



Rapport SAE 15/23/24

LABESTE Bastien—SALHI Morgan—MAKHMAEV Jabrail

Université Côte d'Azur, IUT Réseaux & Télécoms

Session 2023-2024

12 juin 2024

1.	Contexte	5
1.1.	Présentation du Projet	5
1.2.	Présentation de notre solution	5
1.3.	Liste des acronymes utilisés	6
1.4.	Exigences	7
1.4.1.	Exigences Générales	7
1.4.2.	Exigences de Sécurité et Authentification.....	7
1.4.3.	Exigences des Utilisateurs	8
1.4.4.	Exigences des Administrateurs	9
1.4.5.	Exigences des Super-Administrateurs	9
1.4.6.	Exigences de l'Assistance Technique	10
1.4.7.	Exigences de l'Interface Web	10
1.4.8.	Base de Données.....	11
2.	Planification du Projet	12
2.1.	Statut des risques	12
2.2.	RACI	19
2.3.	REX individuel (notre comportement / la SAE)	21
2.4.	Planification	23
2.5.	Diagramme de Gantt	24
3.	Raspberry.....	26
3.1.	Installation et Configuration	26
3.1.1.	Description de l'installation	26
3.1.2.	Configuration du Raspberry (système + applications + configuration)	27
3.2.	Configuration de l'adresse IP	29
3.3.	Consommation du Raspberry (annuelle)	31
3.3.1.	Calculs théoriques et hypothèses	31
3.3.2.	Consommation mesurée (réelle) durant l'utilisation du site	32
3.3.3.	Indication de la consommation annuelle prévue/impact environnemental	35
4.	Programmation	36
4.1.	PhpMyAdmin	36
4.2.	Python.....	37
4.3.	HTML et CSS.....	38
4.4.	PHP	40
4.5.	Javascript	41
4.6.	Échange d'Informations entre les Pages et Sécurité du Site	42
5.	Assurance Qualité Logicielle	46

5.1.	Plan de validation	46
5.2.	Procédures de test.....	50
5.3.	Rapports de test et fiches d'anomalies.....	51
6.	Gestion de Configuration des Logiciels	53
6.1.	Utilisation de GitHub	53
6.2.	Gestion des tags.....	54
7.	Annexes	55
7.1.	Annexe A : Procédures de Test.....	55
7.2.	Annexe B : Rapports de Test	56
7.3.	Annexe C : Fiches d'Anomalies	90
7.4.	Annexe D : Matrice de Traçabilité	94
Figure 1 :	Schéma entités/associations	11
Figure 2 :	Exemple de script python servant a mesuré la consommation d'un Raspberry	32
Figure 3 :	Photo du résultat du script lors du lancement du site.....	33
Figure 4 :	Photo du résultat du script lors de l'exportation des bonnes pratiques.....	34
Figure 5 :	Gestion des tables PhpMyAdmin	36
Figure 6 :	Exemple de test de requête via PhpMyAdmin	36
Figure 7 :	Exemple de code Python utilisé pour générer un PDF	37
Figure 8 :	Exemple de formulaire html	38
Figure 9 :	Exemple de code CSS	39
Figure 10 :	Partie du Code PHP pour la Gestion de la Suppression des Bonnes Pratiques.....	40
Figure 11 :	Fonction Javascript qui charge les options du filtre.....	41
Figure 12 :	Exemple d'utilisation de sessions PHP	42
Figure 13 :	Exemple de création d'un fichier CSV pour échanger des données	43
Figure 14 :	Exemple d'ouverture d'un fichier CSV pour échanger des données	44
Figure 15 :	Exemple de suppression d'un fichier pour éviter les doublons	44
Figure 16 :	Exemple de requête préparée avec des placeholders "?"	45
Figure 17 :	Exemple de vérification de connexion	45
Figure 18 :	Exemple de chiffrement de mot de passe	45
Figure 19 :	Exemple d'incrémentation du nombre de tentative de connexion effectuée	46
Figure 20 :	Exemple de message d'erreur et de redirection	46

Tableau 1 : Liste des acronymes	6
Tableau 2 : Risque n°001.....	13
Tableau 3 : Risque n°002.....	14
Tableau 4 : Risque n°003.....	15
Tableau 5 : Risque n°004.....	16
Tableau 6 : Risque n°005.....	17
Tableau 7 : Risque N°006	18
Tableau 8 : RACI envisagé	19
Tableau 9 : RACI réel.....	20
Tableau 10 : Diagramme de GANTT	25
Tableau 11 : Plan de validation sur la Création de compte	46
Tableau 12 : Plan de validation sur les comptes utilisateurs	48
Tableau 13 : Plan de validation sur les comptes administrateurs et super-administrateurs	48
Tableau 14 : Plan de validation sur l'assistance technique	49
Tableau 15 : Plan de validation sur l'interface web.....	50
Tableau 16 : Liste des procédures de test.....	55
Tableau 17 : Rapport de test 1.....	57
Tableau 18 : Rapport de test 2.....	59
Tableau 19 : Rapport de test 3.....	61
Tableau 20 : Rapport de test 4.....	63
Tableau 21 : Rapport de test 5.....	65
Tableau 22 : Rapport de test 6.....	67
Tableau 23 : Rapport de test 7	69
Tableau 24 : Rapport de test 8.....	71
Tableau 25 : Rapport de test 9.....	73
Tableau 26 : Rapport de test 10.....	75
Tableau 27 : Rapport de test 11.....	77
Tableau 28 : Rapport de test 12.....	79
Tableau 29 : Rapport de test 13.....	81
Tableau 30 : Rapport de test 14.....	83
Tableau 31 : Rapport de test 15.....	85
Tableau 32 : Rapport de test 16.....	87
Tableau 33 : Rapport de test 17.....	89
Tableau 34 : Fiche d'anomalie 1	91
Tableau 35 : Fiche d'anomalie 2	92
Tableau 36 : Fiche d'anomalie 3	93
Tableau 37 : Matrice de traçabilité	94

1. Contexte

1.1. Présentation du Projet

This project aims to enhance the safety and efficiency of avionics testing by identifying and sharing best practices.

The goal is to create a secure and user-friendly website where practices can be defined using keywords, providing visibility into program-specific details and allowing selection of specific phases to access relevant insights. Additionally, it enables exporting all this information into a CSV file format.

As a user, one can manage passwords, select programs to view associated insights, and even add or delete information for particular programs and phases.

Administrators will have additional permissions such as creating or deleting accounts, modifying passwords, or permanently erasing specific information for precise programs and phases.

To achieve this, we will develop a Python application. This app will export all these insights based on keywords into PDF or Excel files.

Our objective is to create a platform to access these valuable insights, allowing modifications if necessary, and contribute to enhancing the safety and efficiency of avionics testing while adhering to standards and requirements

1.2. Présentation de notre solution

L'application CHECKLIST est conçue pour optimiser la gestion et le partage des bonnes pratiques. Elle se compose de deux parties principales :

Application Web :

Développée en PHP, HTML, CSS et JavaScript, cette partie de l'application est responsable de l'interface utilisateur. Elle offre les fonctionnalités suivantes :

- Création, duplication, recherche et suppression des bonnes pratiques : Les utilisateurs peuvent facilement gérer les bonnes pratiques. Celles-ci peuvent être triées en fonction de différents programmes et filtrées via des mots-clés.
- Visualisation des bonnes pratiques : Les utilisateurs peuvent visualiser les bonnes pratiques directement sur le site web.
- Exportation des données : Les données peuvent être exportées en formats Excel et PDF pour une utilisation ultérieure.
- Gestion des comptes utilisateurs : Les administrateurs peuvent créer, modifier et supprimer des comptes utilisateurs avec des droits étendus.
- Sécurité et authentification : L'application permet l'inscription, la connexion, la déconnexion et la modification des mots de passe. Les mots de passe sont stockés de manière chiffrée et des mesures de sécurité sont en place pour protéger les comptes (e.g., blocage après trois tentatives de connexion infructueuses).

Gestion de la Base de Données (BDD) :

La base de données stocke toutes les informations relatives aux bonnes pratiques, aux programmes et aux phases de test. Elle est gérée via phpMyAdmin pour assurer une maintenance et une accessibilité efficaces. Voici les points clés de la gestion de la BDD :

- Gestion des comptes administrateurs : Les administrateurs ont des droits étendus pour gérer les comptes utilisateurs et les programmes.
- Contrôle des accès et des permissions : Différents niveaux de permissions sont définis pour les utilisateurs (Super Administrateur, Administrateur, Opérateur) pour contrôler l'accès aux différentes fonctionnalités.
- Historique et traçabilité : Les actions des utilisateurs sont mémorisées pour des fins d'investigation et de suivi.

1.3. Liste des acronymes utilisés

Le tableau ci-dessous présente les acronymes utilisés tout au long de ce rapport ainsi que leurs significations correspondantes. Cette liste vise à faciliter la compréhension des termes techniques et des abréviations couramment employés dans le contexte du projet.

