Rapport de projet SAÉ 23/24

Thalès14

Groupe :
Al-Abri Mohammed
Caillet Adrien
Hacherouf Sarah
Laurent Arthur
Zoubir Rayan

Table des matières

I. Présentation du projet

The aim of this project, proposed by Thales as part of our SAE, is to create a system capable of automatically or manually taking photos of an avionics test bench, in order to archive its status in a database before each test.

The system we have developed can take a photo on demand, or automatically if one has not been taken for 24 hours. It works day and night, with integrated lighting if required. Photos are stored and consultable via a local website, secured by a login system.

Users can log in, change their password, take a photo, view stored photos or mark them for deletion. Administrators can manage accounts, change passwords, unblock users, rename programs, delete photos and access system logs.

This project reinforces bench traceability, secures test campaigns and improves the reliability of the results obtained.

Translated with DeepL.com (free version)

II. Les Exigences :

#WEB01 - Création d'un site web

#ACC01 – Accessibilité

• #ACC01-01 : L'accès à l'application doit être sécurisé par un système de login/mot de passe.

• #ACC01-02 : L'accès aux fonctionnalités doit être limité selon le profil utilisateur :

• Super Administrateur.

• Administrateur.

• Opérateur.

II.c #LOG01 - Gestion des traces d'actions (logs)

• #LOG01-01 : L'application doit enregistrer tous les événements liés à son utilisation.

• #LOG01-02 : Les événements doivent être classés selon leur importance :

• Informations.

• Warnings.

• Alarmes.

• #LOG01-03 : Chaque événement doit inclure :

• Date.

• Identifiant de l'utilisateur connecté.

• Profil de l'utilisateur.

• Type d'événement.

• Description de l'action effectuée.

#SEC01 - Sécurité des mots de passe

• #SEC01-01 : Le mot de passe doit contenir au moins **n caractères numériques** (entre "0" et "9").

• #SEC01-02 : Le mot de passe doit contenir au moins **p caractères alphabétiques en minuscule** (entre "a" et "z").

• #SEC01-03 : Le mot de passe doit contenir au moins **q caractères alphabétiques en majuscule** (entre "A" et "Z").

• #SEC01-04 : Le mot de passe doit contenir au moins **r caractères spéciaux** parmi les caractères autorisés.

• #SEC01-05 : Le mot de passe ne doit pas contenir d'accents.

• #SEC01-06 : Le mot de passe ne doit pas contenir le login de l'utilisateur.

• #SEC01-07 : Le mot de passe doit être stocké sous une forme chiffrée.

• #SEC01-08 : Le compte doit être bloqué après **3 tentatives de connexion infructueuses**.

• #SEC01-09 : Les paramètres n, p, q, et r doivent être configurables uniquement par un Administrateur.

#USR01 - Gestion des utilisateurs par un Administrateur

• #USR01-01 : Un Administrateur doit pouvoir ajouter ou supprimer des utilisateurs de type Opérateur.

• #USR01-02 : Un Administrateur doit pouvoir définir le mot de passe d'un nouvel utilisateur de type Opérateur.

• #USR01-03 : Un Administrateur doit pouvoir ajouter ou supprimer des utilisateurs de type Administrateur.

• #USR01-04 : Un Administrateur doit pouvoir définir le mot de passe d'un nouvel Administrateur.

• #USR01-05 : Un Administrateur doit pouvoir modifier les paramètres de configuration de l'application.

• #USR01-06 : Un Administrateur doit pouvoir réinitialiser le mot de passe d'un compte verrouillé.

#SADM01 - Gestion des privilèges du Super Administrateur

• #SADM01-01 : L'application doit permettre un seul utilisateur de type Super Administrateur.

• #SADM01-02 : Le compte du Super Administrateur ne doit jamais être bloqué.

• #SADM01-03 : Le Super Administrateur doit avoir les mêmes privilèges qu'un Administrateur.

Le login et mot de passe du Super Administrateur doivent être communiqués oralement au tuteur.`

Exigences techniques supplémentaires :

• **#WEB02** : L'application doit être compatible avec les navigateurs courants (Chrome, Firefox, etc.).

• **#SEC02** : Les données sensibles (logs, mots de passe, etc.) doivent être stockées dans une base de données sécurisée.

• **#INT01** : Les utilisateurs doivent pouvoir accéder à l'application via une interface simple et intuitive.

• **#PERF01** : L'application doit pouvoir gérer au moins 10 utilisateurs simultanés.

Conformité aux standards :

• **#STD01** : L'application doit respecter les normes RGPD concernant la protection des données personnelles.

• **#LOG02** : Les logs utilisateurs doivent être conservés pour une période maximale configurable (par défaut, 6 mois).

6BUT Réseaux et Télécommunications

Création d'un système de prise de photo :

Accessibilité :

• **#PHO01** : Capacité de prendre une photo depuis le site en se connectant.

• **#PHO02** : Capacité de prendre une photo depuis le site sans se connecter.

• **#PHO03** : Capacité de prendre une photo automatiquement après 24 heures sans action Utilisateur.

• **#PHO04** : Capacité de prendre une photo via un programme Python avant/après un test.

• **#PHO05** : Enregistrement ou suppression et reprise de la photo par l'utilisateur.

Fonctionnement :

• **#LED01** : Allumage de la LED automatique si l'environnement est trop sombre.

• **#LED02** : Application du principe de l'hystérésis entre le capteur de luminosité et là LED.

• **#LED03** : Allumage de la LED manuellement par un utilisateur.

III. Programme informatique

Le site web développé dans le cadre de ce projet avait pour objectif principal de proposer une interface simple, fonctionnelle et sécurisée permettant d'interagir avec un banc de test avionique. Il devait permettre :

- Utilisateurs autorisés de déclencher à distance une prise de photo via une interface web,

- Gérer les utilisateurs selon différents rôles avec des droits bien définis (invité, opérateur, administrateur, super administrateur),

- Visualiser, télécharger ou supprimer les photos prises (mise en corbeille, suppression définitive),

- Connecter le site à la base de données afin d'enregistrer toutes les informations liées aux photos (titre, date, heure, chemin, utilisateur),

- Enregistrer automatiquement les actions importantes dans un journal (log) consultable depuis le back-office,

- Afficher dynamiquement les photos selon des critères de recherche (date, titre, ID, utilisateur…).

Le tout devait être développé en PHP côté serveur, avec une base de données SQLite déjà structurée, et un fonctionnement compatible avec un serveur Apache2 sur Raspberry Pi.

Le site web développé répond parfaitement aux besoins d'accessibilité, de gestion des utilisateurs et de gestion des photos définis dans le cahier des charges. En proposant une interface simple et sécurisée, il respecte l'exigence #ACC01 qui impose une authentification par login et mot de passe, ainsi que la gestion des profils avec des droits spécifiques (invité, opérateur, administrateur, super administrateur), conformément aux exigences #ACC01-01 et #ACC01-02. La possibilité pour les utilisateurs autorisés de déclencher une prise de photo via l'interface répond directement à l'exigence #PHO01, tandis que la visualisation, le téléchargement et la suppression des photos correspondent aux besoins fonctionnels #PHO05 et à la gestion des droits utilisateurs (#USR01 et #SADM01). La connexion au système de base de données SQLite garantit un stockage sécurisé et structuré des métadonnées liées aux photos, répondant aux exigences #SEC02 et #WEB01. Enfin, l'enregistrement automatique des actions importantes dans un journal consultable assure la traçabilité attendue selon les exigences #LOG01, et l'affichage dynamique des photos permet une utilisation fluide et efficace conforme à #INT01 et #WEB02. L'utilisation de PHP côté serveur avec un serveur Apache2 sur Raspberry Pi assure une solution compatible et performante, répondant aux contraintes techniques du projet.

i. Exigences principales respectées

- Interface responsive en HTML/CSS, pensée pour une utilisation locale rapide.

- Connexion à la base de données fonctionnelle (consultation, insertion, mise à jour).

- Séparation des rôles avec restrictions d'accès selon les droits.

- Journalisation automatique des actions critiques (prise de photo, suppression).

- Affichage des images sous forme de miniatures cliquables.

- Communication avec le Raspberry Pi pour écrire dans un fichier commande.json, lu ensuite par un script Python.

- Gestion des erreurs, des permissions et des redirections selon les rôles.

Interface de back-office pour les administrateurs (visualisation des logs et des photos supprimées).

ii. Page de Connexion

1. **Présentation générale** :

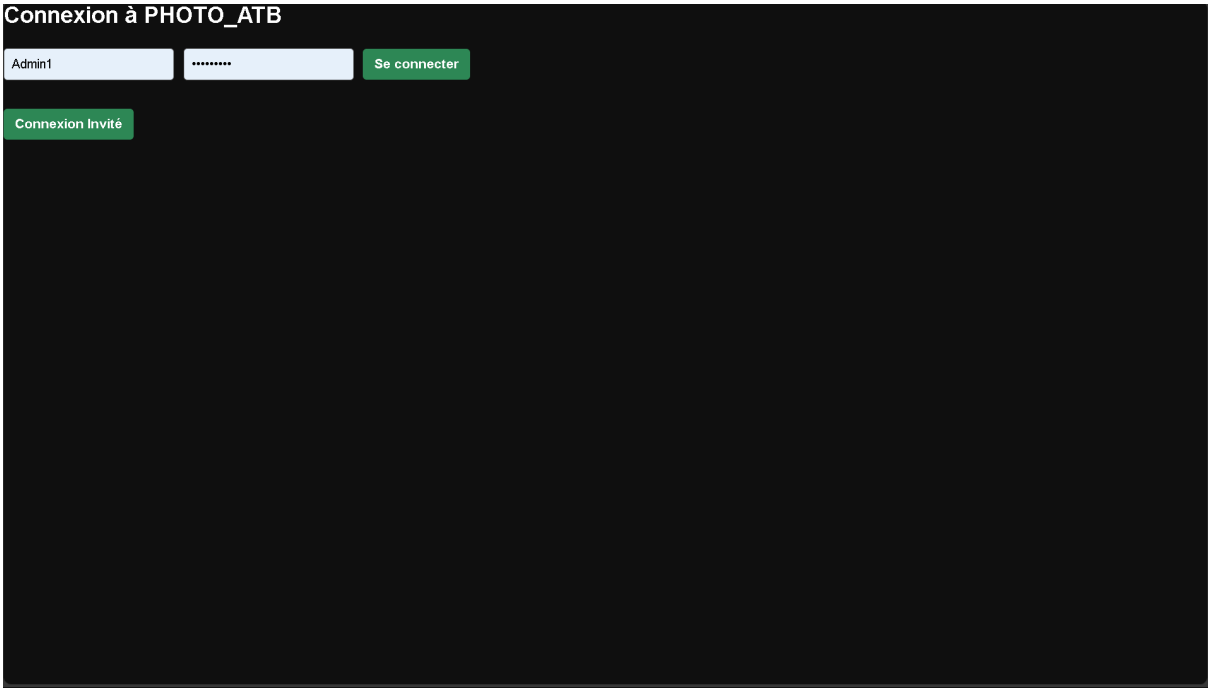


Figure 1 – Page de connexion

Cette interface est la première page du site PHOTO_ATB. Elle permet aux utilisateurs de se connecter en renseignant leur identifiant (login) et leur mot de passe (password). Elle propose également un bouton de connexion en tant qu'invité, qui donne accès à une version limitée du site (prise de photo uniquement).

L'apparence de cette page a été réalisée en HTML/CSS, avec une mise en page responsive et sobre. Le style sombre a été volontairement choisi pour s'intégrer à un environnement technique de laboratoire et rester lisible en conditions de faible luminosité.

iii. Fonctionnement interne (non visible)

- Lorsqu'un utilisateur tente de se connecter, le système vérifie les identifiants à l'aide d'une requête vers la base de données.

- En cas d'erreur, un compteur tentatives est incrémenté automatiquement dans la table USER.

- Lorsque ce compteur atteint 3 tentatives, le compte est bloqué (bloque = 'oui'), empêchant toute connexion ultérieure, sauf intervention manuelle.

- Le Super Administrateur ne peut jamais être bloqué, même après plusieurs erreurs. Ce comportement est géré dans le code en excluant explicitement l'utilisateur Admin1 de cette logique de blocage.

- Chaque tentative (réussie ou échouée) est enregistrée dans une table de logs (LOGS) contenant :

    - L'identifiant,

    - La date/heure,

    - Le type d'action ("connexion"),

    - Un message de statut.

La gestion de la connexion des utilisateurs est conçue pour renforcer la sécurité conformément aux exigences du projet. La vérification des identifiants via une requête vers la base de données assure une authentification fiable, répondant à l'exigence #ACC01-01. Le compteur de tentatives d'échec et le blocage automatique du compte après trois erreurs mettent en œuvre la contrainte de sécurité #SEC01-08, garantissant la protection contre les tentatives d'accès non autorisées. L'exception spécifique pour le Super Administrateur, qui ne peut jamais être bloqué, est conforme à l'exigence #SADM01-02. Par ailleurs, l'enregistrement systématique de chaque tentative de connexion, réussie ou non, dans la table de logs répond aux exigences #LOG01-01 et #LOG01-03, assurant la traçabilité complète des actions utilisateur avec les informations essentielles comme l'identifiant, la date/heure, le type d'action et un message de statut. Cette approche garantit à la fois la sécurité des accès et la transparence des opérations.

