// Gestion des événements
if($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['action']) && $_POST['action'] === 'create_event'){
    $title = trim($_POST['event_title'] ?? '');
    $description = trim($_POST['event_description'] ?? '');
    $event_date = trim($_POST['event_date'] ?? '');
    $event_time = trim($_POST['event_time'] ?? '');
    $location = trim($_POST['event_location'] ?? '');

    if(empty($title) || empty($event_date)){
        $eventError = 'Le titre et la date sont obligatoires.';
    } else {
        $stmt = $pdo->prepare("INSERT INTO events (title, description, event_date, event_time, location) VALUES (?, ?, ?, ?, ?)");
        $stmt->execute([
            $title,
            $description ?: null,
            $event_date,
            $event_time ?: null,
            $location ?: null
        ]);
        $eventSuccess = 'Événement créé avec succès.';
    }
}

if($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['action']) && $_POST['action'] === 'delete_event'){
    $event_id = intval($_POST['event_id'] ?? 0);
    if($event_id > 0){
        $pdo->prepare("DELETE FROM events WHERE id = ?")->execute([$event_id]);
        $eventSuccess = 'Événement supprimé.';
    }
}
```

BTS SERVICES INFORMATIQUES AUX ORGANISATIONS SESSION 2026 ANNEXE
VII-1-B : Fiche descriptive de réalisation professionnelle (recto) Épreuve E6 -
Conception et développement d'applications (option SLAM)

DESCRIPTION D'UNE RÉALISATION PROFESSIONNELLE		N° réalisation : 2
Nom, prénom : TARDY Samuel		N° candidat : 02541476314
Épreuve ponctuelle ☒ Contrôle en cours de formation ☐		Date : 04/04/2026
Organisation support de la réalisation professionnelle RMS Ticket (entreprise fictive)		
Intitulé de la réalisation professionnelle Application de ticketing pour la société RMS Ticket.		
Période de réalisation : 2025 / 2026 Lieu : Lycée Louis Bascan Modalité : ☐ Seul(e) ☒ En équipe		
Compétences travaillées ☒ Concevoir et développer une solution applicative ☒ Assurer la maintenance corrective ou évolutive d'une solution applicative ☒ Gérer les données		
Conditions de réalisation (ressources fournies, résultats attendus) La réalisation professionnelle a été développée à partir de zéro. Résultat attendu : Application web opérationnelle : création, suivi et clôture de tickets de support informatique Système d'authentification sécurisée avec gestion des droits (RBAC : 4 rôles, 20 permissions) Interface d'administration complète (utilisateurs, roles, permissions, tickets, journal) Messagerie interne par ticket pour les échanges utilisateur / technicien Notifications dynamiques des messages non lus (badge en temps réel) Base de données relationnelle normalisée (8 tables), importable via un script SQL		
Description des ressources documentaires, matérielles et logicielles utilisées Hébergement en local sur le Proxmox (XAMPP / Apache). Accessible en ligne sur samueltardy.fr Langages : HTML5, CSS3, JavaScript (vanilla), PHP (sans framework, PDO) Serveur local : Apache via XAMPP Base de données : MariaDB 10.4 (encodage utf8mb4) Éditeur de code : Visual Studio Code Modalités d'accès aux productions et à leur documentation Code source de l'application : https://github.com/Sasukisse/RMS-Ticket samueltardy.fr		

Déploiement local :

1. Copier le projet dans htdocs
2. Créer la base rms_ticket dans phpMyAdmin et importer bdd.sql
3. Vérifier config.php (config XAMPP par défaut, rien à changer en général)
4. Démarrer Apache + MySQL et aller sur <http://localhost/RMS-Ticket-main>

Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs

RMS Ticket est une application web de gestion de tickets de support informatique développée de A à Z en PHP/MySQL sans framework tiers. Elle permet à tout utilisateur authentifié de créer un ticket, de suivre son avancement et d'échanger directement avec le technicien via une messagerie intégrée. Un panneau d'administration complet permet au personnel technique de gérer les utilisateurs, les rôles, les permissions (RBAC), les tickets et de consulter un journal d'activité horodaté. Le code source est disponible sur GitHub.

Modèle conceptuel des données :

La base de données MariaDB rms_ticket est composée de 8 tables :

Table	Principaux attributs	Rôle
users	<i>id, username, nom, prenom, email, password_hash, droit</i>	Comptes utilisateurs
roles	<i>id, name, display_name, description</i>	Roles (super_admin, admin, support, user)
permissions	<i>id, name, display_name, category</i>	20 permissions granulaires
role_permissions	<i>role_id, permission_id</i>	Association N:N roles / permissions
tickets	<i>id, user_id, assigned_to, title, category, type, priority, status</i>	Tickets de support
ticket_responses	<i>id, ticket_id, user_id, response_text, is_admin_response</i>	Messages liés à un ticket
ticket_message_reads	<i>ticket_id, user_id, last_read_at</i>	Suivi lectures (badges non lus)
admin_logs	<i>id, user_id, action, entity_type, details, ip_address</i>	Journal des actions d'administration

Arborescence du projet :

RMS-Ticket-main/

index.php ← Redirection vers la connexion
bdd.sql ← Script SQL initialisation base

Database/

config.php ← Constantes configuration BDD
connection.php ← Connexion PDO à MariaDB

Login/

login.php ← Authentification (session PHP)
login.css
login.js

HomePage/

index.php ← Page d'accueil navigation dynamique
logout.php
assets/

CreateTickets/

create_ticket.php ← Formulaire création ticket
app.js
style.css

Tickets/

my_tickets.php ← Liste des tickets utilisateur
ticket.php ← Vue détaillée d'un ticket
chat_api.php ← API JSON messagerie
notifications_api.php ← API badges non lus
mark_read.php ← Marquage de lecture

AdminPanel/

adminpanel.php ← Panneau d'administration complet
adminpanel.css
adminpanel.js

Fonctionnement général et workflows :

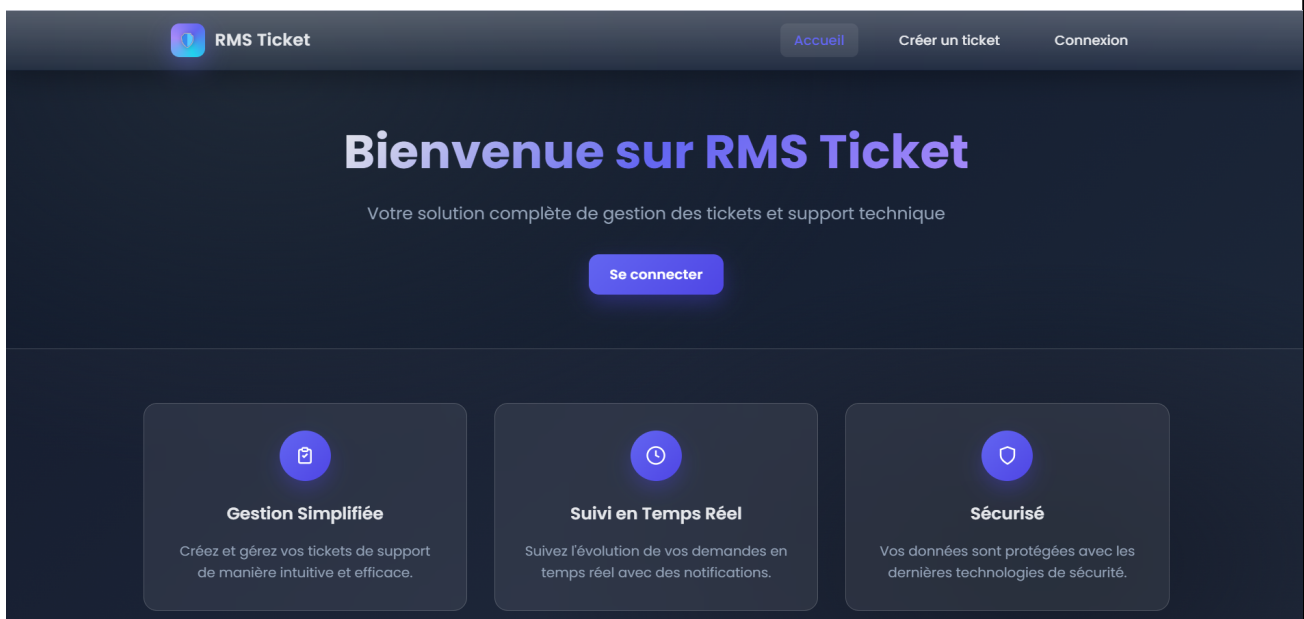
Fonctionnalité	Description
Authentification	Connexion email/mot de passe -> password_verify (bcrypt) -> session PHP. Redirection selon les droits.
Création de ticket	Formulaire valide côté serveur : titre, description, catégorie (Matériel/Logiciel/Réseau/Autre) et type (Incident/Demande).
Suivi des tickets	Vue Mes tickets avec statut colore. Vue détaillée avec modification et clôture (verrouille si ticket ferme).
Messagerie intégrée	API JSON (chat_api.php) : envoi et récupération de messages. Polling JS quasi-temps réel. Distinction visuelle utilisateur/technicien.
Notifications	Badge rouge calcule via notifications_api.php dans la barre de navigation. Mis à jour via mark_read.php.
Panel administration	CRUD utilisateurs, RBAC rôles/permissions, assignation tickets, statut/priorité, journal admin_logs.

Sécurité appliquée :

Mesure	Détail
Mots de passe	PASSWORD_BCRYPT + password_verify (aucun mot de passe stocké en clair)
Requêtes SQL	PDO avec prepare/execute sur toutes les requêtes (protection injection SQL)
Sorties HTML	htmlspecialchars() systématique sur toutes les données affichées (protection XSS)
En-têtes HTTP	X-Frame-Options: SAMEORIGIN, X-Content-Type-Options: nosniff, Referrer-Policy: no-referrer
Contrôle accès	Vérification \$_SESSION['droit'] + permissions RBAC avant chaque opération sensible

Ressources :

Page d'accueil (index.php) :



Extrait du code :