Tableau 1 : Liste des acronymes

Acronyme	Signification
RACI	Responsable, Accountable, Consulted, Informed
BDD	Base de Données
SAE	Situation d'Apprentissage et d'Évaluation
VSCode	Visual Studio Code
REX	Retour d'EXpérience
MAMP	Macintosh, Apache, MySQL, PHP
GitHub	Platform for version control and collaboration
Thalès11	Nom de l'organisation pour le projet
P	Phase ou Planification (selon le contexte)
A	Bastien (dans le contexte du diagramme de Gantt)
B	Morgan (dans le contexte du diagramme de Gantt)
C	Jabrail (dans le contexte du diagramme de Gantt)

1.4. Exigences

1.4.1. Exigences Générales

Référence CHECKLIST-gene-100 : L'application SAE15 sera réalisée en utilisant HTML, CSS, JavaScript, PHP et Python.

Référence CHECKLIST-gene-110 : L'application CHECKLIST sera réalisée avec des bibliothèques, outils, applications libres de droit pour les entreprises.

Référence CHECKLIST-gene-120 : Le code de l'application CHECKLIST sera commenté à 80% au minimum.

Référence CHECKLIST-gene-130 : L'application CHECKLIST devra être compatible avec un système d'exploitation Linux.

Référence CHECKLIST-gene-140 : L'application CHECKLIST devra être écologique, verte.

Référence CHECKLIST-gene-150 : L'application CHECKLIST devra être codée sur un Raspberry.

1.4.2. Exigences de Sécurité et Authentification

Création de Compte :

Référence CHECKLIST-compte-210 : L'identifiant ne doit pas contenir d'accent.

Référence CHECKLIST-compte-220 : L'identifiant doit être unique.

Référence CHECKLIST-compte-230 : L'identifiant doit contenir entre 8 et 30 caractères composés uniquement de chiffres et de lettres.

Référence CHECKLIST-compte-240 : Le mot de passe ne doit pas contenir l'identifiant de l'utilisateur.

Référence CHECKLIST-compte-250 : Le mot de passe ne doit pas contenir d'accent.

Référence CHECKLIST-compte-260 à CHECKLIST-compte-290 : Le mot de passe doit contenir les caractères indiqués lors de la création de compte ou modification de mot de passe.

Référence CHECKLIST-compte-300 : Le mot de passe doit être stocké sous forme chiffrée.

Référence CHECKLIST-compte-310 : Le compte doit être bloqué après 3 tentatives de connexion infructueuses.

Référence CHECKLIST-compte-320 : Les paramètres n, p, q, r doivent être configurables uniquement par un utilisateur de type « Administrateur » ou « Super-administrateur ».

Authentification :

Référence CHECKLIST-Auth-400 : L'application doit permettre de s'inscrire en fournissant un identifiant et un mot de passe.

Référence CHECKLIST-Auth-410 : L'application doit permettre de se connecter en fournissant un mot de passe et un identifiant.

Référence CHECKLIST-Auth-420 : L'application doit permettre de se déconnecter.

Référence CHECKLIST-Auth-430 : L'application doit permettre de modifier son mot de passe.

Référence CHECKLIST-Auth-440 : L'application CHECKLIST doit limiter l'accès aux différentes fonctionnalités en fonction du profil concerné : Super Administrateur, Administrateur, Operateur.

1.4.3. Exigences des Utilisateurs

Référence CHECKLIST-Utilisateur-500 : L'application CHECKLIST doit permettre de sélectionner un programme pour voir toutes les « bonnes pratiques » associées.

Référence CHECKLIST-Utilisateur-510 : L'application CHECKLIST doit permettre de sélectionner une phase pour voir toutes les « bonnes pratiques » associées.

Référence CHECKLIST-Utilisateur-520 : L'application CHECKLIST doit permettre d'écrire des mots-clés pour lister les « bonnes pratiques » correspondantes.

Référence CHECKLIST-Utilisateur-530 : L'application CHECKLIST doit permettre d'exclure un ou plusieurs programmes pour afficher les « bonnes pratiques » restantes.

Référence CHECKLIST-Utilisateur-540 : L'application CHECKLIST doit permettre d'exclure une ou plusieurs phases pour afficher les « bonnes pratiques » restantes.

Référence CHECKLIST-Utilisateur-550 : L'application CHECKLIST doit permettre d'exclure un ou plusieurs mots-clés pour afficher les « bonnes pratiques » restantes.

Référence CHECKLIST-Utilisateur-560 : L'application CHECKLIST doit permettre d'écrire de nouvelles bonnes pratiques pour un programme donné et une phase donnée.

Référence CHECKLIST-Utilisateur-570 : L'application CHECKLIST doit permettre de dupliquer des bonnes pratiques d'un autre programme pour un programme donné et une phase donnée.

Référence CHECKLIST-Utilisateur-580 : L'application CHECKLIST doit permettre de supprimer des bonnes pratiques pour un programme donné et une phase donnée.

1.4.4. Exigences des Administrateurs

Référence CHECKLIST-admin-600 : L'application doit posséder des comptes administrateurs.

Référence CHECKLIST-admin-610 : Les comptes administrateurs peuvent supprimer/ajouter des comptes opérateurs.

Référence CHECKLIST-admin-620 : Les comptes administrateurs peuvent contrôler les accès des utilisateurs.

Référence CHECKLIST-admin-630 : Les comptes administrateurs peuvent afficher la liste d'utilisateurs par ordre alphabétique.

Référence CHECKLIST-admin-650 : Les comptes administrateurs peuvent voir l'historique d'interactions des utilisateurs.

Référence CHECKLIST-admin-660 : L'application CHECKLIST doit permettre à un utilisateur de type « Administrateur » ou « Super-administrateur » de modifier le mot de passe d'un utilisateur dont le compte est verrouillé.

Référence CHECKLIST-admin-670 : Les comptes administrateurs peuvent créer ou supprimer des programmes.

Référence CHECKLIST-admin-680 : Les comptes administrateurs peuvent supprimer définitivement des bonnes pratiques pour un programme donné et une phase donnée.

Référence CHECKLIST-admin-690 : L'application CHECKLIST permet aux « Administrateurs » d'effectuer les mêmes actions qu'un utilisateur « Opérateur ».

1.4.5. Exigences des Super-Administrateurs

Référence CHECKLIST-SuperAdmin-700 : L'application doit posséder un seul compte super-administrateur.

Référence CHECKLIST-SuperAdmin-710 : Le compte super-administrateur peut donner ou retirer les droits d'administrateur à un compte.

Référence CHECKLIST-SuperAdmin-720 : Le compte super-administrateur peut supprimer les comptes administrateur.

Référence CHECKLIST-SuperAdmin-730 : L'application CHECKLIST ne doit pas bloquer le compte du « Super Administrateur ».

Référence CHECKLIST-SuperAdmin-740 : L'application CHECKLIST permet au « Super Administrateur » d'effectuer les mêmes actions qu'un utilisateur « Administrateur ».

Référence CHECKLIST-SuperAdmin-750 : Le login / mot de passe du « Super Administrateur » sera indiqué oralement au tuteur.

1.4.6. Exigences de l'Assistance Technique

Référence CHECKLIST-assistance-810 : L'application CHECKLIST doit permettre d'accéder à un guide d'utilisation.

Référence CHECKLIST-assistance-820 : L'application CHECKLIST doit mémoriser les événements de type : Informations, Warnings, Alarmes.

Référence CHECKLIST-assistance-830 : L'application CHECKLIST doit mémoriser les actions des différents utilisateurs, avec au minimum :

- La date,
- L'utilisateur connecté,
- Le profil de l'utilisateur,
- Le type de l'événement,
- La description de l'action effectuée.

1.4.7. Exigences de l'Interface Web

Référence CHECKLIST-web-900 : L'interface web doit posséder un menu de navigation avec les éléments suivants :

- Guide d'utilisation
- Création de bonnes pratiques (BP)
- Visualisation des logs
- Modification du mot de passe des utilisateurs
- Gestion des bonnes pratiques désactivées
- Suppression de comptes
- Modification des droits des comptes
- Création de comptes

Référence CHECKLIST-web-910 : L'interface web doit permettre le filtrage des bonnes pratiques par phase et programme.

Référence CHECKLIST-web-920 : L'interface web doit permettre la recherche par mots-clés.

Référence CHECKLIST-web-930 : L'interface web doit offrir des actions de masse telles que :

- Exporter les bonnes pratiques sélectionnées
- Dupliquer les bonnes pratiques sélectionnées
- Désactiver les bonnes pratiques sélectionnées
- Supprimer définitivement les bonnes pratiques sélectionnées (pour les administrateurs et super-administrateurs)

Référence CHECKLIST-web-940 : L'interface web doit s'adapter automatiquement à la taille de l'écran (19 pouces minimum).

Référence CHECKLIST-web-950 : L'interface web doit ouvrir lors de son lancement une page de connexion qui permettra de s'authentifier.

Référence CHECKLIST-web-960 : La page de connexion doit posséder un bouton pour se créer un compte.