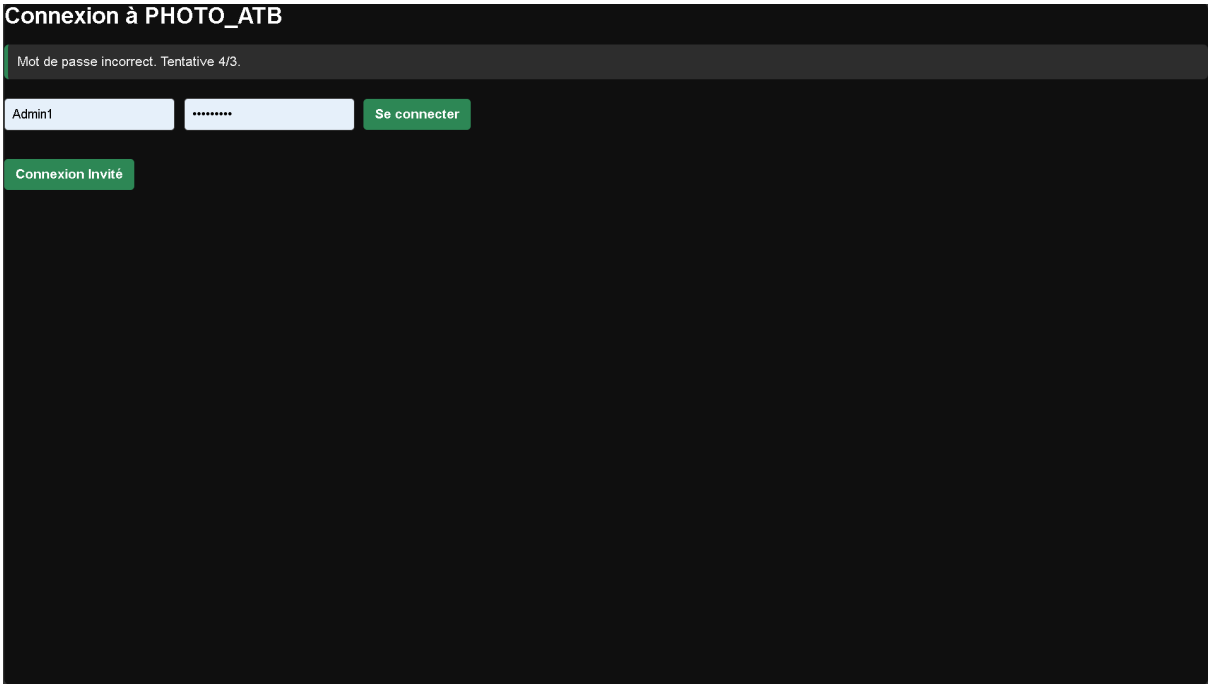


Figure 2 – Message d'erreur si Super Administrateur

1. Message d'erreur en cas d'échec

Quand un utilisateur se trompe de mot de passe, un message s'affiche au-dessus du formulaire :
"Mot de passe incorrect. Tentative 2/3."
Cette indication aide à comprendre l'imminence du blocage, tout en ne révélant aucune information sur la validité du login, ce qui est une bonne pratique en sécurité.

L'affichage d'un message clair indiquant le nombre de tentatives restantes avant le blocage du compte contribue à une bonne expérience utilisateur tout en renforçant la sécurité. Ce retour informe l'utilisateur de l'état de sa connexion sans révéler si le login est valide ou non, respectant ainsi les bonnes pratiques de sécurité pour éviter les attaques par force brute. Cette fonctionnalité répond aux exigences de sécurité implicites dans #SEC01-08 en limitant les risques d'informations sensibles divulguées, tout en améliorant la compréhension et la transparence du système pour l'utilisateur.

2. Déclenchement du blocage automatique

Le blocage automatique du compte après trois tentatives de mot de passe incorrectes est une mesure essentielle pour sécuriser l'accès à l'application et limiter les risques d'attaques par force brute, conformément à l'exigence #SEC01-08. Le champ « bloque » dans la base de données permet de suivre précisément l'état du compte, facilitant ainsi la gestion des utilisateurs.

À partir de ce moment-là, toute tentative de connexion affiche un message d'erreur clair :

"Compte bloqué. Contactez un administrateur."

Cette mesure assure une communication transparente avec l'utilisateur tout en l'incitant à solliciter une intervention manuelle pour débloquer son accès, ce qui correspond aux bonnes pratiques de sécurité et à l'exigence #USR01-06. Cette fonctionnalité garantit un contrôle efficace des accès et protège les données sensibles de l'application.

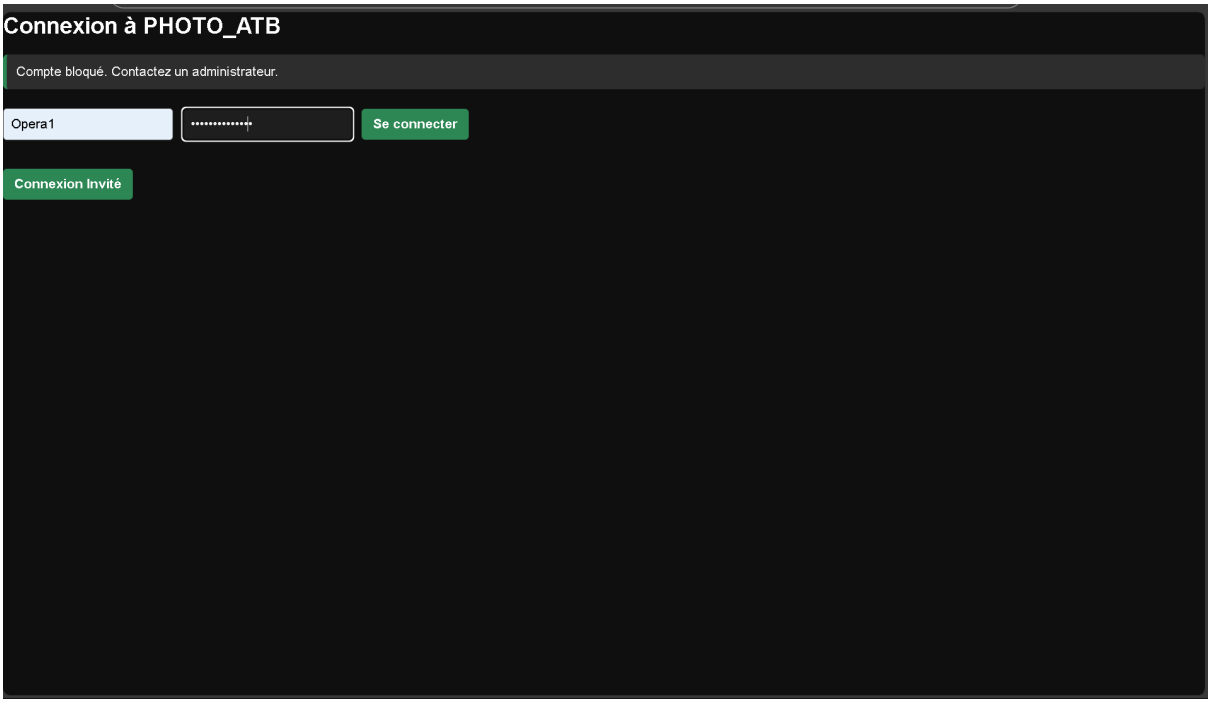


Figure 3 – Message d'erreur si l'utilisateur est bloqué

iv. Connexion Invité – Accès restreint au Mode Photo

1. Accès invité au site

La fonctionnalité « Connexion Invité » permet d'offrir un accès simplifié et contrôlé à la plateforme sans nécessité de créer un compte, ce qui facilite la démonstration ou la consultation rapide des informations. Ce rôle restreint garantit que les invités ne disposent que de permissions limitées, préservant ainsi la sécurité et l'intégrité des données, conformément à l'exigence #ACC01-02 qui prévoit une gestion fine des profils utilisateurs avec des droits adaptés. Cette solution contribue à rendre l'application plus accessible tout en maintenant un contrôle strict des accès.

2. Interface accessible

L'invité est redirigé vers une version simplifiée du site, composée uniquement de deux éléments dans le menu de gauche :

- Mode Photo
- Déconnexion

L'objectif est de limiter les interactions sensibles, tout en permettant de tester l'interface de déclenchement de photo.

On respecte l'exigence #ACC01-02 qui impose de limiter les fonctionnalités selon le profil utilisateur. La restriction des accès pour le rôle invité garantit une utilisation sécurisée tout en permettant un accès minimal à l'interface photo.

3. Fonctionnement de la page

L'affichage présenté comporte :

- Une zone d'aperçu caméra (fixe, sans flux en direct)
- Un bouton « Prendre une photo », actif
- Un sélecteur LED (On/Off/Auto) permettant de contrôler le déclencheur lumineux sur le matériel, via le Raspberry Pi

Dans la version actuelle, l'aperçu est statique et affiche la dernière photo prise par l'utilisateur. Cela est volontaire, car le Raspberry Pi n'a pas la puissance suffisante pour afficher un retour vidéo en temps réel dans le navigateur.

Cette interface répond aux exigences #PHO01 et #PHO02 en permettant à l'utilisateur de prendre une photo via le site, qu'il soit connecté ou non. Le choix d'un aperçu statique correspond à une contrainte matérielle liée aux capacités du Raspberry Pi, garantissant ainsi une expérience utilisateur fluide tout en assurant le contrôle manuel de la LED conformément aux exigences #LED01 et #LED03.

4. Actions disponibles pour un invité

L'invité peut :

- Déclencher une prise de photo : cela écrit un fichier JSON contenant les instructions à destination du Raspberry Pi.

- Choisir l'état de la LED au moment de la prise (utile pour les conditions d'éclairage).

- Se déconnecter à tout moment depuis le menu.

Cette fonctionnalité respecte les exigences #PHO02 pour permettre la prise de photo sans connexion, ainsi que #LED03 qui autorise le contrôle manuel de l'éclairage par l'utilisateur. Elle garantit aussi une gestion sécurisée des sessions, en conformité avec #ACC01-01 pour la déconnexion.

5. Restrictions spécifiques au rôle « invité »

- L'utilisateur ne peut pas accéder aux pages suivantes :
    - home.php (accueil personnalisé)
    - visionnage.php (liste des photos)
    - voir_photo.php (aperçu des photos)
    - back-office.php (gestion des utilisateurs, logs, paramètres)
- Toute tentative d'accès à ces pages redirige automatiquement vers mode-photo.php.
- L'invité n'est pas journalisé dans les logs à la prise de photo (aucun user_id stocké).
- Les photos prises par un invité sont tout de même stockées, mais ne sont pas liées à un identifiant personnel.

Cette gestion des accès répond strictement à l'exigence #ACC01-02 en limitant les droits de l'utilisateur invité aux seules fonctionnalités autorisées. Le fait de rediriger les accès non autorisés renforce la sécurité et la robustesse de l'application. Par ailleurs, la non-journalisation des actions des invités respecte la confidentialité tout en assurant la traçabilité minimale requise par #LOG01-01.

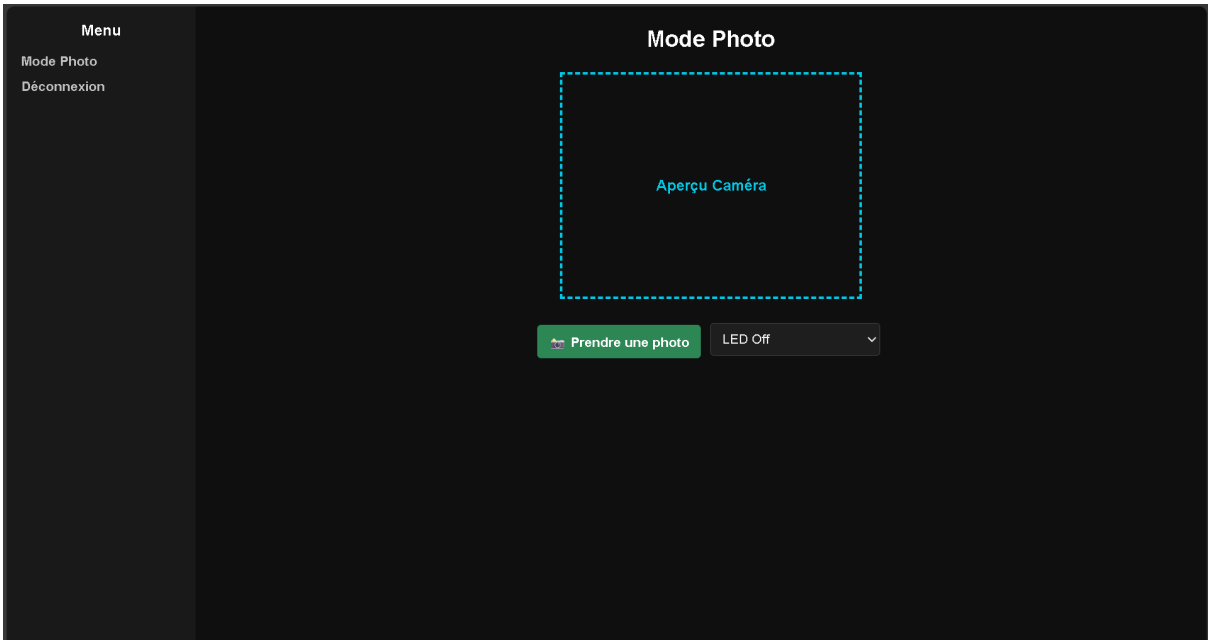


Figure 4 – Page du mode invité (accès limité)

v. Page d'Accueil – Espace personnalisé des utilisateurs connectés

1. Présentation générale

Après une connexion réussie avec un identifiant administrateur, opérateur ou super administrateur, l'utilisateur est redirigé vers la page d'accueil personnalisée. Cette interface centralise l'accès aux dernières photos prises, ainsi qu'aux menus complets selon les droits du rôle.

La page débute par un message de bienvenue affichant le nom de l'utilisateur et son rôle, ce qui permet d'identifier immédiatement le niveau de privilège en cours

Cette organisation respecte les exigences #ACC01-01 et #ACC01-02 en assurant une connexion sécurisée et une gestion des droits adaptée à chaque profil utilisateur. L'affichage personnalisé améliore l'expérience utilisateur tout en garantissant un accès contrôlé aux fonctionnalités en fonction des privilèges attribués.

2. Données affichées

Le tableau présente les 5 dernières photos enregistrées dans la base de données, triées par date décroissante. Chaque photo est affichée avec les éléments suivants :

- Miniature de la photo
- Date de capture
- Heure de capture
- ID Photo
- Titre de la photo
- Nom de l'utilisateur ayant déclenché la photo
- Bouton de suppression logique

Ce tableau répond aux exigences #PHO05 et #WEB01 en permettant à l'utilisateur de visualiser facilement les photos récentes, avec des informations complètes pour chaque élément. La présence d'un bouton de suppression logique garantit la gestion sécurisée des fichiers, conformément à la politique de conservation et suppression des données.