```
<!-- Contenu principal -->
<main class="main-content">
  <div class="hero-section">
    <div class="hero-content">
      <h1 class="hero-title">Bienvenue sur RMS Ticket</h1>
      <p class="hero-subtitle">
        Votre solution complète de gestion des tickets et support technique
      </p>

      <?php if (isset($_SESSION['user_id'])): ?>
        <div class="welcome-user">
          <p>Bonjour <strong><?php echo htmlspecialchars($_SESSION['prenom']) . ' ' . $_SESSION['nom']; ?></strong> !</p>
          <p>Vous êtes connecté en tant que <em><?php echo htmlspecialchars($_SESSION['username']); ?></em></p>
        </div>
      <?php endif; ?>

      <div class="action-buttons">
        <?php if (isset($_SESSION['user_id'])): ?>
          <a href="../CreateTickets/create_ticket.php" class="btn btn-primary">
            Créer un ticket
          </a>

          <?php else: ?>
            <a href="../Login/login.php" class="btn btn-primary">
              Se connecter
            </a>
          <?php endif; ?>
        </div>
      </div>
```

Création d'un ticket (create_ticket.php) :

The screenshot shows a web interface for creating a support ticket. The header includes the 'RMS Ticket' logo and navigation links: 'Accueil', 'Créer un ticket' (highlighted), 'Mes tickets', and 'Déconnexion'. The main form is titled 'Créer un ticket' and contains the following fields:

- Type:** Two radio button options: 'Incident' (with a fire icon) and 'Demande' (with a document icon). Below each option is a brief description.
- Catégorie:** A dropdown menu with the placeholder text 'Choisir une catégorie'.
- Nom & Prénom:** A text input field containing 'Mathieu Gallienne'.
- Email:** A text input field containing 'mathieu.gallienne@user.fr'.
- Titre du ticket:** A label for the final title field, which is partially visible at the bottom.

Extrait du code :

```
<?php
session_start();
include '../Database/connection.php';

// Vérifier si l'utilisateur est connecté
if (!isset($_SESSION['user_id'])) {
    header('Location: ../Login/login.php');
    exit();
}

// Gestion soumission du formulaire
$success = false;
$error = false;
$errors = [];

if ($_SERVER["REQUEST_METHOD"] === "POST") {
    $title = trim($_POST["title"] ?? '');
    $description = trim($_POST["description"] ?? '');
    $category = trim($_POST["category"] ?? '');
    $type = trim($_POST["type"] ?? '');

    // Validation des données
    if (empty($title)) { $errors[] = "Le titre est obligatoire."; }
    if (empty($description)) { $errors[] = "La description est obligatoire."; }
    if (empty($category)) { $errors[] = "La catégorie est obligatoire."; }
    if (empty($type)) { $errors[] = "Le type est obligatoire."; }

    if ($title !== '' && mb_strlen($title) < 4) { $errors[] = "Le titre doit contenir au moins 4 caractères."; }
    if ($description !== '' && mb_strlen($description) < 10) { $errors[] = "La description doit contenir au moins 10 caractères."; }

    if (empty($errors)) {
        try {
```