Référence CHECKLIST-web-960.1 : La page de création de compte doit afficher un message d'erreur si l'utilisateur ne respecte pas les conditions pendant la création de compte.

Référence CHECKLIST-web-960.2 : La page de création de compte stockera les identifiants cryptés dans une base de données.

Référence CHECKLIST-web-970 : L'interface web doit avoir un bon contraste visuel.

Référence CHECKLIST-web-980 : L'interface web devra avoir des boutons supplémentaires en cas de connexion avec un compte administrateur ou super-administrateur.

Référence CHECKLIST-web-990 : Les boutons supplémentaires permettront d'utiliser toutes les fonctionnalités des programmes admin et super-admin.

Référence CHECKLIST-web-990.1 : Lors de la tentative d'effacement d'un compte ou d'un programme, l'interface web demandera une confirmation à l'utilisateur avant de supprimer.

1.4.8. Base de Données

Référence CHECKLIST-bdd-1000 : L'application implémentera le schéma Entités/Associations suivant :

Schéma entités/associations :

- Les Utilisateurs gèrent plusieurs programmes et un programme sera affiché pour plusieurs utilisateurs.
- Un Programme contient plusieurs phases et une phase peut être contenue dans plusieurs programmes.
- Une phase inclut plusieurs bonnes pratiques et une bonne pratique peut être incluse dans plusieurs phases.

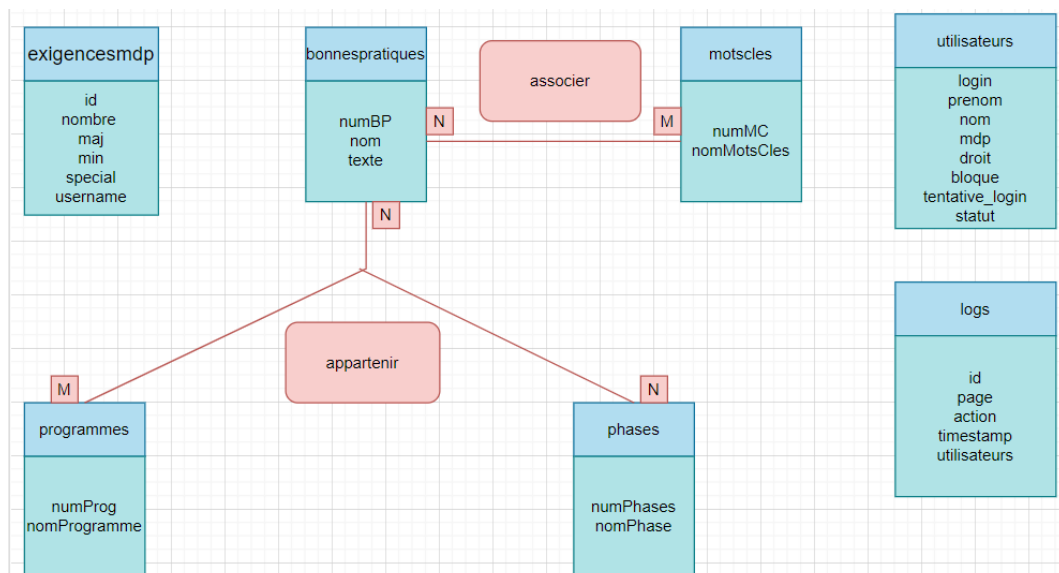


Figure 1 : Schéma entités/associations

Schéma logique :

<u>xxx</u> : clé primaire #xxx : clé étrangère

- utilisateurs (login, prénom, nom, mdp, droit, bloque, tentative_login, statut)
- programmes (numProg, nomProgramme)
- phases (numPhases, nomPhase)
- bonnespratiques (numBP, nom, texte)
- motscles (numMC, nomMotsCles)
- appartenance (numAppart, #BP, #Programme, #Phases, active)
- association (numAssoc, #BP, #motscles)
- logs (id, page, action, timestamp, utilisateurs)
- exigencesmdp(id, nombre, maj, min, special, username)

2. Planification du Projet

2.1. Statut des risques

Cette section présente une revue des risques identifiés lors de la première partie du projet (SAE15). Chaque risque sera décrit en termes de causes, conséquences, actions de gestion mises en place, et l'état actuel de ces actions. Cette revue permet d'évaluer l'efficacité des mesures prises et de déterminer si des ajustements sont nécessaires. Une matrice des risques est également fournie pour visualiser l'importance relative de chaque risque.

Indice du risque	Gravité	Probabilité
Rouge	Élevée	Élevée
Jaune	Moyenne	Moyenne
Vert	Faible	Faible

Tableau 2 : Risque n°001

Registre des risques						
Projet : Checklist	Organisation : Thalès11	Source : Interne (Ressources humaine) Contrôlé par : Bastien Labeste Supporté par : Thalès11		Date : 05/12/2023		
Risque du scénario et son ampleur						
N°001	Nom du risque : Départ inattendu de membres clés de l'équipe					
Causes et Conséquences : -Cause possible - difficultés académiques, désir de réorientation -Conséquence - perte de ressources clés pour le projet						
Gravité	Probabilité	Indice du risque	Rouge	Jaune	Vert	Domaine du risque
Elevé	Moyenne	Elevé				Ressource humaine
Risk Decision and Action						
Accepter le risque :		non		Réduire le risque : oui		
Mesures à prendre :		Moyens de vérification :		Réduction attendue :		
-Mise en place d'aides pour les devoirs -Ajustement de la charge de travail		Vérifications régulières des ressentis du groupe		Atténuation des difficultés académiques		
Actions prises : Création d'un groupe de classe et entraide			Statut : réalisé			
Approuvé par la Direction du Projet : Oui			Rang du risque : 1			
Nom : Makhmaev Jabrail Date : 17/12/2023		Signature :				

Débriefing : Timothée a quitté l'équipe en raison de difficultés académiques et d'un désir de réorientation, malgré les mesures prises. En conséquence, chaque membre du projet a dû prendre en charge des tâches supplémentaires, ce qui a augmenté leur charge de travail individuelle et affecté l'avancement global du projet.

Tableau 3 : Risque n°002

Registre des risques						
Projet : Checklist	Organisation : Thalès11	Source : Interne (Gestion du projet) Contrôlé par : Bastien Labeste Supporté par : Thalès11	Date : 05/12/2023			
Risque du scénario et son ampleur						
N°002	Nom du risque : Mauvaise Gestion des Délais					
Causes et Conséquences :						
-Cause possible – Différences dans les opinions et les styles de travail au sein du groupe						
-Conséquence - Conflits au sein de l'équipe, diminution de la productivité.						
Gravité	Probabilité	Indice du risque	Rouge	Jaune	Vert	Domaine du risque
Moyenne	Élevé	Elevé				Gestion de Projet
Risk Decision and Action						
Accepter le risque :		non	Réduire le risque :		oui	
Mesures à prendre :		Moyens de vérification :		Réduction attendue :		
-Élaboration d'un plan de projet détaillé avec des jalons clairs. -Utilisation d'outils de gestion de projet pour le suivi et la planification. -Mise en place de rappels réguliers et de revues d'avancement.		- Suivi du plan de projet, rapports d'avancement réguliers.		Respect des délais		
Actions prises : Mise en place d'un groupe de projet sur discord, utilisation d'outils de gestion tel qu'un RACI, utilisation d'un wiki			Statut : réalisé			
Approuvé par la Direction du Projet : Oui			Rang du risque : 2			
Nom : Makhmaev Jabrail		Signature :				
Date : 17/12/2023						

Débriefing : Les divergences dans les styles de travail ont contribué à une mauvaise gestion des délais, causant une diminution de la productivité et des retards dans le projet. Malgré les plans détaillés et les outils de gestion mis en place, les résultats n'ont pas été à la hauteur des attentes.

Tableau 4 : Risque n°003

Registre des risques						
Projet : Checklist	Organisation : Thalès11	Source : Externe (Matériel) Contrôlé par : Bastien Labeste Supporté par : Thalès11	Date : 05/12/2023			
Risque du scénario et son ampleur						
N°003	Nom du risque : Défaillance du matériel de développement BlackBerry					
Causes et Conséquences :						
-Cause possible – panne matérielle, obsolescence -Conséquence - retard dans le développement, coûts supplémentaires						
Gravité	Probabilité	Indice du risque	Rouge	Jaune	Vert	Domaine du risque
Elevé	Faible	Moyen				Matériel
Risk Decision and Action						
Accepter le risque : oui			Réduire le risque : non			
Mesures à prendre :		Moyens de vérification :		Réduction attendue :		
-Mettre en place un plan de maintenance préventive pour le matériel. -Avoir des ressources de remplacement disponibles en cas de défaillance		-Suivi des activités de maintenance préventive -Disponibilité des ressources de remplacement		-Minimisation des risques de défaillance -Réduction des impacts en cas de panne.		
Actions prises : Aucunes			Statut : réalisé			
Approuvé par la Direction du Projet : Oui			Rang du risque : 3			
Nom : Makhmaev Jabrail Date : 17/12/2023		Signature :				

Débriefing : Le Raspberry Pi utilisé pour le développement a subi une défaillance matérielle, ce qui a causé un retard significatif dans le développement du projet. Des ressources de remplacement n'étaient pas immédiatement disponibles, exacerbant les retards.