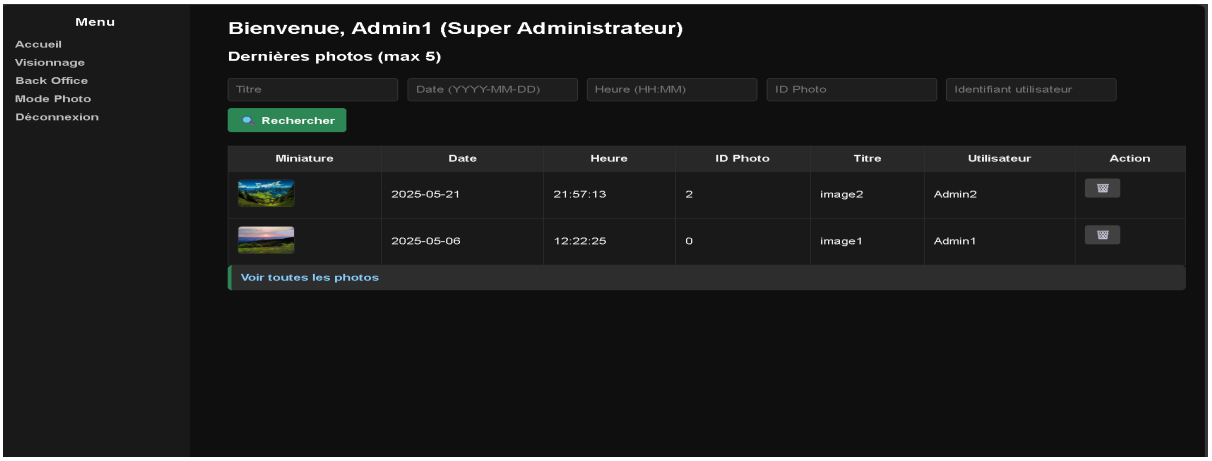


Figure 5 – Page d'accueil après connexion

3. Système de recherche multicritère

Un bandeau de recherche en haut permet de filtrer les photos sur plusieurs critères :

- Par titre
- Par date exacte (format AAAA-MM-JJ)
- Par heure précise (format HH:MM)

- Par ID de la photo

- Par identifiant utilisateur

La recherche se fait côté serveur en PHP via des requêtes paramétrées vers SQLite, filtrant dynamiquement les résultats du tableau.

Ce système de filtrage dynamique répond à l'exigence #INT01 en offrant une interface simple et intuitive pour naviguer dans l'historique des photos. La réalisation côté serveur garantit la sécurité des requêtes et la robustesse du traitement, conformément aux bonnes pratiques de développement.

4. Règles d'accès

- Cette page est inaccessible aux invités : une tentative d'accès en tant qu'invité redirige automatiquement vers la page mode-photo.php.

- Seuls les utilisateurs ayant un rôle valide (Opérateur, Administrateur, Super Administrateur) peuvent y accéder.

Cette restriction d'accès respecte l'exigence #ACC01-02 en limitant les fonctionnalités selon le profil utilisateur, assurant ainsi la sécurité et la confidentialité des données en empêchant les invités d'accéder aux pages sensibles du site.

5. Suppression

La suppression est logique. Cela signifie que les photos sont simplement marquées comme supprimées dans la base (supr = 'oui'), mais ne sont pas effacées physiquement du système de fichiers. Elles restent consultables depuis le back-office**.**

Cette méthode de suppression répond à l'exigence #WEB01 en permettant une gestion sécurisée et réversible des photos, tout en assurant la traçabilité et la conservation des données importantes conformément aux bonnes pratiques de gestion des fichiers.

6. Sécurité et robustesse

- Les entrées utilisateurs (filtres) sont systématiquement échappées et nettoyées pour éviter les injections SQL ou XSS.

- L'action de suppression déclenche une journalisation complète dans la table LOGS (avec utilisateur, date, heure, type d'action, etc.).

Ces mesures garantissent la conformité aux exigences #SEC02 pour la sécurité des données et #LOG01 pour la gestion complète des traces d'actions, assurant ainsi la protection contre les attaques et la traçabilité des opérations sensibles.

vi. Page de Visionnage

1. Présentation générale

La page de visionnage est accessible à tous les utilisateurs authentifiés (Opérateurs, Administrateurs, Super Administrateurs). Elle permet de consulter l'ensemble des photos enregistrées, avec la possibilité de les trier, les filtrer ou les consulter individuellement en plein écran. Elle sert également de point d'accès aux fonctions de suppression logique des photos.

2. Fonctions principales

Chaque ligne du tableau affiche une photo enregistrée avec les informations suivantes :

- Miniature (affichage réduit de la photo)
- Date de capture
- Heure de capture
- ID unique de la photo (clé primaire en base)
- Titre donné à la photo (enregistré depuis le script de prise de photo)
- Nom de l'utilisateur à l'origine de la capture
- Lien "Voir" permettant d'ouvrir la photo en grand format
- Icône de corbeille permettant la suppression logique

3. Système de filtrage

En haut de la page, plusieurs champs de recherche permettent de filtrer les résultats :

- Par titre
- Par date précise (AAAA-MM-JJ)
- Par heure
- Par ID Photo
- Par identifiant utilisateur

Le filtrage repose sur des requêtes sécurisées vers la base SQLite. Les données sont échappées et validées avant exécution**.**

4. Sécurité et gestion des droits

- Les invités n'ont pas accès à cette page (redirection automatique).

- Comme sur la page d’accueil, la suppression est logique : la photo n’est pas supprimée physiquement, mais son champ supr est passé à 'oui' dans la table PHOTO.

5. Journalisation

Chaque suppression effectuée par un utilisateur est automatiquement enregistrée dans la table LOGS, avec les informations suivantes :

- ID de l’utilisateur

- Date et heure de l’action

- Type de log : Information

- Détail de l’action : "Photo mise à la corbeille ID X"

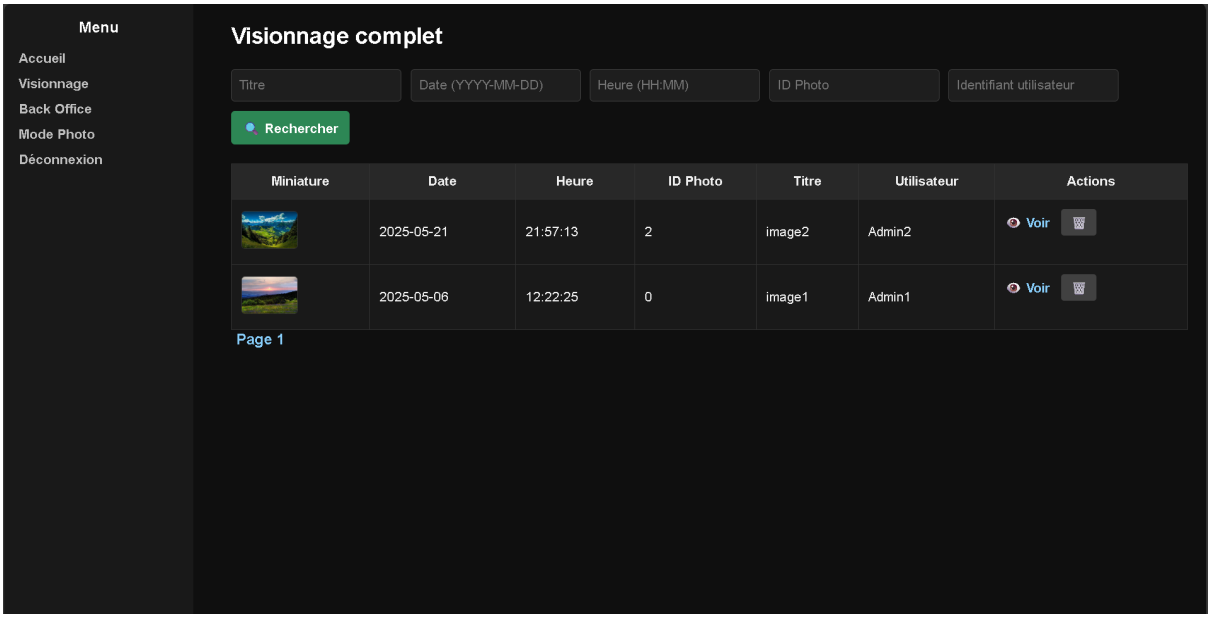
Cela permet un suivi intégral de toutes les suppressions et actions sensibles.

6. Accès à la consultation en grand

En cliquant sur le lien "Voir", l’utilisateur est redirigé vers une page spécifique (voir_photo.php) qui affiche la photo en grand format, accompagnée de ses métadonnées (titre, utilisateur, date, heure). Cette fonction est accessible à tous les utilisateurs.

7. Pagination

Si le nombre de photos dépasse 10, un système de pagination dynamique permet de naviguer entre les pages de résultats, avec indication de la page actuelle. La logique utilisée évite la surcharge mémoire et permet une expérience fluide même avec de nombreuses entrées.



vii. Page de Consultation Individuelle d'une Photo

1. Présentation générale

Cette page permet à un utilisateur connecté de consulter une photo en plein écran, accompagnée de ses métadonnées essentielles (titre, date, heure et utilisateur associé). Elle s'affiche à la suite d'un clic sur le bouton "Voir" présent dans la page visionnage.php.

2. Fonctionnalités accessibles

Une fois la page ouverte, l'utilisateur peut :

- Visualiser la photo en grand format.
- Consulter les informations de prise de vue :
    - Date
    - Heure
    - Identifiant de l'utilisateur à l'origine de la prise.
- Télécharger l'image au format original, via un lien direct.
- Revenir à la liste précédente (visionnage) via le lien ← Retour au visionnage.

Cette interface répond aux exigences #WEB01 pour une navigation intuitive et complète, ainsi qu'à #ACC01-02 en restreignant l'accès aux utilisateurs autorisés, tout en offrant des fonctionnalités essentielles pour la gestion et la consultation des photos.

Le lien de téléchargement est sécurisé et pointé vers le dossier protégé /assets/images/.

3. Sécurité et contrôle d'accès

- L'accès à cette page est interdit aux utilisateurs non connectés (invités).

- Seules les photos non supprimées sont consultables. Les photos passées en corbeille (champ supr = 'oui') ne peuvent pas être affichées ici.

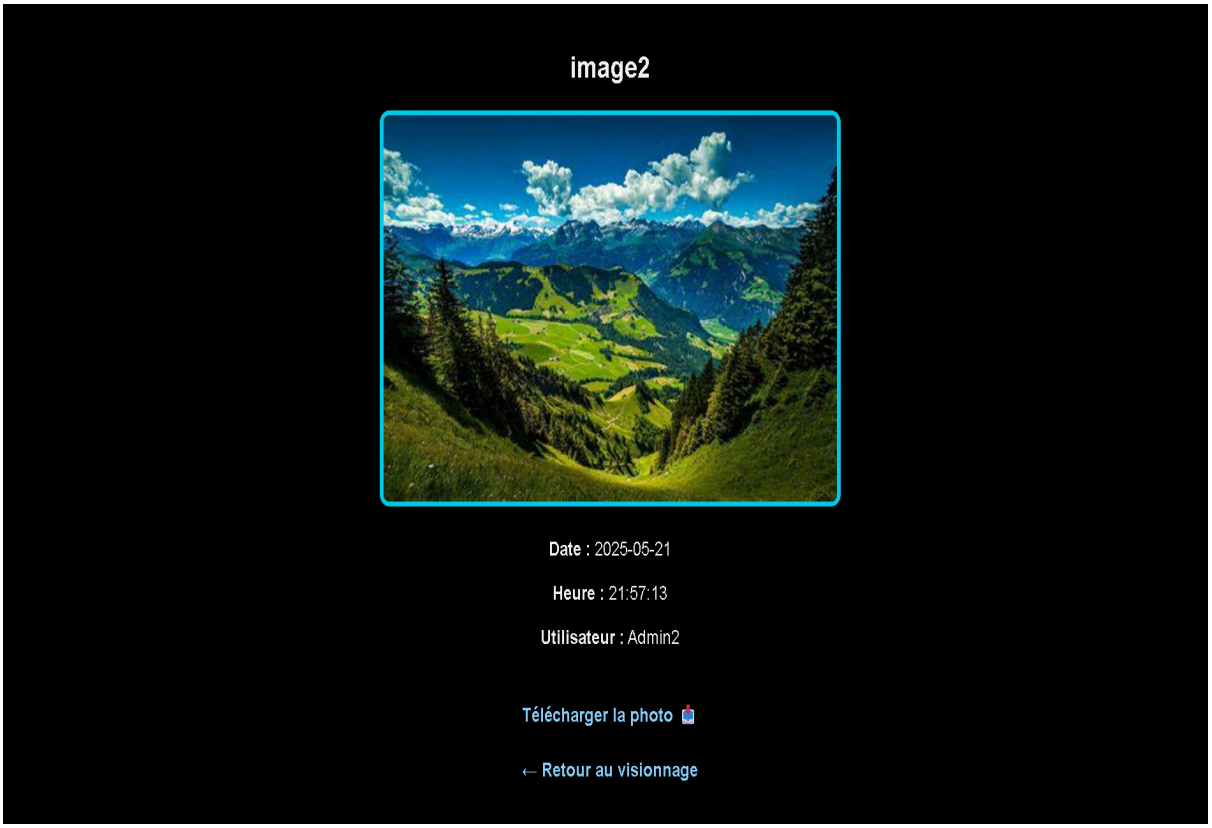


Figure 7 – Aperçu d'une photo en taille réelle

viii. Page de Prise de Photo

1. Objectif de la page

Cette page permet à un utilisateur connecté (Opérateur, Administrateur ou Super Administrateur) de prendre une photo depuis le système Raspberry Pi relié à un PICO WH. Elle constitue l'interface principale de capture pour le projet PHOTO_ATB.

2. Fonctionnalités disponibles

Sur cette interface, l'utilisateur peut :

- Déclencher une photo grâce au bouton "Prendre une photo".
- Choisir l'état de la LED du dispositif (On ou Off), qui est transmis via un fichier commande.json au Raspberry Pi.
- Prévisualiser la dernière photo prise, affichée directement dans l'encadré de droite.
- Supprimer la photo affichée, en la mettant dans la corbeille logique (supr = 'oui' dans la base de données), sans suppression physique.

La suppression est disponible uniquement immédiatement après la prise, pour éviter une mauvaise manipulation sur une autre photo.

3. Communication avec le Raspberry Pi et le PICO

- Lors du clic sur "Prendre une photo", une instruction JSON est générée côté PHP et écrite dans un fichier commande.json.
- Ce fichier est lu en boucle par un script Python côté Raspberry, qui déclenche la prise de photo via la caméra et enregistre l'image dans /var/www/html/assets/images/.
- Une fois la photo prise, un enregistrement est effectué automatiquement dans la base de données PHOTO (titre, nom du fichier, date, heure, ID utilisateur).

Les échanges sont synchronisés et reposent sur la lecture/écriture sécurisée de fichiers partagés dans un dossier commun au serveur web et au script Python.

Cette solution respecte les exigences #PHO01 pour permettre la prise de photo via le site, ainsi que #PHO04 qui spécifie l'utilisation d'un programme Python pour déclencher la capture avant ou après un test. Le mécanisme d'écriture du fichier JSON et sa lecture par le

script Python assure une communication fiable et sécurisée entre le serveur web et le matériel, garantissant la cohérence des données enregistrées dans la base.

4. Sécurité et restrictions

- Le chemin du fichier est automatiquement généré par PHP (avec timestamp unique), évitant tout écrasement.

- La suppression est uniquement logique (mise en corbeille) pour assurer la traçabilité.

Cette approche répond à l'exigence #PHO05 concernant la gestion des photos, en assurant que chaque fichier photo a un nom unique évitant les conflits. La suppression logique respecte également les besoins de traçabilité des données, conformément à l'exigence #LOG01 qui impose la conservation des informations relatives aux actions effectuées dans l'application.

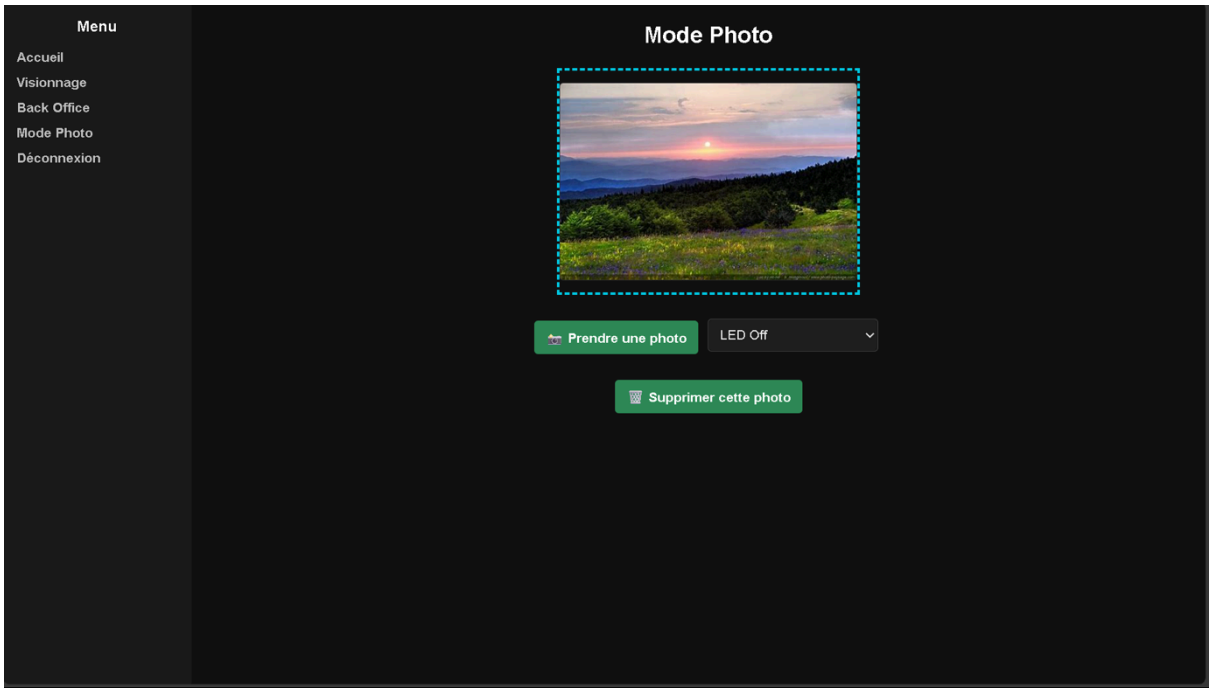


Figure 8 – Page du mode photo avec prise d'image possible

ix. Back Office – Gestion des utilisateurs, sécurité et logs

Le back office constitue la partie la plus sensible et stratégique du site PHOTO_ATB. Il est réservé exclusivement aux utilisateurs ayant le rôle d'Administrateur ou de Super Administrateur. Toute tentative d'accès non autorisé est bloquée par une vérification de session dès le chargement de la page.

Ce panneau de gestion permet de contrôler l'ensemble des utilisateurs, d'ajuster les niveaux de sécurité, de consulter l'historique complet des actions effectuées et de restaurer ou supprimer définitivement les photos envoyées à la corbeille.

Cette organisation respecte pleinement les exigences #ACC01-02 et #SADM01-03 en limitant l'accès au back-office aux seuls Administrateurs et Super Administrateurs. La vérification systématique de la session garantit la sécurité d'accès, conformément à l'exigence #ACC01-01. Par ailleurs, le contrôle des utilisateurs et la gestion des actions réalisées répondent aux besoins exprimés dans #USR01 et #LOG01 pour une administration efficace et sécurisée du système.

1. Gestion des utilisateurs

Le premier bloc permet l'ajout d'un nouvel utilisateur avec un identifiant, un mot de passe, un nom, un prénom et un rôle. Des règles strictes ont été mises en place :

- Un Administrateur peut uniquement créer des comptes Opérateur ;
- Seul un Super Administrateur peut ajouter un Administrateur ;
- Il est impossible d'ajouter un Super Administrateur ;
- En cas de tentative d'ajout d'un identifiant déjà existant, une erreur est affichée ;
- Lors de la création, le mot de passe saisi est automatiquement validé selon les paramètres définis ci-dessous, puis stocké de manière sécurisée (hashé).

Ce bloc répond aux exigences #USR01-01 et #USR01-03 en limitant la création des comptes aux types appropriés selon le rôle de l'utilisateur (Administrateur ou Super Administrateur). L'interdiction d'ajouter un Super Administrateur assure le respect de #SADM01-01. La validation stricte et le stockage sécurisé des mots de passe sont conformes aux exigences de sécurité #SEC01-07 et aux règles définies dans #SEC01-01 à #SEC01-06. La gestion des erreurs lors de doublons correspond à une bonne pratique de robustesse fonctionnelle.

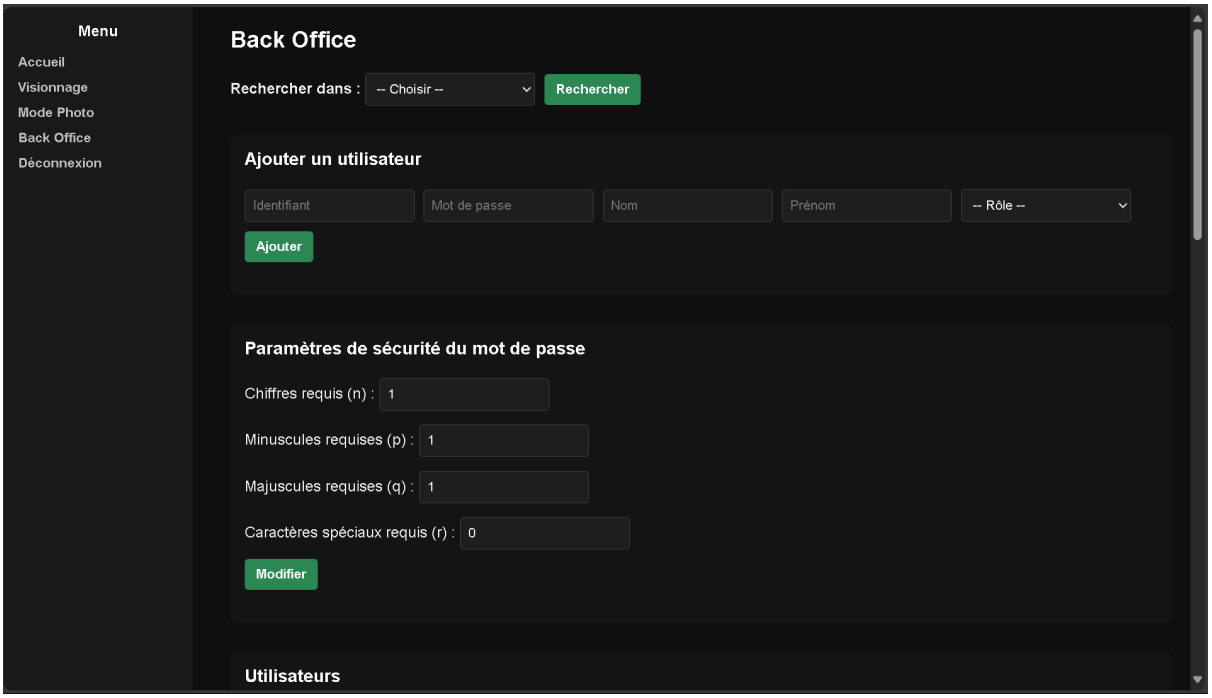


Figure 9 – ajout et règles dans back office

2. Paramètres de sécurité du mot de passe

Le second bloc permet de configurer dynamiquement les règles de sécurité liées aux mots de passe :

- n : nombre minimal de chiffres ;
- p : nombre de lettres minuscules ;
- q : nombre de lettres majuscules ;
- r : nombre de caractères spéciaux.

Ces paramètres peuvent être modifiés uniquement par un Administrateur ou un Super Administrateur. Chaque modification est automatiquement journalisée dans la base de données.

Ce bloc répond aux exigences #SEC01-01 à #SEC01-04 en permettant la configuration dynamique des critères de complexité des mots de passe. La restriction de cette configuration aux Administrateurs et Super Administrateurs respecte #USR01-05 et #SEC01-09. La journalisation automatique des modifications assure la traçabilité conformément à #LOG01-01 et #LOG01-03.

3. Liste des utilisateurs et actions possibles

Une table dynamique liste tous les utilisateurs du système avec les informations suivantes : identifiant, rôle, nombre de tentatives, état du compte (bloqué ou non). Depuis cette table, plusieurs actions sont possibles :

- Débloquer un utilisateur (après 3 échecs de connexion) ;
- Modifier le rôle ou le mot de passe (sauf pour un Super Administrateur) ;
- Supprimer un compte (sauf les Super Administrateurs qui ne peuvent être supprimés) ;

Toutes les actions sont soumises à des restrictions selon le rôle de l'utilisateur connecté, et chaque modification est enregistrée dans les logs.

Cette fonctionnalité respecte les exigences #USR01-01 à #USR01-06 en permettant la gestion complète des utilisateurs, incluant le déblocage, la modification des rôles et mots de passe, ainsi que la suppression de comptes, tout en protégeant le compte unique du Super Administrateur (#SADM01-01 et #SADM01-02). Les restrictions d'accès selon le rôle connecté garantissent la sécurité et la séparation des privilèges (#ACC01-02). Enfin, la journalisation systématique des actions répond aux exigences #LOG01-01 et #LOG01-03 pour assurer la traçabilité.

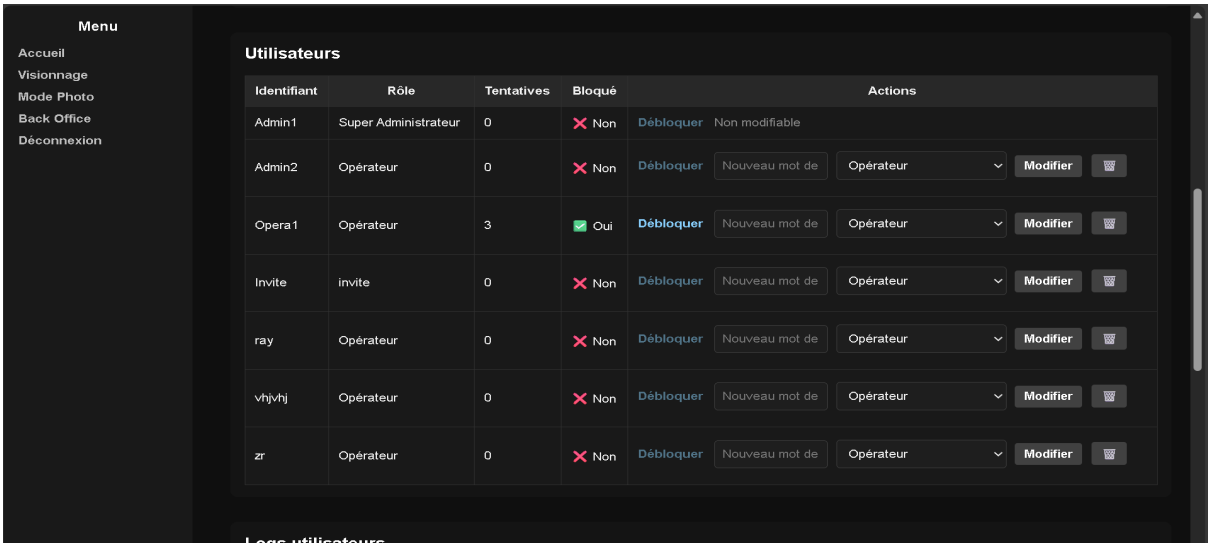


Figure 10 – Interface de modification des utilisateurs

À noter :

- Le compte Super Administrateur (ex. : "Admin1") ne peut jamais être bloqué, ni modifié, ni supprimé.

- Un compte est automatiquement bloqué au bout de 3 tentatives échouées, sauf pour le Super Administrateur.

- Chaque tentative (réussie ou échouée), ajout, modification ou suppression déclenche une ligne dans les logs utilisateurs, horodatée.

4. Suivi des actions : logs

Une table affiche l'ensemble des actions enregistrées dans le système. Chaque log contient :

- La date et l'heure ;

- L'utilisateur concerné (identifiant + prénom + nom) ;

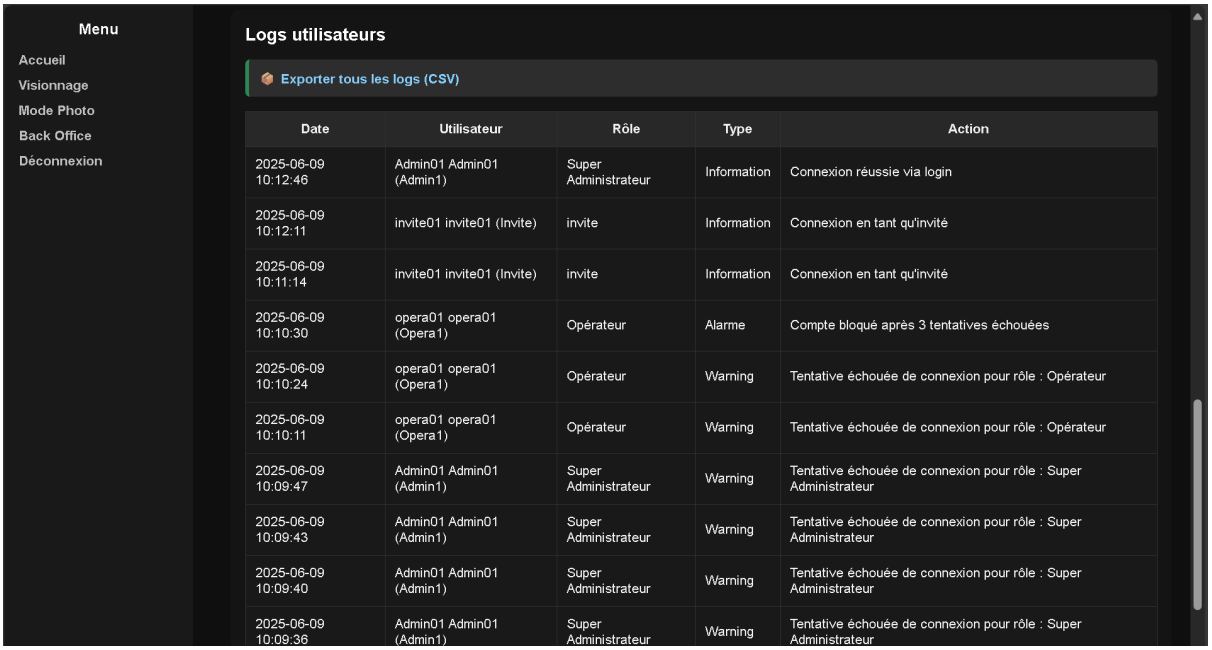
- Son rôle ;

- Le type de log (Information, Warning, ou Alarme) ;

- Une description précise de l'action effectuée (connexion, tentative échouée, modification de mot de passe, etc.).