Tableau 5 : Risque n°004

Registre des risques						
Projet : Checklist	Organisation : Thalès11	Source : Interne (Membres de l'équipe) Contrôlé par : Bastien Labeste Supporté par : Thalès11	Date : 05/12/2023			
Risque du scénario et son ampleur						
N°004	Nom du risque : Divergences d'interprétation des exigences du client					
Causes et Conséquences :						
-Cause possible – manque de communication, compréhension variable des exigences -Conséquence - livrables ne répondant pas aux attentes du client, retards						
Gravité	Probabilité	Indice du risque	Rouge	Jaune	Vert	Domaine du risque
Elevé	Moyenne	Elevé				Ressource humaine
Risk Decision and Action						
Accepter le risque : non			Réduire le risque : oui			
Mesures à prendre :		Moyens de vérification :		Réduction attendue :		
- Organiser des réunions régulières de clarification -Utiliser des outils de gestion de projet pour suivre et documenter les différentes interprétations.		Alimentation du Wiki		Nullification du risque		
Actions prises : Mise en place d'un Wiki, d'un groupe de projet sur discord, utilisation d'outils de gestion tel qu'un RACI			Statut : réalisé			
Approuvé par la Direction du Projet : Oui			Rang du risque : 4			
Nom : Makhmaev Jabrail		Signature :				
Date : 17/12/2023						

Débriefing : Un ou plusieurs membres de l'équipe ont accumulé un retard significatif dans leur travail en raison de divergences dans l'interprétation des exigences. Cela a entraîné une surcharge de travail pour d'autres membres de l'équipe qui ont dû compenser ces retards pour maintenir le projet sur les rails.

Tableau 6 : Risque n°005

Registre des risques						
Projet : Checklist	Organisation : Thalès11	Source : Interne Contrôlé par : Bastien Labeste Supporté par : Thalès11			Date : 05/12/2023	
Risque du scénario et son ampleur						
N°005	Nom du risque : Différences d'opinions, styles de travail divergents					
Causes et Conséquences :						
-Cause possible – Différences dans les opinions et les styles de travail au sein du groupe						
-Conséquence - Conflits au sein de l'équipe, diminution de la productivité.						
Gravité	Probabilité	Indice du risque	Rouge	Jaune	Vert	Domaine du risque
Moyenne	Élevé	Elevé				Dynamique de groupe
Risk Decision and Action						
Accepter le risque :		non	Réduire le risque :		oui	
Mesures à prendre :		Moyens de vérification :		Réduction attendue :		
-Encourager la communication ouverte et le partage d'idées. -Définir des rôles et responsabilités clairs pour chaque membre du groupe.		-Feedback lors des réunions d'équipe		-Amélioration de la communication -Réduction des conflits potentiels.		
Actions prises : Mise en place d'un groupe de projet sur discord, utilisation d'outils de gestion tel qu'un RACI			Statut : réalisé			
Approuvé par la Direction du Projet : Oui			Rang du risque : 5			
Nom : Makhmaev Jabrail		Signature :				
Date : 17/12/2023						

Débriefing : Les différences dans les styles de travail et les opinions ont conduit à une diminution de la productivité au sein de l'équipe. Les conflits internes ont retardé la progression du projet, malgré les efforts pour améliorer la communication.

Tableau 7 : Risque N°006

Registre des risques						
Projet : Checklist	Organisation : Thalès11	Source : Interne (Gestion du projet) Contrôlé par : Bastien Labeste Supporté par : Thalès11	Date : 05/12/2023			
Risque du scénario et son ampleur						
N°006	Nom du risque : Difficultés à faire les taches demandés					
Causes et Conséquences :						
-Cause possible – Manque de compétences techniques, manque d'investissement						
-Conséquence - Difficultés dans la mise en œuvre technique, risque de retards.						
Gravité	Probabilité	Indice du risque	Rouge	Jaune	Vert	Domaine du risque
Moyenne	Moyenne	Elevé				Gestion de Projet
Risk Decision and Action						
Accepter le risque :		non	Réduire le risque :		oui	
Mesures à prendre :		Moyens de vérification :		Réduction attendue :		
-Identifier les lacunes de compétences au début du projet. -Adapter la gestion des taches		- Retours d'expérience sur les taches et discussions régulières		Amélioration des compétences nécessaires à la réalisation des taches		
Actions prises : Adaptation du RACI, groupe de travail sur Discord			Statut : réalisé			
Approuvé par la Direction du Projet : Oui			Rang du risque : 6			
Nom : Makhmaev Jabrail		Signature :				
Date : 17/12/2023						

Débriefing : Un ou plusieurs membres du groupe ont manqué d'investissement, ce qui a causé des difficultés dans la mise en œuvre technique. En conséquence, d'autres membres ont dû prendre en charge des tâches supplémentaires pour compenser, augmentant ainsi leur charge de travail.

Conclusion : La gestion des risques identifiés lors de la phase 1 a montré que plusieurs de ces risques se sont concrétisés, malgré les mesures préventives mises en place. Les actions correctives ont permis de limiter certains impacts, mais n'ont pas été suffisantes pour éviter les retards et les surcharges de travail. Les départs inattendus de membres clés, les divergences d'interprétation des exigences, la défaillance du matériel et les différences de styles de travail ont tous contribué à des défis significatifs tout au long du projet. Cependant, grâce à la détermination et aux efforts supplémentaires des membres de l'équipe, le projet a pu être mené à bien. Cette expérience a mis en lumière l'importance de la gestion proactive des risques et la nécessité de disposer de plans d'urgence robustes pour faire face aux imprévus.

2.2. RACI

Légende :

- R - Responsable (Responsible) : Celui qui fait le travail.
- A - Autorité (Accountable) : Celui qui est en charge et a le dernier mot.
- C - Consulté (Consulted) : Celui qui est consulté pour des avis ou des informations.
- I - Informé (Informed) : Celui qui est tenu au courant de l'avancement.

RACI envisagé :

Tableau 8 : RACI envisagé

Tâche	Bastien	Morgan	Jabrail
Statut des risques, RACI, planification, REX équipe sur la dynamique et la contribution de chacun	R	A	C
Configuration du Raspberry Pi	R	C	A
Mise à jour du rapport de projet	R	A	C
Mise à jour de la base de données (BDD)	C	R	A
Écriture du schéma BDD en SQL	A	R	C
Documentation du code (commentaires, nommage des variables et fonctions en anglais)	R	A	C
Développement de l'interface Web et de l'application Web	A	C	R
Tests unitaires et d'intégration du code	C	R	A

Tâche	Bastien	Morgan	Jabrail
Tests de validation et rapports de test	A	R	C
Création des scénarios de test	A	R	C
Réponse aux questions et support utilisateur	C	A	R
Gestion des fiches d'anomalies	A	R	C
Description de l'installation et configuration du Raspberry Pi (système + applications + configuration)	R	A	C
Création de la matrice de traçabilité	C	R	A

RACI réel :

Tableau 9 : RACI réel

Tâche	Bastien	Morgan	Jabrail
Statut des risques, RACI, planification, REX équipe sur la dynamique et la contribution de chacun	R	R	R
Configuration du Raspberry Pi	R	C	A
Mise à jour du rapport de projet	R	A	C
Mise à jour de la base de données (BDD)	R	R	A
Écriture du schéma BDD en SQL	R	A	C
Documentation du code (commentaires, nommage des variables et fonctions en anglais)	R	A	R
Développement de l'interface Web et de l'application Web	R	C	R
Tests unitaires et d'intégration du code	A	R	R

Tâche	Bastien	Morgan	Jabrail
Tests de validation et rapports de test	A	R	C
Création des scénarios de test	A	R	C
Réponse aux questions et support utilisateur (GitHub, ...)	R	A	R
Gestion des fiches d'anomalies	A	R	C
Description de l'installation et configuration du Raspberry Pi (système + applications + configuration)	R	A	C
Création de la matrice de traçabilité	R	R	A

2.3. REX individuel (notre comportement / la SAE)

Bastien :

Pendant le projet Checklist, j'ai ressenti fortement l'impact d'une équipe réduite. Initialement prévu pour être mené par quatre ou cinq membres, nous n'étions que trois. Cela a ajouté une pression supplémentaire sur chaque membre, et j'ai particulièrement ressenti ce poids. Mes coéquipiers avaient des approches de travail différentes de la mienne, ce qui a parfois compliqué notre collaboration.

Mon engagement dans ce projet a été total. J'ai participé activement à toutes les phases du projet, de la planification à la mise en œuvre, en passant par les tests et la documentation. J'ai pris en charge une grande partie du développement du site web, ce qui a représenté une part significative du travail global. J'ai maintenu une communication ouverte et régulière avec mes coéquipiers via les outils de communication disponibles (Discord, emails). Malgré nos différences de style de travail, nous avons toujours cherché à faciliter la collaboration en partageant des mises à jour régulières et en s'aidant mutuellement.