Ce système permet un suivi complet, fiable et exportable (CSV) de l'ensemble des activités critiques liées à la sécurité et à l'usage de la plateforme.

Cette fonctionnalité répond pleinement aux exigences de traçabilité et de suivi sécuritaire (#LOG01-01 à #LOG01-04) en offrant une visibilité complète sur les actions utilisateurs, leur nature et leur contexte. La classification des logs par type (Information, Warning, Alarme) permet une gestion fine des incidents, facilitant ainsi la surveillance et l'audit. De plus, l'export au format CSV garantit la portabilité et l'analyse externe des données, renforçant la conformité aux bonnes pratiques de sécurité.



Date	Utilisateur	Rôle	Type	Action
2025-06-09 10:12:46	Admin01 Admin01 (Admin1)	Super Administrateur	Information	Connexion réussie via login
2025-06-09 10:12:11	invite01 invite01 (Invite)	invite	Information	Connexion en tant qu'invité
2025-06-09 10:11:14	invite01 invite01 (Invite)	invite	Information	Connexion en tant qu'invité
2025-06-09 10:10:30	opera01 opera01 (Opera1)	Opérateur	Alarme	Compte bloqué après 3 tentatives échouées
2025-06-09 10:10:24	opera01 opera01 (Opera1)	Opérateur	Warning	Tentative échouée de connexion pour rôle : Opérateur
2025-06-09 10:10:11	opera01 opera01 (Opera1)	Opérateur	Warning	Tentative échouée de connexion pour rôle : Opérateur
2025-06-09 10:09:47	Admin01 Admin01 (Admin1)	Super Administrateur	Warning	Tentative échouée de connexion pour rôle : Super Administrateur
2025-06-09 10:09:43	Admin01 Admin01 (Admin1)	Super Administrateur	Warning	Tentative échouée de connexion pour rôle : Super Administrateur
2025-06-09 10:09:40	Admin01 Admin01 (Admin1)	Super Administrateur	Warning	Tentative échouée de connexion pour rôle : Super Administrateur
2025-06-09 10:09:36	Admin01 Admin01 (Admin1)	Super Administrateur	Warning	Tentative échouée de connexion pour rôle : Super Administrateur

a) *Figure 11 – Tableau de logs dans le back-office*

5. Restauration et suppression définitive de photos

Enfin, le dernier bloc du back office permet de visualiser toutes les photos marquées comme "supprimées" (champ supr = 'oui' en base). Deux options sont possibles :

- Restaurer une photo : elle redevient visible dans l’accueil et le visionnage.

- Supprimer définitivement : le fichier est effacé du serveur et la base de données est mise à jour.

Ces actions sont également journalisées dans les logs avec les détails de la photo (ID, chemin, utilisateur).

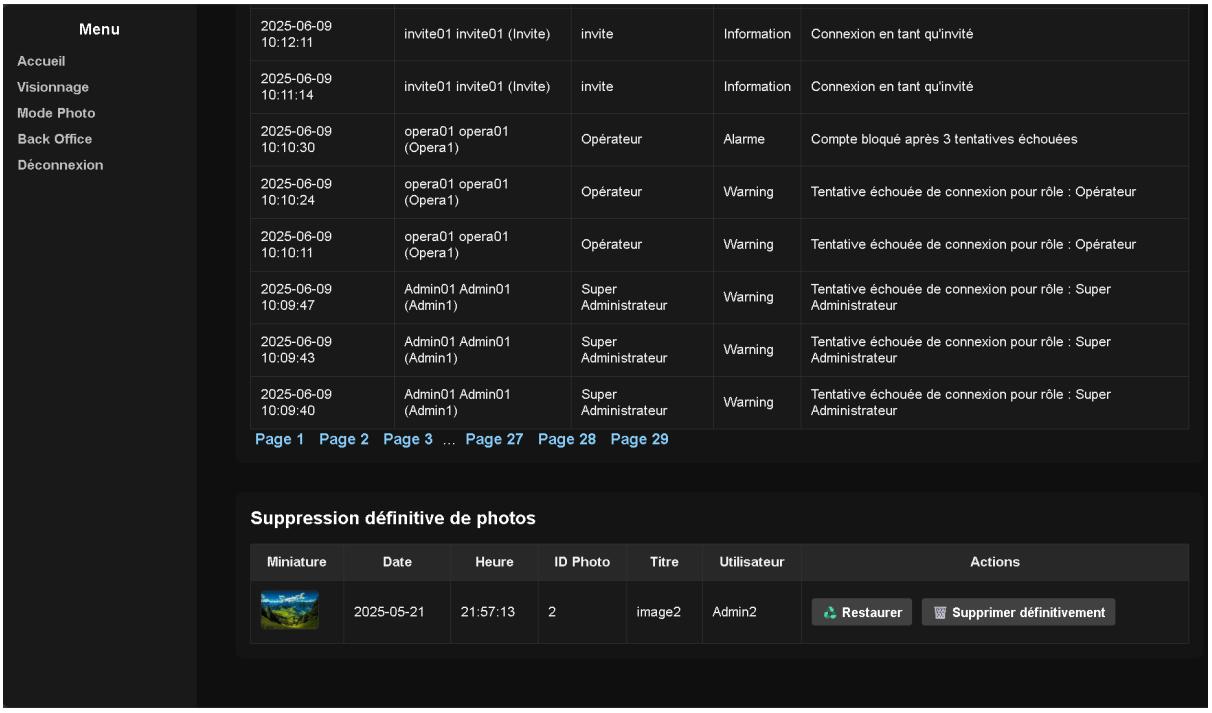


Figure 12 – Tableau de suppression définitive des photos

x. Échanges entre les pages et sécurisation du site PHOTO_ATB

L'architecture du site PHOTO_ATB repose sur une navigation multi-pages PHP, avec une logique serveur bien structurée, contrôlée via les sessions, les rôles utilisateurs et les vérifications côté serveur.

1. Navigation et transitions entre pages

Le menu latéral permet un accès direct aux principales sections du site :

- login.php (page d'authentification)
- home.php (accueil avec les dernières photos)
- visionnage.php (visionnage complet des photos)
- voir_photo.php (affichage d'une photo en plein écran)
- mode-photo.php (prise de photo via le PICO et le Raspberry)
- back-office.php (espace de gestion avancée)
- logout.php (déconnexion)

Chaque page vérifie au préalable si une session utilisateur est active ($_SESSION['user_id']) et si le rôle est compatible avec l'accès demandé. Sinon, une redirection automatique vers login.php est effectuée. Cette gestion évite les accès non autorisés par simple URL.

2. Sécurisation des accès

Pour éviter toute faille de sécurité basique, plusieurs mécanismes ont été mis en place :

- Vérification stricte du rôle utilisateur sur les pages critiques (back office, gestion utilisateurs, etc.).

- Protection contre les comptes bloqués : un utilisateur ayant dépassé le nombre de tentatives autorisées ne peut plus se connecter (hors Super Administrateur).

- Aucun Super Administrateur ne peut être supprimé ou modifié.

- Les mots de passe sont hachés avec l'algorithme password_hash avant enregistrement.

- Un système de journalisation (logs) enregistre toutes les connexions, tentatives échouées, suppressions, modifications, etc.

IV. Description de l'installation + configuration du Raspberry PI 3 et PICO WH

Le processus complet de mise en place du système embarqué basé sur un Raspberry Pi 3 et un Raspberry Pi Pico WH. Il permet de réinstaller l'environnement de travail sur une nouvelle carte SD et un nouveau Pico de manière autonome.

i. Matériel et logiciels nécessaires

Matériel requis :

- Raspberry Pi 3 avec alimentation
- Carte microSD (≥ 8 Go, classe 10 recommandée)
- Raspberry Pi Pico WH
- Écran HDMI + câble
- Clavier et souris (USB)
- Câble micro-USB pour connecter le Pico WH
- Connexion Internet (via Ethernet ou Wi-Fi)

Logiciels à télécharger :

Logiciel	**Lien de téléchargement**
Raspberry Pi Imager	https://www.raspberrypi.com/software
Raspberry Pi OS (avec Desktop)	Disponible directement dans l'Imager
MicroPython (UF2 pour Pico WH)	https://micropython.org/download/rp2-pico-w/
Thonny IDE (éditeur Python)	https://thonny.org

ii. Installation de Raspberry Pi OS sur la carte SD

- Télécharger et installer Raspberry Pi Imager depuis raspberrypi.com/software.

- Insérer la carte SD dans l'ordinateur.

1. Lancer Raspberry Pi Imager :

- Choose OS → Raspberry Pi OS (32-bit) (avec interface graphique)
- Choose Storage → sélectionner la carte SD
- Cliquer sur Write et patienter jusqu'à la fin du flashage.

2. Une fois terminé, retirer la carte SD en toute sécurité.

iii. Premier démarrage du Raspberry Pi 3

1. Insérer la carte SD dans le Raspberry Pi 3.
2. Connecter l'écran, le clavier, la souris, puis l'alimentation.
3. Le système démarre automatiquement et affiche un assistant de configuration initiale :

- Choix de la langue, pays et fuseau horaire
- Définition d'un nouveau mot de passe pour l'utilisateur pi
- Connexion à un réseau Wi-Fi (ou Ethernet déjà actif)
- Proposition de mettre à jour le système
- Réglage de l'affichage (overscan si besoin)

4. Cliquer sur Finish, puis redémarrer si demander.

iv. Configuration logicielle du Raspberry Pi 3

1. Ouvrir le **Terminal** depuis le menu principal.
2. Mettre à jour le système :

```
#sudo apt update
#sudo apt full-upgrade -y
```

3. Installer les outils nécessaires au projet :

   #sudo apt install python3 python3-pip -y

   #pip3 install flask

   #sudo apt install python3-serial python3-picamera -y

4. Créer un dossier de projet :

   #mkdir ~/Site_Thales

V. Configuration d'un serveur local (Apache2 + PHP) sur un Raspberry Pi 3 + Déploiement d'un site web

1. Prerequis :

   - Raspberry Pi 3 avec Raspberry Pi OS

   - Connexion Internet

   - Terminal avec droits sudo

   - Ton projet web prêt dans un dossier

   2. Installation d'Apache2 et PHP :

   sudo apt update

   sudo apt install apache2 php libapache2-mod-php

   3. Nettoyage du contenu par defaut d'Apache :

   sudo rm -r /var/www/html/*

5. Copier ton site web dans le dossier du serveur :

   sudo cp -r ~/Bureau/mon_site/* /var/www/html/

6. Donner les droits d'acces au serveur web :

   sudo chown -R www-data:www-data /var/www/html/

```
sudo chmod -R 755 /var/www/html/
sudo chmod -R 775 /var/www/html/assets/images
sudo chmod 664 /var/www/html/mon_site/BdD_SAE15_V2.db
```

7. Redemarrer le serveur Apache apres modifications : sudo systemctl restart apache2

8. Verification finale : Ouvre http://localhost/ dans ton navigateur. votre site devrait s'afficher correctement.

VI. Configuration du Raspberry Pi Pico

1. Brancher le Pico à un ordinateur tout en maintenant le bouton **BOOTSEL** jusqu'à ce qu'un disque nommé RPI-RP2 apparaisse.
2. Télécharger le fichier .uf2 de MicroPython depuis : https://micropython.org/download/rp2-pico-w/
3. Glisser-déposer le fichier .uf2 dans le disque RPI-RP2. Le Pico redémarre automatiquement.
4. Ouvrir **Thonny IDE**, aller dans :

- Outils > Options > Interpréteur
- Sélectionner : MicroPython (Raspberry Pi Pico)
- Choisir le bon port série

i. Test du Raspberry Pi Pico avec un script simple

Pour vérifier le bon fonctionnement du Raspberry Pi Pico, nous avons utilisé un script Python très simple exécuté depuis l'environnement Thonny. Ce code permet de faire clignoter la LED intégrée toutes les 0.5 secondes, ce qui confirme que la carte est bien détectée et opérationnelle.
Voici le script utilisé :

```
import machine

import time
```

```
led = machine.Pin(25, machine.Pin.OUT)
while True:
 led.toggle()
time.sleep(0.5)
```

Ce test de base est une première étape importante avant d'implémenter des programmes plus complexes. Il valide la communication entre l'ordinateur et la carte, ainsi que la configuration correcte du port USB et de l'environnement de développement.

Grâce à ce guide, nous sommes capables de réinstaller complètement le système sur une nouvelle carte SD ainsi que de reprogrammer un nouveau Raspberry Pi Pico WH.

VII. Configuration du Rasberry sur une adresse IP particulière pour un sous-réseau spécifique

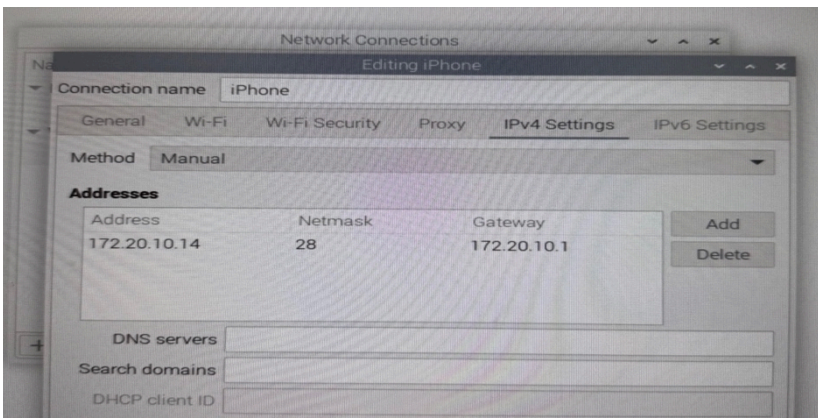
Nous avons configuré manuellement le Raspberry Pi afin de lui attribuer une adresse IP fixe correspondant à un sous-réseau spécifique. Cette configuration a été réalisée directement depuis l'interface graphique du système.

Voici les étapes que nous avons suivies :

1. Nous avons accédé aux paramètres réseau en cliquant sur l'icône Wi-Fi située en haut à droite du bureau, puis en sélectionnant "Edit Connections…".
2. Nous avons ensuite choisi la connexion Wi-Fi active (nommée *iPhone* dans notre cas) et cliqué sur "Edit…".
3. Dans l'onglet "IPv4 Settings", nous avons sélectionné la méthode "Manual", puis renseigné les informations suivantes :

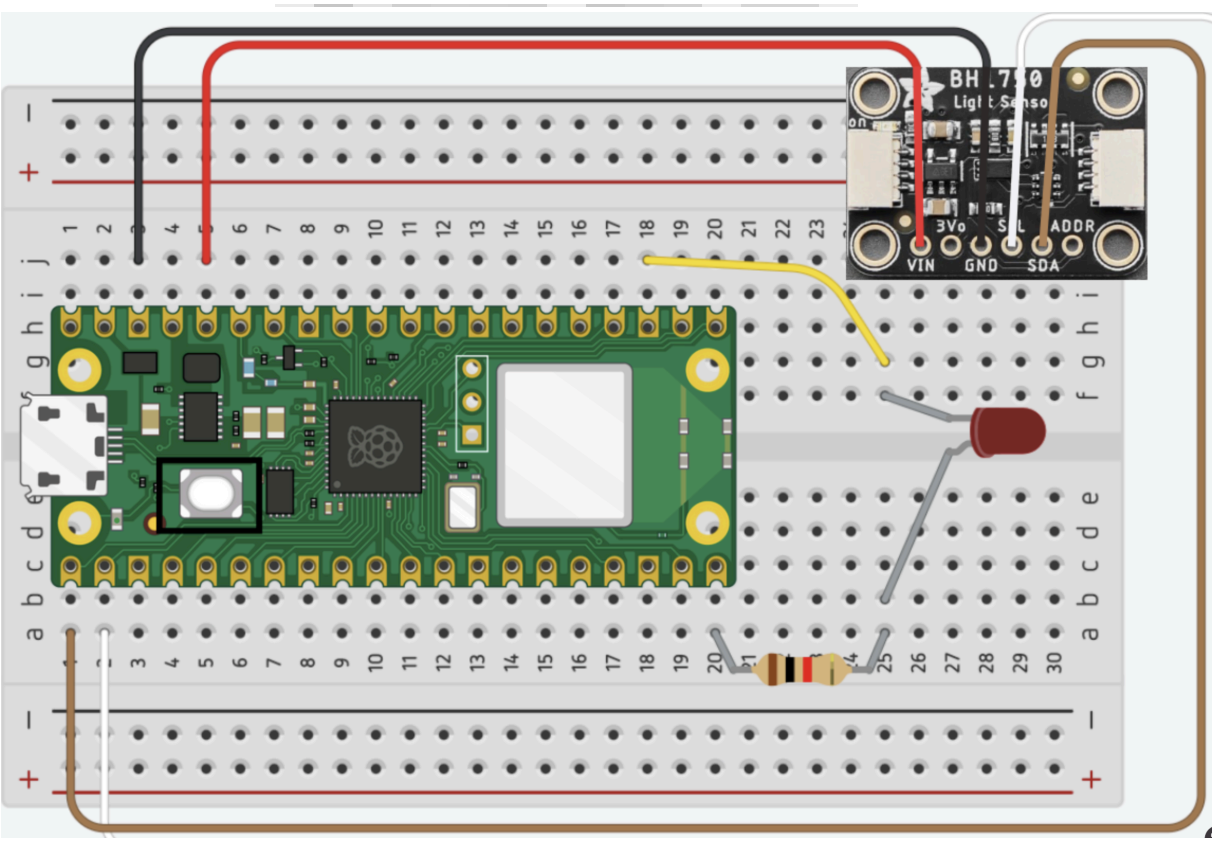
- Adresse IP : 172.20.10.14
- Masque de sous-réseau : /28
- Passerelle (Gateway) : 172.20.10.1

4. Après avoir validé les modifications en cliquant sur "Save", nous avons redémarré la connexion pour appliquer les nouveaux paramètres.



Cette configuration permet au Raspberry Pi de disposer d'une adresse réseau stable, facilitant la communication avec les autres équipements du réseau.

VIII. Schéma électrique du montage



Notre montage est composé :

- D'un Raspberry Pi PICO W
- D'une LED rouge à deux branches

- D’une résistance 180 Ohms

- D’un capteur de lumière ambiante Adafruit BH1750

- De 5 fils de liaison de couleur diOérente (jaune, rouge, noir, blanc et marron)

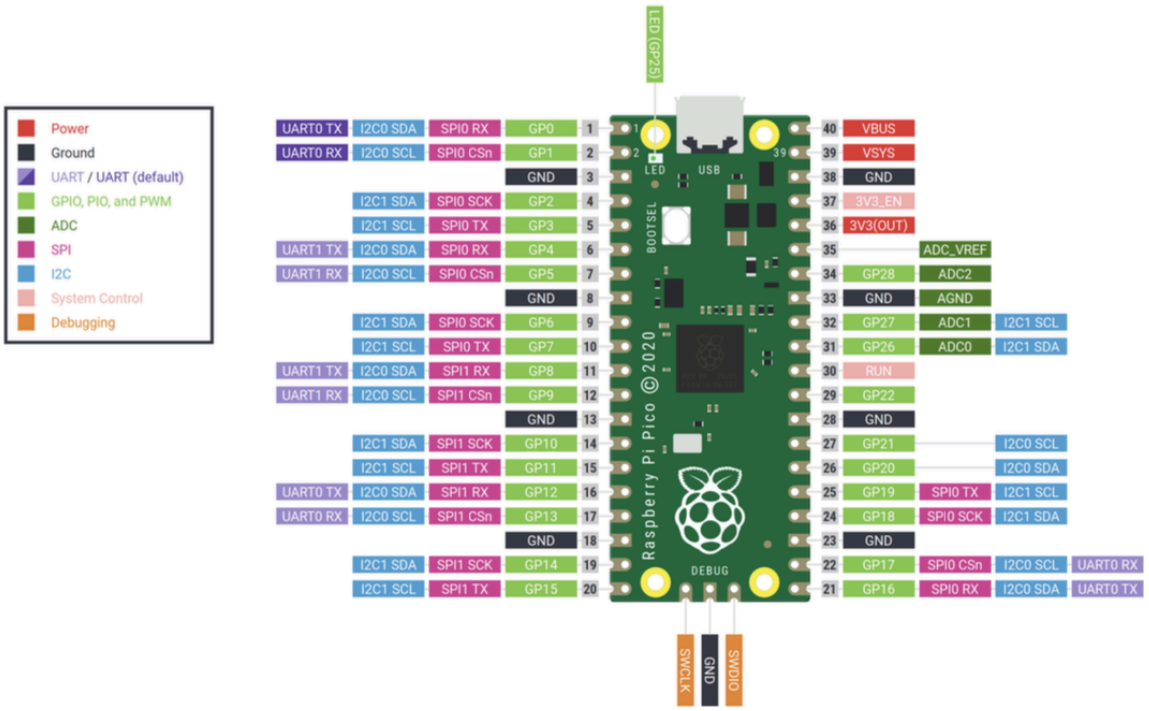
Liaisons de la LED :

La LED est connectée du côté positif par la résistance au PIN GP15 de la PICO (ligne 20 de la breadboard) qui est alimenté (ou non) en fonction des conditions d’utilisation du système. Elle est ensuite reliée côté - au point de terre par un fil jaune.

Liaison du capteur de luminosité :

L’Adafruit BH1750 possède 4 liaisons distinctes à la PICO. La première, avec le fil rouge, permet l’alimentation du module. Elle relie le PIN VIN (d’alimentation) du capteur au PIN 3V3(OUT) de la PICO qui fournit une tension de 3,3V. La seconde liaison est celle vers le point de terre (GND). Le fil noir connecte ce PIN à un PIN GND de la PICO. La troisième liaison correspond à la clock I2C. Le fil blanc permet de connecter le PIN clock I2C des deux équipements (capteur et PICO) afin de renvoyer la clock du capteur. Enfin le dernier PIN du capteur relié est le I2C SDA, soit le Pin renvoyant les données du capteur. Ce PIN est relié au PIN de même type de la PICO.

Bien qu’ayant décidé de fournir un montage électrique tel, il existe de nombreuses autres possibilités de branchement. La plupart des types de PIN dont nous avions besoin (PIN GND, I2C SDA, I2C SCL, GP) de relier à la PICO sont présent de multiple fois sur celle-ci comme nous pouvons le voir ci-dessous :



Source de la photo :

https://fr.vittascience.com/learn/tutorial.php?id=724/manipulation-des-gpio-avec-la-Raspberry-pi-pico

Pour ce travail nous nous sommes aidés de :

- https://github.com/flrrth/pico-bh1750/tree/main/bh1750

- https://www.google.com/url?sa=t&source=web&rct=j&opi=89978449&url=https://www.youtube.com/watch%3Fv%3Dl9MAZu7yvN4&ved=2ahUKEwic-c2P8-mNAxVmZqQEHUhEIT8QtwJ6BAgMEAI&usg=AOvVaw2DQp8R-HY6vjScpVP-qsg0

- https://randomnerdtutorials.com/raspberry-pi-pico-bh1750-micropython/

- https://randomnerdtutorials.com/raspberry-pi-pico-w-pinout-gpios/#power-pins

- https://learn.adafruit.com/adafruit-bh1750-ambient-light-sensor/pinouts

IX. Schéma de la base de données

Pour mieux concevoir et créer la base de données nécessaire pour le projet nous avons mis en place un schéma entités-association et un schéma logique :

Schéma entités-associations :

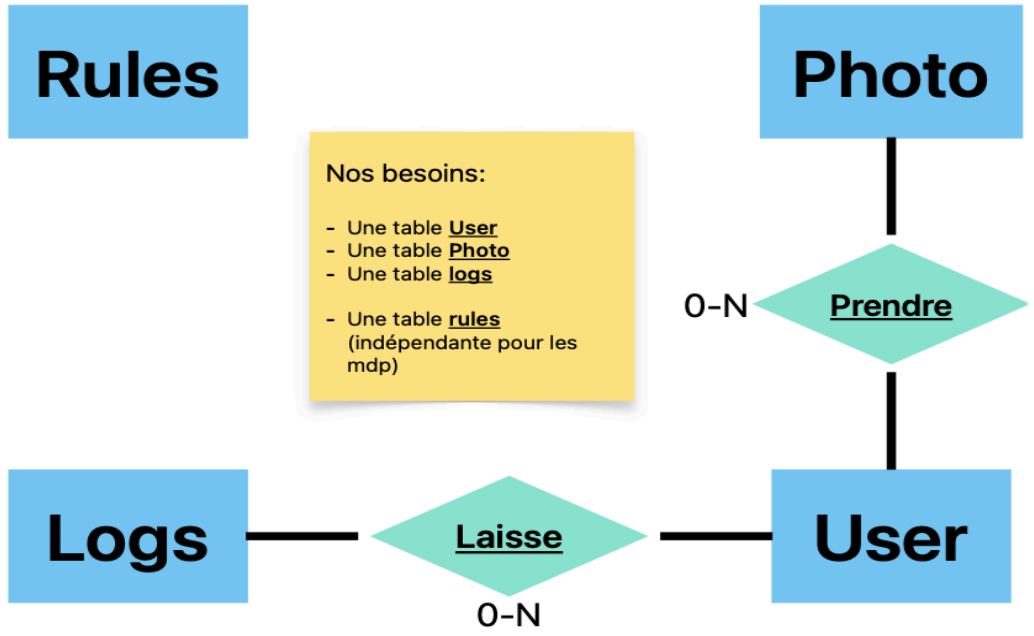
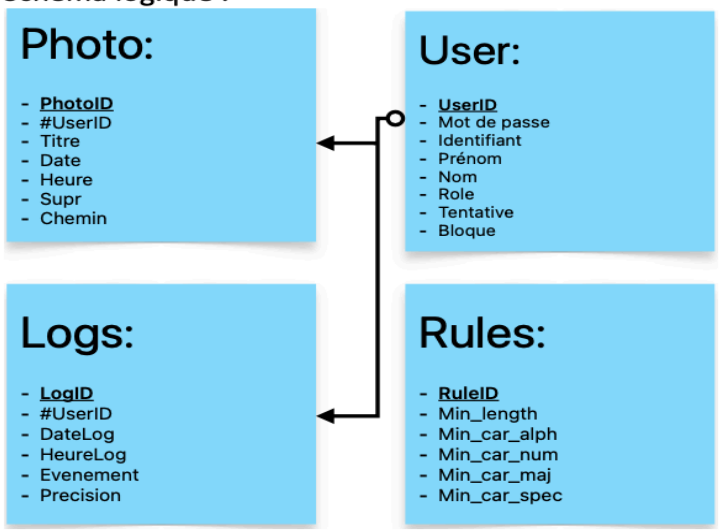


Schéma logique :



Nous obtenons ensuite une base de données normalisé 1NF, 2NF et 3NF qui possède en tout 4 tables :

La table User permet de définir chaque utilisateur au sein du site.
Elle se compose :

- D'un userID unique pour tous les enregistrements et servant à identifier n'importe quel compte utilisateur. Il est présent sous la forme d'un entier et nous sert de clé primaire de la table.
- D'un mdp qui correspond tout simplement au mot de passe haché de l'utilisateur sous la forme d'une chaine de caractère.
- D'un acID qui est l'identifiant de l'utilisateur utilisé pour se connecter au site comme chaine de caractère.
- D'un prénom et d'un nom qui sont l'identité publique de l'utilisateur, ce qui est affiché sur le site.
- D'un rôle qui donne accès ou non à toutes les fonctionnalités/droits sur le site. Soit opérateur, soit administrateur ou bien super administrateur.
    - Le super administrateur n'a aucune limita7on mais il n'existe qu'un seul compte ayant ce rôle.
    - Les administrateurs, eux, ont accès à toutes les fonctionnalités du site mais n'ont pas de pouvoir sur le super admin, ils servent majoritairement à contrôler les différents utilisateurs opérateur.
    - Les opérateurs n'ont aucun accès aux back office, et n'exerce donc aucun pouvoir sur les autres utilisateurs. Ils ne peuvent pas consulter les logs, supprimer définitivement des photos ou créer de nouveaux comptes utilisateur. Concrètement e rôle permet simplement de prendre des photos, les supprimer partiellement et les visionner au sein du site web.
- D'un nombre de tentatives qui correspond au nombre d'échecs d'authentification du compte sous forme d'entier.