Le développement intensif du site web m'a permis de renforcer mes compétences en HTML, CSS, PHP et JavaScript. J'ai également acquis des compétences supplémentaires en gestion de bases de données avec phpMyAdmin. J'ai appris à mieux structurer et documenter le code, ce qui est essentiel pour la maintenance et la future extensibilité du projet. La gestion du temps a été cruciale. J'ai dû planifier mon travail de manière rigoureuse pour accomplir toutes les tâches nécessaires, en particulier à cause de la charge de travail supplémentaire due à notre équipe réduite. J'ai utilisé des outils de gestion de projet comme le diagramme de Gantt et des listes de tâches pour suivre les progrès et m'assurer que toutes les tâches étaient couvertes.

Le départ de Timothée, en raison de difficultés académiques et d'un désir de réorientation, a été un coup dur. J'ai dû reprendre une partie de ses tâches, ce qui a augmenté ma charge de travail. J'ai essayé de compenser en planifiant mieux mon temps et en priorisant les tâches critiques. Nos différences de styles de travail ont parfois causé des frictions et des retards.

J'ai tenté de minimiser l'impact de ces différences en trouvant des terrains d'entente et en organisant des réunions régulières pour aligner nos efforts. En raison des retards et de la répartition inégale du travail, j'ai souvent dû prendre en charge des tâches supplémentaires pour assurer l'avancement du projet.

Malgré ces défis, ce projet a été une expérience enrichissante. J'ai développé de nombreuses compétences techniques et amélioré ma capacité à gérer des projets et des équipes. Cette expérience a également mis en lumière l'importance de la répartition équitable des tâches et de la motivation de tous les membres de l'équipe. Si je devais refaire ce projet, j'insisterais sur une meilleure répartition des tâches dès le début et sur la mise en place de mécanismes pour encourager une participation plus active de tous les membres.

Morgan :

La SAE dans laquelle nous avons pu participer en collaboration avec l'entreprise Thales était un très bon projet dans son ensemble, le fait que toutes les compétences que l'on a acquises au cours de l'année puissent toutes se mettre dans ce même projet.

Travailler avec Thales était déjà très stimulant car c'est une entreprise de renom et très connu dans notre domaine, et le fait qu'en plus de cela être parmi l'un des groupes à avoir réalisé le meilleur site nous permettait d'avoir une visite guidée dans l'entreprise et que les équipes de Thalès reprennent le meilleur site met un réel objectif, une utilité à ce projet, ce qui le rend très stimulant.

Cependant, au cours de l'année un membre de notre équipe nous a quittés, nous laissant en groupe de 3. Cette perte a eu un impact significatif sur notre rythme car plus de travail demandé et des phases de démotivation se sont installés pour ma part.

Malgré tout, nous avons réussi à nous reprendre en main grâce à notre chef d'équipe Bastien qui a parfaitement assumé son rôle.

Pour conclure, cette SAE a été une expérience enrichissante qui nous a permis de voir de quoi nous pouvions être capable en réunissant toutes les compétences vues au cours de l'année. Elle avait aussi un objectif concret, celui de réellement servir pour les équipes de Thales.

Jabrail :

C'est la première fois que je travaille sur un projet scolaire qui a duré toute l'année, j'ai trouvé que le faire en collaboration avec une entreprise comme Thales, c'est une bonne idée, pour pouvoir en quelques sortes s'immerger, et en plus de ça mieux connaître cette entreprise avec la présence d'un des employés qui nous accompagne et qui aide pour mieux comprendre l'objectif de ce projet.

Pour finir correctement ce projet, j'ai réalisé à quel point il fallait avoir une bonne organisation des tâches réparties dans le groupe.

Pour réaliser ce projet-là, on a eu des cours sur les différents langages de programmations nécessaires pour le faire et que je ne connaissais pas avant, et que les cours nous ont permis de les approfondir.

Ce projet m'a aussi permis de découvrir de nouveaux outils qui sont pratiques comme GitHub, MAMP pour les serveurs locaux, et phpMyAdmin. Et aussi de nouvelles technologies comme la carte Raspberry qui était utile pour faire le projet.

J'étais un peu confus sur ce qui était demandé par l'encadrant au début de l'année, mais j'ai pu ensuite comprendre ce qui était attendu à la fin du projet.

Je m'entends bien avec mon groupe, au premier semestre, on était 4 mais y en a un qui s'est réorienté, mais on a pu avancer en étant à 3 avec mon camarade Morgan et le chef de projet Bastien.

Ce projet nous permet vraiment de développer des compétences que je n'aurais jamais acquises autrement, c'est ça que je trouve le plus intéressant.

2.4. Planification

La planification est une étape cruciale dans la gestion de tout projet. Elle permet de définir les différentes phases du projet, d'assigner des tâches aux membres de l'équipe, de fixer des délais et de prévoir les ressources nécessaires. Cette section détaille le processus de planification du projet SAE, en mettant en évidence les outils utilisés, les jalons clés, et les ajustements effectués en cours de route.

Outils de Planification Utilisés :

Pour structurer et suivre l'avancement du projet, nous avons utilisé plusieurs outils de planification :

- Diagramme de Gantt : Cet outil a permis de visualiser les différentes tâches du projet, leurs durées, et leurs interdépendances.
- Calendrier partagé : Pour synchroniser les réunions et les jalons importants.

Décomposition du Projet en Phases :

Phase 1 : Planification et Préparation

- Configurer le Raspberry : 06/02/2024 - 15/02/2024
- Mise à jour du rapport : 06/02/2024 - 07/03/2024
- Statut des risques : 06/02/2024 - 07/03/2024
- Mise à jour de la BDD : 06/02/2024 - 07/03/2024
- Écriture de la BDD en SQL : 06/02/2024 - 07/03/2024
- Faire une liste des scénarios de tests (Plan de Validation) : 06/02/2024 - 28/03/2024
- Description de l'installation du Raspberry : 06/02/2024 - 07/03/2024
- Configurer le Raspberry sur une adresse IP particulière : 15/02/2024 - 07/03/2024

Phase 2 : Développement

- Interface WEB, Appli WEB : 10/03/2024 - 30/05/2024
- Programme informatique : commentaires, nom des variables et des fonctions en anglais : 10/03/2024 - 30/05/2024
- Tests codes : 01/06/2024 - 10/06/2024
- Tests rapports : 01/06/2024 - 10/06/2024
- Gérer le GitHub : 10/03/2024 - 10/06/2024

Phase 3 : Finalisation et Test

- Consommation annuelle du Raspberry et description des mesures : 01/06/2024 - 10/06/2024
- Configuration du Raspberry pour un sous-réseau spécifique : 01/06/2024 - 10/06/2024
- Fiches d'anomalies : 01/06/2024 - 10/06/2024
- Rédaction du REX équipe et individuel : 05/06/2024 - 10/06/2024
- Ajout des acronymes utilisés : 05/06/2024 - 10/06/2024
- Analyse des résultats : 05/06/2024 - 10/06/2024

Ajustements en Cours de Route :

Comme pour tout projet, des ajustements ont été nécessaires au fur et à mesure de l'avancement :

- Réallocation des tâches : Suite au départ de Timothée, il a été nécessaire de redistribuer ses tâches parmi les membres restants.
- Révisions des délais : Certains délais ont dû être ajustés en raison de la charge de travail supplémentaire et des imprévus techniques, comme la défaillance du Raspberry Pi.
- Adaptation des priorités : Les priorités ont été réévaluées périodiquement pour s'assurer que les tâches critiques étaient traitées en premier.

2.5. Diagramme de Gantt

Structure du Diagramme :

Le diagramme de Gantt est divisé en plusieurs phases principales, chacune contenant des tâches spécifiques assignées à différents membres de l'équipe. La durée de chaque tâche, la date de début, la date de fin et le pourcentage d'avancement sont indiqués pour chaque tâche.