La table Photo permet de définir chaque photo enregistrée de manière unique.

Cette table se compose :

- D'un photoID unique qui, tout comme le userID, sert à identifier n'importe quelle photo et a le rôle de clé primaire.
- Du userID de l'auteur de la photo, faisant référence à la table User comme clé étrangère de cette table.
- D'un titre renvoyant au titre de la photo sous la forme d'une chaine de caractère.
- D'une date et d'une heure, la date et l'heure de la capture de la photo (sous forme DATE et TIME).
- D'un champ supr prenant comme valeur « oui » ou « non » correspondant à la suppression partielle de la photo. Si la valeur est « oui » la photo n'est plus affichée sur le site mais un administrateur peut la restaurer. La suppression totale s'effectuer lorsqu'un administrateur décide d'effacer une photo supprimée partiellement en allant

dans les back office. Dans ce cas la photo est enlevée complètement de la Base de données.

- Du chemin vers la photo, du dossier jusqu'au fichier, permettant ensuite d'y accéder plus facilement en faisant une simple requête à la base. Ce chemin est enregistré sous forme de chaine de caractère.

La table Logs permet d'enregistrer de façon unique chaque action effectuée sur le site/ logs pour permettre à l'administrateur de vérifier l'activité sur le site.

Cette table se compose :

D'un logID, clé primaire de cette table, pour identifier plus facilement chaque enregistrement de logs sous forme d'entier.
Du userID de l'utilisateur ayant effectué l'action, renvoyant au userID de la table
User et qui nous sert ici de clé étrangère.
D'une date et d'une heure correspondant au moment précis de l'action, avec jour mois année, heure minute... (sous forme DATE et TIME).
D'un champ évènement qui permet de caractériser l'action ayant été effectué sur le site. Elle prend comme valeur une chaine de caractère, par exemple « suppression d'une photo ».
D'un champ plus qui permet d'ajouter au log plus de détails, toujours sous forme de chaine de caractères.

Enfin la table Rule permet de définir les règles de création d'un mot de passe.
Pour cela elle utilise les champs :

- ruleID qui permet d'identifier de manière unique chaque règle créée), sous forme d'entier.
- n qui définit le nombre minimal de caractère numérique (0-9), aussi sous forme d'entier par défaut mis à 1 comme tous les champs suivants.
- p qui définit le nombre minimal de caractère en minuscule (A-Z).
- q qui définit le nombre minimal de caractère en majuscule (A-Z).
- r qui définit le nombre minimal de caractère spéciaux.

X. Plan de validation, Procédures de tests, rapports de tests et fiches d'anomalies

1. Plan de Validation

Dans le cadre de notre projet, un ensemble de tests fonctionnels a été défini et exécuté afin de valider les principales fonctionnalités du système. Chaque test correspond à un scénario réel d'utilisation et s'appuie sur des exigences précises répertoriées (ex. : #ACC01, #SEC01, #PHO01, etc.). L'objectif de cette phase était de garantir que le système répond bien aux attentes techniques et fonctionnelles, tout en assurant la sécurité et la stabilité des modules critiques.

Trois grandes séries de tests ont été réalisées. Le premier test (TEST-R02) concernait la vérification du système d'authentification et de gestion des utilisateurs dans la version v2.0. Il a permis de confirmer que la connexion est bien sécurisée, que les mots de passe sont chiffrés, et que l'administration des utilisateurs est fonctionnelle. Le second test (TEST-R04), mené sur la version v3.0, portait sur la gestion avancée des comptes utilisateurs, la personnalisation des politiques de mot de passe et les droits selon les profils. Tous les critères de sécurité ont été respectés. Enfin, le test TEST-R03 s'est concentré sur le module de capture photo automatique via Raspberry Pi. Il a validé les prises de vue depuis l'interface web, le déclenchement automatique après inactivité, ainsi que l'enregistrement et la suppression des photos.

Les résultats de ces tests ont montré que l'ensemble des fonctionnalités critiques sont opérationnelles et conformes aux exigences du cahier des charges. Aucune anomalie bloquante n'a été détectée.

Voir Annexe 1

2. Synthèse des résultats des tests fonctionnels

Les différents rapports de test réalisés ont permis de valider les principales fonctionnalités du système. Ils couvrent notamment l'authentification sécurisée des utilisateurs, la gestion des rôles et des droits d'accès, ainsi que le système de capture photo automatique et manuelle. Chaque test a été mené selon un protocole strict, incluant la préparation de l'environnement, l'exécution des scénarios et la vérification des résultats par rapport aux critères définis. Les résultats obtenus confirment la conformité du système aux spécifications fonctionnelles et de sécurité. L'ensemble des fonctionnalités critiques a été testé avec succès, garantissant un fonctionnement fiable et sécurisé des modules principaux.

Voir Annexe 2

3. Description des procédures de test et résultat

Les tests fonctionnels ont été menés selon une procédure rigoureuse pour s'assurer du bon fonctionnement et de la robustesse du système. Tout d'abord, l'accès au site a été effectué via l'adresse HTTP classique ainsi que directement par l'adresse IP du serveur, avec une page de login s'affichant correctement. La connexion a ensuite été testée avec un compte valide, permettant l'accès au tableau de bord correspondant au profil utilisateur. Un administrateur a procédé à l'ajout d'un nouvel utilisateur, dont la présence a été confirmée dans la base de données. Par ailleurs, la gestion des erreurs a été évaluée en provoquant trois tentatives de connexion infructueuses successives, ce qui a entraîné le blocage automatique du compte et l'affichage d'un message d'erreur explicite.

Les actions réalisées ont également été suivies via les logs, qui enregistrent précisément chaque événement avec l'identifiant de l'utilisateur concerné et le type d'action effectuée. En complément, le système de détection de luminosité a été testé en couvrant le capteur, ce qui a déclenché l'allumage automatique d'une LED. Enfin, la fonctionnalité de capture photo a été validée en lançant manuellement une prise de photo depuis l'interface web : l'image a été capturée, nommée selon la convention définie, puis stockée correctement sur le serveur. Cette série de tests garantit ainsi la conformité des fonctionnalités essentielles du système, tant au niveau opérationnel que sécuritaire.

4. Analyse et suivi des anomalies identifiées

Malgré ces résultats positifs, plusieurs anomalies ont été détectées et documentées de manière détaillée dans des fiches spécifiques. Ces anomalies concernent principalement des problèmes liés à la gestion des photos marquées comme supprimées, des conflits potentiels dans la génération des noms de fichiers lors de prises simultanées, ainsi que des failles mineures dans l'export des logs. Des solutions ont été proposées ou mises en œuvre lorsque possible, tandis que d'autres restent à traiter pour améliorer la robustesse globale du système. Ces fiches permettent un suivi précis et une traçabilité des dysfonctionnements identifiés.

Pour plus de détails, les fiches de tests et d'anomalies complètes sont disponibles en annexe de ce rapport.

Voir Annexes 3

XI. Gestion de Projet

1. Gant

Le diagramme de Gantt élaboré pour le projet SAE 15 s'est révélé être un outil central pour structurer la gestion du projet sur l'ensemble du semestre, de février à juin 2025. Il a permis une planification rigoureuse des différentes phases, notamment la planification des tâches, la répartition des rôles, l'évaluation des risques, la rédaction du rapport et la préparation de la présentation finale. Les tâches principales ont été clairement identifiées : développement du site web (PHP, HTML, CSS), configuration du Raspberry Pi et du microcontrôleur Pico WH, développement Python, mise en place de la base de données, création du schéma électrique, ainsi que la définition du plan de validation, des tests et des fiches d'anomalies.

Chaque activité était associée à un ou plusieurs membres de l'équipe selon un tableau RACI implicite, garantissant ainsi la bonne répartition des responsabilités. Par exemple, la configuration du Raspberry et du Pico a été réalisée principalement en mai par Mohammed, Adrien et Rayan, tandis que le développement du site web s'est étalé d'avril à mi-mai avec une implication collective. La base de données, initialement prévue sur trois semaines, s'est en réalité étalée de mi-février à début avril, illustrant une adaptation réaliste à la complexité des tâches. De plus, la partie tests et validation a été menée progressivement tout au long du projet, en lien avec l'évolution des modules développés.

Ce planning visuel a facilité la coordination de l'équipe, permis un suivi précis des avancements à l'aide d'indicateurs (barres de progression, durées effectives) et a favorisé les ajustements en cas de décalage ou de surcharge. Le diagramme de Gantt a ainsi joué un rôle essentiel dans la structuration du travail collaboratif, en assurant à la fois le respect des délais, la fluidité de la communication et la clarté des objectifs partagés.

Planification

Titre du Projet	SAE 15
Chef de Projet	Adrien
Membres du Groupe	Adrien, Arthur, Mohammed, Rayan, Sarah
Date Démarrage Projet (Planning)	20/02/2025

1.1	P	Planification des tâches	20/02/2025	25/02/2025	5	100%	Terminée
1.1	R	Planification des tâches	20/02/2025	25/02/2025	5	100%	Terminée
1.2	P	Répartition des tâches	25/02/2025	28/02/2025	3	100%	Terminée
1.2	R	Répartition des tâches	25/02/2025	28/02/2025	3	100%	Terminée
1.3	P	RACI	01/03/2025	05/03/2025	4	100%	Terminée
1.3	R	RACI	01/03/2025	05/03/2025	4	100%	Terminée
1.4	P	Analyse des risques	06/03/2025	15/03/2025	9	100%	Terminée
1.4	R	Analyse des risques	06/03/2025	15/03/2025	9	100%	Terminée
2.0	P, R	Développement technique	20/02/2025	31/05/2025	100	100%	
2.1	P	Programme Python	25/02/2025	15/03/2025	21	100%	Terminée
2.1	R	Programme Python	25/04/2025	15/05/2025	20	100%	Terminée
2.2	P	Base de Données	01/03/2025	22/03/2025	21	100%	Terminée
2.2	R	Base de Données	24/01/2025	25/04/2025	49	100%	Terminée
2.3	P	Création site web (PHP, HTML, CSS)	25/03/2025	15/05/2025	51	100%	Terminée
2.3	R	Création site web (PHP, HTML, CSS)	20/03/2025	22/05/2025	40	100%	Terminée
2.4	P	Configuration Raspberry	20/02/2025	05/03/2025	10	100%	Terminée
2.4	R	Configuration Raspberry	01/05/2025	05/06/2025	30	100%	Terminée
2.5	P	Configuration Pico	20/02/2025	20/05/2025	89	100%	Terminée
2.5	R	Configuration Pico	01/05/2025	05/06/2025	30	100%	Terminée
2.6	P	Schéma électrique	01/05/2025	31/05/2025	30	100%	Terminée
2.6	R	Schéma électrique	16/05/2025	05/06/2025	30	100%	Terminée
3.0	P, R	Validation et Tests	01/03/2025	15/06/2025	106	100%	
3.1	P	Plan de validation	01/03/2025	30/04/2025	60	100%	Terminée
3.1	R	Plan de validation	27/03/2025	05/06/2025	60	100%	Terminée
3.2	P	Tests et fiche anomalie	15/04/2025	15/06/2025	61	100%	Terminée
3.2	R	Tests et fiche anomalie	07/04/2025	05/06/2025	61	100%	Terminée
4.0	P, R	Présentation du projet	01/06/2025	19/06/2025	18	70%	
4.1	P	Bilan d'avancement	01/06/2025	10/06/2025	9	100%	Terminée
4.1	R	Bilan d'avancement	01/06/2025	10/06/2025	9	100%	Terminée
4.2	P	Préparation du rapport	05/06/2025	15/06/2025	10	80%	En cours
4.2	R	Préparation du rapport	05/06/2025	15/06/2025	10	80%	En cours
4.3	P	Préparation présentation orale	10/06/2025	18/06/2025	8	50%	En cours
4.3	R	Préparation présentation orale	10/06/2025	18/06/2025	8	50%	En cours
4.4	P	Réalisation du diaporama	12/06/2025	19/06/2025	7	40%	En cours
4.4	R	Réalisation du diaporama	12/06/2025	19/06/2025	7	40%	En cours

2. Risques

Registre des Risques					
Projet :	Photo_atb	Groupe :	Thalès14	Date de Mise à jour:	23/10/24
Id :	1	Créateur:	Moahdrimeden	Date création:	23/10/24
Titre :	Mauvaise entente dans le groupe				
Cause :	une problème au sein de l'équipe / tensions liées au projet				
Conséquence :	Retard / tensions au sein de l'équipe / Non communication de l'équipe				
Sévérité (1)	Probabilité (1)	Rouge (2)	Jaune (2)	Vert (2)	Type (3)
3	2		X		Planning
Décisions sur le risque & action(s) à mettre en œuvre					
Risque accepté (4):	NON		Risque refusé (4) :	OUI	
Actions pour prévenir le risque :	Bonne entente dans le groupe / aider les autres				
	Réduction attendue:	Faire un groupe / pouvoir communiquer avec tous			
	Statut action(s):	Réaction nécessaire / sur la durée			
Actions à mettre en place si risque rencontré :	Communication avec le ou les membres dans l'IUT / Essayer de trouver un accord commun				

(1) valeur entre 1 et 5
(2) mettre une croix dans la case correspondante
(3) Technique ou Planning
(4) oui ou non

Registre des Risques					
Projet :	Photo_atb	Groupe :	Thalès14	Date de Mise à jour:	23/10/24
Id :	1	Créateur:	Moahdrimeden	Date création:	23/10/24
Titre :	Dysfonction de la caméra				
Cause :	Casse de l'appareil / mauvais entretien				
Conséquence :	Retard / cout / retard pour tests				
Sévérité (1)	Probabilité (1)	Rouge (2)	Jaune (2)	Vert (2)	Type (3)
2	1			X	Technique
Décisions sur le risque & action(s) à mettre en œuvre					
Risque accepté (4):	OUI		Risque refusé (4) :	NON	
Actions pour prévenir le risque :	Bien s'occuper des appareils / les entretenir				
	Réduction attendue:	Communication au encadrants			
	Statut action(s):	Réaction nécessaire / plus tôt résolu = mieux			
Actions à mettre en place si risque rencontré :	Achat d'une nouvelle caméra ou remplacement avec les professeurs responsables				