Légende des membres de l'équipe :

- A : Bastien
- B : Morgan
- C : Jabrail

Visualisation du diagramme de Gantt :

Tableau 10 : Diagramme de GANTT

Numéro	Type	Titre de la Tâche	Réalisateur Tâche	Date Début	Durée (jour)	Date Fin (calculée)	Avancement Tâche (en %)
T 1.0	P, R	Phase 1		06/02/24	50	28/03/24	100%
T 1.1	P	Configurer le Raspberry	A	06/02/24	10	15/02/24	100%
T 1.2	P	Mise à jour du rapport	A	06/02/24	30	07/03/24	100%
T 1.3	P	Statut des risques	A	06/02/24	30	07/03/24	100%
T 1.4	P	Mise à jour de la BDD	C	06/02/24	30	07/03/24	100%
T 1.5	P	Écriture de la BDD en SQL	C	06/02/24	30	07/03/24	100%
T 1.6	P	Faire une liste des scénarios de tests (Plan de Validation)	C	06/02/24	50	28/03/24	100%
T 1.7	P	Description de l'installation du Raspberry	A	06/02/24	30	07/03/24	100%
T 1.8	P	Configurer le Raspberry sur une adresse IP particulière	B	15/02/24	20	07/03/24	100%
T 1.9.1		Matrice de traçabilité	C	06/02/24	50	28/03/24	100%
T 1.9.2	P	Procédures de test	C	06/02/24	50	28/03/24	100%
T 2.0	P, R	Phase 2		10/03/24	92	30/05/24	100%
T 2.1	P	Interface WEB, Appli WEB	B	10/03/24	82	30/05/24	100%
T 2.2	P	Programme informatique : commentaires, nom des variables et des fonctions en anglais	A	10/03/24	82	30/05/24	100%
T 2.3	P	Tests codes	C	01/06/24	10	10/06/24	100%
T 2.4	P	Tests rapports	C	01/06/24	10	10/06/24	100%

Numéro	Type	Titre de la Tâche	Réalisateur Tâche	Date Début	Durée (jour)	Date Fin (calculée)	Avancement Tâche (en %)
T 2.5	P	Gérer le GitHub	A	10/03/24	92	10/06/24	100%
T 3.0	P, R	Phase 3		01/06/24	12	12/06/24	100%
T 3.1	P	Consommation annuelle du Raspberry et description des mesures	B	01/06/24	12	12/06/24	100%
T 3.2	P	Configuration du Raspberry pour un sous-réseau spécifique	B	01/06/24	12	12/06/24	100%
T 3.3	P	Fiches d'anomalies	C	01/06/24	12	12/06/24	100%
T 3.4	P	Rédaction du REX équipe et individuel	A	05/06/24	7	12/06/24	100%
T 3.5	P	Ajout des acronymes utilisés	A	05/06/24	7	12/06/24	100%
T 3.6	P	Analyse des résultats	A, B, C	05/06/24	7	12/06/24	100%

3. Raspberry

3.1. Installation et Configuration

3.1.1. Description de l'installation

Préparation du Matériel :

Pour installer et configurer le Raspberry Pi 3, les éléments suivants sont nécessaires :

- Un Raspberry Pi 3
- Une carte microSD de 16 Go ou plus
- Un ordinateur pour télécharger et préparer le système d'exploitation
- Un lecteur de carte microSD pour l'ordinateur
- Un écran connecté via HDMI
- Un clavier et une souris USB
- Une alimentation pour le Raspberry Pi

Téléchargement et Installation du Système d'Exploitation :

- Depuis un ordinateur, allez sur le site officiel de Raspberry Pi (<https://www.raspberrypi.org/software/>).
- Téléchargez et installez le Raspberry Pi Imager.

Préparer la Carte microSD :

- Insérez la carte microSD dans le lecteur de carte de l'ordinateur.
- Lancez le Raspberry Pi Imager.
- Choisissez le système d'exploitation à installer : Raspberry Pi OS (32-bit).
- Sélectionnez la carte microSD comme cible.
- Cliquez sur l'icône de la roue dentée pour ouvrir les paramètres avancés.
- Configurez les paramètres locaux : cochez "Set locale settings", choisissez "France" en disposition du clavier, puis cliquez sur "Save".
- Cliquez sur "Write" pour écrire l'image sur la carte microSD.

Insérer la Carte microSD dans le Raspberry Pi :

- Une fois l'image écrite, éjectez la carte microSD de l'ordinateur.
- Insérez la carte microSD dans le Raspberry Pi.

Connecter les Périphériques :

- Connectez l'écran au Raspberry Pi via le port HDMI.
- Branchez le clavier et la souris USB.
- Connectez l'alimentation pour démarrer le Raspberry Pi.

3.1.2. Configuration du Raspberry (système + applications + configuration)

Modifier les Paramètres par Défaut :

- Allez dans Menu (icône du Raspberry Pi) > Préférences > Raspberry Pi Configuration.
- Changez le mot de passe par défaut dans l'onglet System.
- Allez dans l'onglet Localisation et configurez les paramètres régionaux :
 - Country : France
 - Timezone : Paris
 - Keyboard : Français
 - Wi-Fi Country : France

Connecter le Raspberry Pi à Internet :

- Cliquez sur l'icône Wi-Fi dans la barre des tâches.
- Sélectionnez le réseau Wi-Fi souhaité et entrez le mot de passe.
- Vérifiez que le Raspberry Pi est connecté à Internet.

Mise à Jour du Système :

- Sur l'interface graphique, ouvrez le terminal pour exécuter les commandes nécessaires.
- Exécutez les commandes suivantes pour mettre à jour le système :
 - `sudo apt update`
 - `sudo apt upgrade`
- Redémarrez le Raspberry Pi après la mise à jour.

Installation des Applications Nécessaires :

- Sur l'interface graphique, ouvrez le terminal pour exécuter les commandes nécessaires.
- Exécutez la commande suivante pour installer Apache :
 - `sudo apt install apache2 -y`
- Pour vérifier que le serveur web fonctionne, ouvrez un navigateur et tapez l'adresse IP du Raspberry Pi. Vous devriez voir la page par défaut d'Apache.

Installer PHP :

- `sudo apt install php libapache2-mod-php -y`

Installer MariaDB et PHP-MYSQL :

- `sudo apt install mariadb-server php-mysql -y`

Redémarrer le serveur Apache pour appliquer les changements :

- `sudo service apache2 restart`

Lancer le monitor MariaDB avec la commande suivante :

- `sudo mysql --user=root`
- exécutez les commandes suivantes pour configurer l'utilisateur root :
 - `DROP USER 'root'@'localhost';`
 - `CREATE USER 'root'@'localhost' IDENTIFIED BY 'root';`
 - `GRANT ALL PRIVILEGES ON *.* TO 'root'@'localhost';`
 - `QUIT;`

Installer phpMyAdmin :

- `sudo apt install phpmyadmin -y`
- Lors de l'installation, sélectionnez apache2 comme serveur web.
- Pour la configuration de phpMyAdmin, sélectionnez NO car le package n'est pas indispensable.
- Retournez dans le navigateur et tapez l'adresse IP du Raspberry Pi suivie de `/phpmyadmin` pour accéder à phpMyAdmin.
- Connectez-vous avec root comme utilisateur et root comme mot de passe.

Importer la Base de Données :

- Ouvrez une autre fenêtre du navigateur et allez sur https://github.com/BastienLab/Projet_Thales11 pour récupérer le fichier SQL de la base de données.

Importer le Fichier SQL dans phpMyAdmin :

- Retournez sur phpMyAdmin, allez dans l'onglet Import.
- Cliquez sur Choisir un fichier et sélectionnez le fichier SQL de la base de données récupéré depuis GitHub.
- Cliquez sur Go pour importer le fichier.
- Vérifiez que la base de données thales11 a bien été importée.

Télécharger le Site Web :

- Retournez sur le GitHub et téléchargez le dossier Site_web_VF.
- Ouvrez le gestionnaire de fichiers, cliquez sur la racine /, puis naviguez vers var, www, html.
- Supprimez les fichiers existants dans ce répertoire.
- Insérez les fichiers et dossiers du dossier Site_web_VF dans ce répertoire.
- Installer les Dépendances Python
- Installer les Packages Python Nécessaires :
 - `sudo pip install pandas xlswriter fpdf`
 - `sudo python3 -m pip install numpy`
 - `sudo apt-get install libopenblas-base`

3.2. Configuration de l'adresse IP

Pour configurer le Raspberry Pi 3 afin qu'il utilise une adresse IP spécifique dans un sous-réseau particulier, il faut suivre ces étapes :

Pré-requis :

- Raspberry Pi 3 avec Raspbian OS installé et interface graphique (GUI) activée.
- Accès au routeur ou aux paramètres réseau de l'environnement Thales Alenia Space.
- Adresse IP, masque de sous-réseau, passerelle et serveur DNS fournis par l'administrateur réseau.

Étapes de Configuration :

1.Connexion au Raspberry Pi :

- Connectez un écran et un clavier directement au Raspberry Pi et l'allumer.

2.Ouverture des Paramètres Réseau :

- Cliquez sur l'icône de réseau dans la barre des tâches (généralement en haut à droite).
- Sélectionnez "Wireless & Wired Network Settings" ou "Paramètres de Réseau Filaires et Wi-Fi" (selon la langue du système).

3.Configuration de l'Adresse IP Statique :

- Dans la fenêtre des paramètres réseau, sélectionnez l'interface "eth0" (pour Ethernet) ou "wlan0" (pour Wi-Fi) en fonction du type de connexion.
- Cliquez sur le bouton "IPv4 Settings" ou "Paramètres IPv4".

4.Modification des Paramètres IPv4 :

- Changez la "Méthode" de "Automatique (DHCP)" à "Manuel".
- Cliquez sur le bouton "Ajouter" pour ajouter une nouvelle adresse IP statique.

5.Saisie des Paramètres Réseau :

- Adresse : Entrez l'adresse IP fournie, par exemple 10.x.x.x.
- Masque de sous-réseau : Entrez le masque de sous-réseau, par exemple 255.255.255.0.
- Passerelle : Entrez l'adresse de la passerelle, par exemple 10.x.x.1.
- Serveurs DNS : Entrez les adresses des serveurs DNS fournis, par exemple 8.8.8.8, 8.8.4.4.

6.Application des Changements :

- Cliquez sur "Enregistrer" pour appliquer les changements.
- Fermez la fenêtre des paramètres réseau.

7.Redémarrage de la Connexion Réseau :

- Cliquez de nouveau sur l'icône de réseau et déconnectez-vous puis reconnectez-vous au réseau pour que les nouveaux paramètres prennent effet.
- Alternativement, redémarrez le Raspberry Pi pour vous assurer que les paramètres sont appliqués.

8.Vérification de la Configuration Réseau :

- Ouvrez un terminal sur le Raspberry Pi.
- Tapez la commande suivante pour vérifier que l'adresse IP a bien été assignée :
 - `ip addr show eth0` (Vous devriez voir l'adresse IP spécifiée sous inet).

9.Test de Connectivité :

- Testez la connectivité en pingant la passerelle et un serveur DNS :
 - `ping -c 4 10.x.x.1`
 - `ping -c 4 8.8.8.8`

10. Remplacement des 'x' dans les Adresses IP :

- Adresse IP (10.x.x.x) : Utilisez l'adresse IP spécifique assignée à votre Raspberry Pi par l'administrateur réseau.
- Passerelle (10.x.x.1) : Utilisez l'adresse de la passerelle réseau fournie par l'administrateur réseau.

3.3. Consommation du Raspberry (annuelle)

3.3.1. Calculs théoriques et hypothèses

Initialement, nous avons utilisé la commande `vcgencmd measure_volts core` pour mesurer la tension du cœur du Raspberry Pi, qui nous a donné une valeur de 1.3125V. Pour estimer la consommation énergétique, nous avons dû prendre en compte la résistance interne, qui n'est généralement pas spécifiée par le fabricant.

En utilisant une hypothèse de résistance interne de 0.005 ohms, nous avons initialement calculé la consommation comme suit :

- $P = V^2 / R = (1.3125)^2 / 0.005 \approx 344.06$ watts

Cependant, cette valeur s'est avérée irréaliste car une résistance interne de 0.005 ohms est extrêmement faible pour ce type de matériel.

Nous avons ensuite ajusté la résistance interne à une valeur plus plausible, comme 0.1 ohms :

- $P = V^2 / R = (1.3125)^2 / 0.1 \approx 17.22$ watts

Cette valeur était également trop élevée pour la consommation typique d'un Raspberry Pi 3. Nous avons donc décidé de simplifier notre approche en utilisant des valeurs de consommation de base réalistes compris entre 1.5 W et 5W (en se basant sur ce qui est trouvable sur internet).

```
import os
import time

def get_cpu_usage():
    with open('/proc/stat', 'r') as f:
        lines = f.readlines()
    for line in lines:
        if 'cpu ' in line:
            cpu_data = line.split()
            total_time = sum(int(i) for i in cpu_data[1:])
            idle_time = int(cpu_data[4])
            return total_time, idle_time
    return 0, 0

def get_temp():
    temp = os.popen("vcgencmd measure_temp").readline()
    return float(temp.replace("temp=", "").replace("\n", ""))

def get_volts():
    volts = os.popen("vcgencmd measure_volts").readline()
    return float(volts.replace("volt=", "").replace("\n", ""))

def estimate_power_usage(cpu_usage):
    # Consommation de base (estimée) en watts
    base_power = 1.5 # W, consommation minimale au repos
    # Consommation additionnelle par % d'utilisation du CPU
    max_power = 5 # W, consommation maximale à pleine charge
    additional_power_perc = (max_power - base_power) / 100 # W par % d'utilisation du CPU

    power_usage = base_power + (cpu_usage * additional_power_perc)
    return power_usage

prev_total_time, prev_idle_time = get_cpu_usage()

while True:
    time.sleep(1)
    total_time, idle_time = get_cpu_usage()
    delta_total_time = total_time - prev_total_time
    delta_idle_time = idle_time - prev_idle_time

    cpu_usage = (1 - (delta_idle_time / delta_total_time)) * 100
    temp = get_temp()
    volts = get_volts()
    power_usage = estimate_power_usage(cpu_usage)

    print(f"CPU Usage: {cpu_usage:.2f}%, Temperature: {temp:.2f}C, Voltage: {volts:.4f}V, Estimated Power Usage: {power_usage:.2f}W")

    prev_total_time = total_time
    prev_idle_time = idle_time
```

Figure 2 : Exemple de script python servant à mesurer la consommation d'un Raspberry

3.3.2. Consommation mesurée (réelle) durant l'utilisation du site

Pour mesurer la consommation réelle du Raspberry Pi 3 durant l'utilisation du site, nous avons utilisé ce script Python pour estimer la consommation d'énergie basée sur l'utilisation du CPU. Le script a été exécuté dans deux scénarios principaux :

Projet Checklist

Lancement du site :

```
rt@raspberrypi:~/Downloads $ python3 test-script-conso.py
CPU Usage: 0.25%, Temperature: 49.40C, Voltage: 1.3062V, Estimated Power Usage: 1.70W
CPU Usage: 2.57%, Temperature: 49.90C, Voltage: 1.3062V, Estimated Power Usage: 1.75W
CPU Usage: 2.17%, Temperature: 49.90C, Voltage: 1.3062V, Estimated Power Usage: 1.74W
CPU Usage: 1.70%, Temperature: 49.40C, Voltage: 1.3062V, Estimated Power Usage: 1.73W
CPU Usage: 6.57%, Temperature: 49.40C, Voltage: 1.3062V, Estimated Power Usage: 1.82W
CPU Usage: 9.85%, Temperature: 49.40C, Voltage: 1.3062V, Estimated Power Usage: 1.89W
CPU Usage: 2.43%, Temperature: 49.40C, Voltage: 1.3062V, Estimated Power Usage: 1.75W
CPU Usage: 5.56%, Temperature: 49.40C, Voltage: 1.3062V, Estimated Power Usage: 1.81W
CPU Usage: 11.82%, Temperature: 50.50C, Voltage: 1.3062V, Estimated Power Usage: 1.92W
CPU Usage: 12.99%, Temperature: 49.90C, Voltage: 1.3062V, Estimated Power Usage: 1.95W
CPU Usage: 29.80%, Temperature: 51.50C, Voltage: 1.3062V, Estimated Power Usage: 2.27W
CPU Usage: 27.91%, Temperature: 50.50C, Voltage: 1.3062V, Estimated Power Usage: 2.23W
CPU Usage: 32.84%, Temperature: 51.00C, Voltage: 1.3062V, Estimated Power Usage: 2.32W
CPU Usage: 36.41%, Temperature: 50.50C, Voltage: 1.3062V, Estimated Power Usage: 2.39W
CPU Usage: 34.48%, Temperature: 51.50C, Voltage: 1.3062V, Estimated Power Usage: 2.35W
CPU Usage: 53.94%, Temperature: 51.50C, Voltage: 1.3062V, Estimated Power Usage: 2.72W
CPU Usage: 68.20%, Temperature: 52.60C, Voltage: 1.3062V, Estimated Power Usage: 3.00W
CPU Usage: 72.32%, Temperature: 52.60C, Voltage: 1.3062V, Estimated Power Usage: 3.07W
CPU Usage: 50.61%, Temperature: 52.10C, Voltage: 1.3062V, Estimated Power Usage: 2.66W
CPU Usage: 45.34%, Temperature: 51.50C, Voltage: 1.3062V, Estimated Power Usage: 2.56W
CPU Usage: 36.08%, Temperature: 52.10C, Voltage: 1.3062V, Estimated Power Usage: 2.39W
CPU Usage: 72.86%, Temperature: 52.60C, Voltage: 1.3062V, Estimated Power Usage: 3.08W
CPU Usage: 88.75%, Temperature: 53.70C, Voltage: 1.3062V, Estimated Power Usage: 3.39W
CPU Usage: 81.46%, Temperature: 53.70C, Voltage: 1.3062V, Estimated Power Usage: 3.25W
CPU Usage: 86.68%, Temperature: 54.80C, Voltage: 1.3062V, Estimated Power Usage: 3.35W
CPU Usage: 90.19%, Temperature: 55.80C, Voltage: 1.3062V, Estimated Power Usage: 3.41W
CPU Usage: 93.75%, Temperature: 56.90C, Voltage: 1.3062V, Estimated Power Usage: 3.48W
CPU Usage: 80.84%, Temperature: 57.50C, Voltage: 1.3062V, Estimated Power Usage: 3.24W
CPU Usage: 68.61%, Temperature: 56.90C, Voltage: 1.3062V, Estimated Power Usage: 3.00W
CPU Usage: 27.14%, Temperature: 56.40C, Voltage: 1.3062V, Estimated Power Usage: 2.22W
CPU Usage: 36.27%, Temperature: 56.40C, Voltage: 1.3062V, Estimated Power Usage: 2.39W
CPU Usage: 63.30%, Temperature: 56.90C, Voltage: 1.3062V, Estimated Power Usage: 2.90W
CPU Usage: 31.19%, Temperature: 56.40C, Voltage: 1.3062V, Estimated Power Usage: 2.29W
CPU Usage: 33.82%, Temperature: 56.90C, Voltage: 1.3062V, Estimated Power Usage: 2.34W
CPU Usage: 60.25%, Temperature: 58.00C, Voltage: 1.3062V, Estimated Power Usage: 2.84W
CPU Usage: 82.85%, Temperature: 59.10C, Voltage: 1.3062V, Estimated Power Usage: 3.27W
CPU Usage: 77.09%, Temperature: 58.50C, Voltage: 1.3062V, Estimated Power Usage: 3.16W
CPU Usage: 78.70%, Temperature: 58.00C, Voltage: 1.3062V, Estimated Power Usage: 3.20W
CPU Usage: 85.19%, Temperature: 56.40C, Voltage: 1.3062V, Estimated Power Usage: 3.32W
```