(1) valeur entre 1 et 5
(2) mettre une croix dans la case correspondante
(3) Technique ou Planning
(4) oui ou non

Registre des Risques					
Projet :	Photo_atb	Groupe :	Thales 14		
Id :		Créateur:	Groupe		
Titre :	Retard dans la livraison des livrables				
Cause :	Sous-estimation du temps, surcharge de travail, mauvaise répartition				
Conséquence :	Rendu incomplet ou en retard, perte de points à l'évaluation				
Sévérité (1)	Probabilité (1)	Rouge (2)	Jaune (2)	Vert (2)	Type (3)
			X		
Décisions sur le risque & action(s) à mettre en œuvre					
Risque accepté (4):			Risque refusé (4) :		
Actions pour prévenir le risque :	Planification réaliste, suivi hebdomadaire, mise en place d'un Gantt et de réunions régulières				
Actions à mettre en place si risque rencontré :	Réorganisation des tâches, mise en place d'un sprint intensif, réduction de périmètre si nécessaire				

Registre des Risques					
Projet :	Photo_atb	Groupe :	Thales 14		
Id :		Créateur:	Groupe		
Titre :	Panne du Raspberry ou du Pico				
Cause :	Défaut matériel, mauvaise manipulation				
Conséquence :	Blocage des tests, impossibilité de valider certaines fonctionnalités				
Sévérité (1)	Probabilité (1)	Rouge (2)	Jaune (2)	Vert (2)	Type (3)
			X		
Décisions sur le risque & action(s) à mettre en œuvre					
Risque accepté (4):			Risque refusé (4) :		
Actions pour prévenir le risque :	Avoir un matériel de secours, manipuler avec précaution, tests progressifs				
Actions à mettre en place si risque rencontré :	Remplacement immédiat du matériel ou basculement sur un émulateur temporaire				

Registre des Risques					
Projet :	Photo_atb	Groupe :	Thales 14		
Id :		Créateur:	Groupe		
Titre :	Problème d'accès à l'interface web				
Cause :	Bugs dans le code, mauvais lien IP ou DNS				
Conséquence :	Impossibilité de tester ou de démontrer certaines fonctionnalités				
Sévérité (1)	Probabilité (1)	Rouge (2)	Jaune (2)	Vert (2)	Type (3)
			X		
Décisions sur le risque & action(s) à mettre en œuvre					
Risque accepté (4):			Risque refusé (4) :		
Actions pour prévenir le risque :	Vérification régulière de l'hébergement local, IP statique si possible, test multi-navigateurs				
Actions à mettre en place si risque rencontré :	Correction du bug en urgence, basculement temporaire sur localhost				

3. RACI

Le tableau RACI que nous avons utilisé pour le projet SAE a été très utile pour organiser notre travail en équipe. En définissant pour chaque tâche qui était responsable, qui devait valider, qui devait être consulté et qui devait être informé, on a pu clarifier les rôles de chacun dès le départ. Les tâches importantes, comme le développement du site web, la programmation Python ou la configuration du matériel (Pico WH, Raspberry Pi), ont été confiées à un référent, ce qui a rendu le suivi des avancées beaucoup plus simple. Grâce à ça, on a évité les doublons, mieux communiqué et assuré que tout le monde était impliqué au bon moment. Le RACI nous a vraiment aidés à structurer le projet et à travailler ensemble de façon efficace jusqu'au bout.

Tâches	Adrien	Arthur	Mohammed	Rayan	Sarah
Gestion de Projet	I	A	I	I	R
Création du site Web	I	C	I	R	C
Configuration Raspberry Pi	A	I	R	C	I
Configuration Pico	C	R	C	C	I
Programme Python	C	I	R	I	I
Base de données	R	I	I	C	C
Schéma électrique	C	I	C	R	I

R : **Responsible** Responsable d'une tâche

A : **Accountable** Approuve le travail final

C : **Consulted** Personne à solliciter pour demander de l'aide

I : **Informed** Les personnes qui doivent être tenues informées des tâches

XII. Retour d'Expérience individuel :

i. Al-Abri Mohammed :

Ce projet m'a permis de mettre en pratique mes compétences en systèmes embarqués et en développement web. J'ai participé à l'installation du Raspberry Pi, à la configuration du serveur local, et à l'intégration du Pico pour la mesure de la lumière ambiante. J'ai également écrit un script Python pour automatiser certaines fonctions liées à la capture d'images.

Grâce à ce travail, j'ai amélioré mes connaissances en Python, en gestion de base de données, et en communication entre appareils.

C'était une expérience enrichissante qui m'a appris à mieux travailler en équipe et à résoudre des problèmes techniques de manière autonome.

ii. Caillet Adrien :

Ce second semestre, j'ai développé de façon plus concrète la solution technique du projet. En passant de la Base de Données avec SQLite, à la configuration du Raspberry Pi PICO W jusqu'à mise en place du montage électrique sur la breadboard.

Cela m'a permis de m'améliorer sur des aspects techniques alors pas encore abordés au sein de la première partie du projet. Je ressors de ce travail avec plus de connaissances et d'expériences qui pourront m'être utiles.

Malgré les imprévus et difficultés rencontré durant nos travaux, ce fut un vrai plaisir et je suis fier du travail que nous avons fourni. Je suis plus qu'enthousiaste à l'idée de travailler sur de nouveau projet semblable à l'avenir.

iii. Hacherouf Sarah :

Pendant ce deuxième semestre, je me suis principalement occupée de la gestion de projet. J'ai organisé les réunions, je me suis occupé du Gantt, rempli le RACI et géré les fiches d'anomalie.

J'ai aussi participé à la vérification et à la validation des tests pour m'assurer que tout fonctionnait comme prévu. J'ai vérifié la base de données afin de voir si elle était fonctionnelle.

Cette expérience m'a vraiment appris à mieux m'organiser, gérer une équipe, à suivre l'avancement du projet et à m'adapter quand il y avait des imprévus. Je remercie mon équipe pour son sérieux, sa bonne humeur et le soutien tout au long du projet, c'est ce qui a rendu ce semestre aussi enrichissant.

iv. Laurent Arthur :

Pour ma partie du projet, je devais faire un programme sur le Raspberry Pi Pico pour lire la lumière avec un capteur et allumer une LED si la lumière est faible. Je devais aussi envoyer un message au Raspberry Pi 3 pour qu'il puisse prendre une photo.

Au début, j’ai eu du mal à comprendre comment faire fonctionner le capteur et comment envoyer les messages. Grâce à l’aide de mes camarades, j’ai pu avancer et réussir à faire fonctionner le programme correctement.

Cette partie m’a permis d’apprendre à utiliser un capteur avec le Pico et à communiquer avec un autre appareil.

v. Zoubir Rayan :

Ce projet m’a permis de mettre en pratique tout ce que j’ai appris en développement web, notamment en HTML, CSS, PHP et sécurité. J’ai pris en charge la création complète du site, son lien avec la base de données et son interaction avec le Raspberry Pi.

C’était parfois complexe, mais très formateur. J’ai aussi apprécié le travail en équipe, surtout lors des tests finaux où tout devait fonctionner ensemble. Finalement, c’est un projet dont je suis fier.

XIII. Annexes

1. Annexe 1

Rapport de vérification / validation			
ID du test: TEST-R02	**Problème :** Version v2.0	**Système testé :** Système authentication et gestion des utilisateur **Problème SUT :** v2.0	**Vérification Résultat:** FAIL / PASS
Numéro requis : #ACC01-01 : Connexion sécurisée par login/mot de passe #SEC01-07 : Stockage chiffré des mots de passe #USR01-01 à 04 : Gestion utilisateurs par l'admin			

Description du test :

Ce test a pour objectif de valider la sécurité du système d'authentification : création, modification et suppression des utilisateurs par un administrateur, chiffrement des mots de passe et contrôle d'accès aux pages

Préparation au test :

Non.	Description de l'activité
I	Démarrage du serveur web et base de données
II	Connexion avec compte administrateur
III	Création de comptes test et configuration des droits

Exécution des tests :

Non.	Description de l'activité	Critères de réussite/échec	Statut(Réussite, Échec)	Remarques
1	Connexion avec un compte valide	Accès autorisé à l'interface utilisateur	**Réussite**	
2	Connexion avec mot de passe invalide (3 fois)	Compte vérouillé	**Réussite**	Testé avec un compte test
3	Tentive d'accès sans connexion	Redirection vers le login	**Réussite**	

Résumé:

L'ensemble des fonctionnalités critiques liées à la sécurité et à la gestion des utilisateurs a été validé. Le système répond aux exigences d'authentification, de stockage sécurisé des identifiants et de contrôle d'accès. Aucune anomalie n'a été relevée lors de ce test.

Rapport de vérification / validation			
ID du test: TEST-R04	**Problème :** Version v3.0	**Système testé :** Accès sécurité à l'interface web + gestion des comptes **Problème SUT :** v1.4	**Vérification Résultat:** FAIL / PASS
Numéro requis : #ACC01-01 : Authentification par login/mot de passe #ACC01-02 : Droits selon le profil utilisateur #USR01-01 à 06 : Gestion des comptes par l'Admin #SEC01-01 à 09 : Sécurité des mots de passe			

Description du test :

Ce test vise à valider les fonctionnalités de sécurité du site : accès restreint, gestion des utilisateurs par l'administrateur, et politiques de mots de passe robustes, y compris le blocage après échecs.

Préparation au test :

Non.	Description de l'activité
I	Création des utilisateurs (Admin, Opérateur, Super Admin)
II	Configuration des paramètres de mot de passe
III	Déploiement du module d'authentification sécurisé

Exécution des tests :

Non.	Description de l'activité	Critères de réussite/échec	Statut(Réussite, Échec)	Remarques
1	Connexion avec identifiants valides	Accès selon le rôle	**Réussite**	
2	Ajout d'un nouvel Opérateur via l'Admin	Nouveau compte fonctionnel	**Réussite**	
3	Configuration personnalisée de la politique de mot de passe	Longueur, majuscules, etc. respectées	**Réussite**	

Résumé:

L'ensemble des tests de sécurité liés à l'accès, la gestion de comptes, et la robustesse des mots de passe est conforme aux spécifications. Le site permet une authentification sécurisée et une séparation claire des droits selon le profil. Tous les scénarios de blocage, création, et personnalisation des paramètres sont fonctionnels.

Rapport de vérification / validation			
ID du test: TEST-R03	**Problème :** Version v2.1	**Système testé :** Système de capture photo automatique (Raspberry Pi + caméra) **Problème SUT :** v1.3	**Vérification Résultat:** FAIL / PASS
Numéro requis : #PHO01 : Photo après connexion utilisateur #PHO02 : Photo sans connexion utilisateur #PHO03 : Photo auto après 24h d'inactivité #PHO04 : Prise de photo via script Python #PHO05 : Enregistrement/suppression de la photo			

Description du test :

Ce test vise à valider les fonctionnalités de prise de photo via le site web, que ce soit avec ou sans connexion utilisateur, ainsi que les déclenchements automatiques programmés et les options de suppression ou conservation de la photo par l'utilisateur.

Préparation au test :

Non.	Description de l'activité
I	Démarrage du Raspberry Pi avec caméra connectée
II	Déploiement du site web avec bouton de capture
III	Programmation du script Python pour capture automatique

Exécution des tests :

Non.	Description de l'activité	Critères de réussite/échec	Statut(Réussite, Échec)	Remarques
1	Connexion utilisateur et capture depuis l'interface	Photo enregistrée et affichée	Réussite	
2	Déconnexion et capture sans login	Photo possible sans authentification	Réussite	
3	Supression de photo via l'interface	Fichier supprimé, lien invalide	Réussite	

Résumé:

L'ensemble des fonctionnalités liées à la prise de photo automatique et manuelle a été testé avec succès. Le système est conforme aux exigences #PHO01 à #PHO05, et aucune anomalie n'a été détectée. Le test du déclenchement automatique sur inactivité est particulièrement concluant.

2. Annexe 2

2. Exemple de procédure de test

Test n°	Étapes	Résultat attendu
T1	Accéder au site via : http:// ou l'adresse IP	Page de login affichée
T2	Se connecter avec un compte valide	Accès au tableau de bord selon le profil
T3	Observer l'ajout d'un utilisateur par un administrateur	Le nouvel utilisateur est ajouté à la base
T4	Provoquer 3 mauvaises tentatives de connexion	Compte bloqué et message d'erreur affiché
T5	Consulter les logs après diverses actions utilisateur	Chaque action est enregistrée avec ID utilisateur, type d'événement
T6	Couvrir le capteur de luminosité	LED s'allume automatiquement
T7	Lancer une prise de photo depuis l'interface web	Une image est capturée, nommée et stockée

3. Annexe 3

N°	Page concernée	Description de l'anomalie	Niveau	Corrigée ?	Solution mise en place / Recommandée
A1	`voir_photo.php`	Erreur blanche lorsqu'une photo marquée comme supprimée (`supr = 'oui'`) est consultée	Moyen	Oui	Ajout du filtre `supr = 'non'` dans la requête SQL
A2	`back-office.php`	Possibilité de restaurer une photo supprimée sans vérifier si le fichier existe	Moyen	Oui	Ajout d'une vérification avec `file_exists()`
A3	`back-office.php`	Export CSV mal encodé (accents/virgules mal interprétés dans Excel)	Faible	Oui	Utilisation de `fputcsv()` avec tabulation ou guillemets d'encapsulation
A4	`mode-photo.php`	Conflit potentiel : deux utilisateurs à la même seconde génèrent le même nom de photo	Élevé	Oui	Ajout de l'`user_id` dans le nom du fichier photo
A5	`mode-photo.php`	Conflit si deux utilisateurs utilisent le même compte et prennent une photo en simultané	Élevé	Oui	Génération unique du nom + vérification session

A6	`login.php`	Faible protection CSRF si l'utilisateur fait un refresh rapide	**Moyen**	**Oui**	Mise en place d'un token CSRF serveur
A7	`visionnage.php`	Problème de filtre sur la date dans la liste des photos	**Faible**	**Oui**	Correction de la requête SQL avec condition sur la date
A8	`mode-photo.php`	LED automatique non activée sur le Pico WH lors de la capture	**Moyen**	**Oui**	Activation du GPIO dans le script Python
A9	`back-office.php`	**Double soumission possible si F5 après login (essai comptabilisé)**	**Faible**	**Non (non bloquant)**	Anomalie connue, sans effet critique. Non traitée car sans impact majeur.