Figure 3 : Photo du résultat du script lors du lancement du site

Usage CPU : Varie entre 0.25% et 93.19%

Température : Varie entre 49.40°C et 66.10°C

Tension : Constante à 1.3062V

Consommation estimée : Entre 1.70W et 3.41W

Lancement de l'exportation des bonnes pratiques :

CPU Usage: 5.08%, Temperature: 52.60C, Voltage: 1.3062V, Estimated Power Usage: 1.80W
CPU Usage: 3.69%, Temperature: 52.60C, Voltage: 1.3062V, Estimated Power Usage: 1.77W
CPU Usage: 10.76%, Temperature: 53.70C, Voltage: 1.3062V, Estimated Power Usage: 1.90W
CPU Usage: 42.30%, Temperature: 54.80C, Voltage: 1.3062V, Estimated Power Usage: 2.50W
CPU Usage: 80.60%, Temperature: 56.90C, Voltage: 1.3062V, Estimated Power Usage: 3.23W
CPU Usage: 69.78%, Temperature: 56.90C, Voltage: 1.3062V, Estimated Power Usage: 3.03W
CPU Usage: 88.81%, Temperature: 58.00C, Voltage: 1.3062V, Estimated Power Usage: 3.39W
CPU Usage: 67.49%, Temperature: 58.00C, Voltage: 1.3062V, Estimated Power Usage: 2.98W
CPU Usage: 92.35%, Temperature: 57.50C, Voltage: 1.3062V, Estimated Power Usage: 3.45W
CPU Usage: 81.91%, Temperature: 59.10C, Voltage: 1.3062V, Estimated Power Usage: 3.26W
CPU Usage: 74.13%, Temperature: 60.10C, Voltage: 1.3062V, Estimated Power Usage: 3.11W
CPU Usage: 88.81%, Temperature: 61.20C, Voltage: 1.3062V, Estimated Power Usage: 3.39W
CPU Usage: 78.21%, Temperature: 61.20C, Voltage: 1.3062V, Estimated Power Usage: 3.19W
CPU Usage: 85.71%, Temperature: 61.80C, Voltage: 1.3062V, Estimated Power Usage: 3.33W
CPU Usage: 82.90%, Temperature: 62.30C, Voltage: 1.3062V, Estimated Power Usage: 3.28W
CPU Usage: 93.76%, Temperature: 63.40C, Voltage: 1.3062V, Estimated Power Usage: 3.48W
CPU Usage: 82.85%, Temperature: 63.40C, Voltage: 1.3062V, Estimated Power Usage: 3.27W
CPU Usage: 79.04%, Temperature: 63.90C, Voltage: 1.3062V, Estimated Power Usage: 3.20W
CPU Usage: 78.25%, Temperature: 63.40C, Voltage: 1.3062V, Estimated Power Usage: 3.19W
CPU Usage: 89.13%, Temperature: 63.40C, Voltage: 1.3062V, Estimated Power Usage: 3.39W
CPU Usage: 85.96%, Temperature: 64.50C, Voltage: 1.3062V, Estimated Power Usage: 3.33W
CPU Usage: 95.31%, Temperature: 65.00C, Voltage: 1.3062V, Estimated Power Usage: 3.51W
CPU Usage: 93.88%, Temperature: 65.50C, Voltage: 1.3062V, Estimated Power Usage: 3.48W
CPU Usage: 88.28%, Temperature: 65.50C, Voltage: 1.3062V, Estimated Power Usage: 3.38W
CPU Usage: 92.47%, Temperature: 65.50C, Voltage: 1.3062V, Estimated Power Usage: 3.46W
CPU Usage: 94.55%, Temperature: 66.60C, Voltage: 1.3062V, Estimated Power Usage: 3.50W
CPU Usage: 96.54%, Temperature: 66.10C, Voltage: 1.3062V, Estimated Power Usage: 3.53W
CPU Usage: 90.28%, Temperature: 66.10C, Voltage: 1.3062V, Estimated Power Usage: 3.42W
CPU Usage: 93.00%, Temperature: 66.10C, Voltage: 1.3062V, Estimated Power Usage: 3.47W
CPU Usage: 61.25%, Temperature: 66.10C, Voltage: 1.3062V, Estimated Power Usage: 2.86W
CPU Usage: 74.01%, Temperature: 66.10C, Voltage: 1.3062V, Estimated Power Usage: 3.11W
CPU Usage: 68.43%, Temperature: 65.50C, Voltage: 1.3062V, Estimated Power Usage: 3.00W
CPU Usage: 51.96%, Temperature: 66.10C, Voltage: 1.3062V, Estimated Power Usage: 2.69W
CPU Usage: 45.28%, Temperature: 65.00C, Voltage: 1.3062V, Estimated Power Usage: 2.56W
CPU Usage: 34.53%, Temperature: 64.50C, Voltage: 1.3062V, Estimated Power Usage: 2.36W
CPU Usage: 6.19%, Temperature: 62.30C, Voltage: 1.3062V, Estimated Power Usage: 1.82W
CPU Usage: 9.29%, Temperature: 61.80C, Voltage: 1.3062V, Estimated Power Usage: 1.80W
CPU Usage: 5.57%, Temperature: 52.60C, Voltage: 1.3062V, Estimated Power Usage: 1.77W

Figure 4 : Photo du résultat du script lors de l'exportation des bonnes pratiques

Usage CPU : Varie entre 5.08% et 92.47%

Température : Varie entre 52.60°C et 66.20°C

Tension : Constante à 1.3062V

Consommation estimée : Entre 1.77W et 3.51W

Ces mesures montrent que la consommation du Raspberry Pi 3 reste majoritairement dans la plage de 1.5W à 5 W, en fonction de la charge du CPU et des activités en cours. Ces résultats confirment les estimations théoriques discutées précédemment.

3.3.3. Indication de la consommation annuelle prévue/impact environnemental

Estimation annuelle maximum :

Pour estimer la consommation annuelle du Raspberry Pi, nous considérons une utilisation continue 24 heures sur 24, 7 jours sur 7. En prenant une consommation moyenne estimée de 2.65W (comme moyenne entre les valeurs mesurées), nous calculons comme suit :

Consommation journalière = $2.65W \times 24h = 63.6Wh$

Consommation annuelle = $63.6Wh \times 365 \text{ jours} = 23.214kWh$

Ainsi, la consommation annuelle maximum prévue pour le Raspberry Pi 3 est de **23.214 kWh**.

Estimation annuelle réaliste :

Pour une estimation plus réaliste, nous considérons une utilisation de 5 heures par jour, 5 jours par semaine. Le calcul de la consommation devient :

Consommation hebdomadaire = $2.65W \times 5h/\text{jour} \times 5 \text{ jours/semaine} = 66.25Wh/\text{semaine}$

Consommation annuelle = $66.25Wh/\text{semaine} \times 52 \text{ semaines/an} = 3.445 kWh$

Ainsi, la consommation annuelle estimée pour le Raspberry Pi 3, en considérant une utilisation de 5 heures par jour, 5 jours par semaine, est de **3.445 kWh**.

Impacts Environnemental :

Nous allons calculer l'impact environnemental de l'utilisation du Raspberry PI 3 en termes d'émissions de CO₂. Sachant qu'en France depuis Juin 2024 l'intensité carbone en gCO₂eq/KWh était de 18g selon le site app.electricitymaps.com .

Utilisation annuelle permanent = $23,21 \times 18 = 417,78g$

- 23,21KWh équivaut à **417,78g** d'émissions de CO₂

Dans un cas plus réaliste avec 5h d'utilisations par jour = $3.44 \times 18 = 61.92g$

- 3.44 KWh équivaut à **61.92g** d'émissions de CO₂.

4. Programmation

4.1. PhpMyAdmin

Nous avons utilisé PhpMyAdmin dans le cadre de ce projet pour gérer la base de données de manière efficace et intuitive. Grâce aux différents cours sur SQL et nos recherches personnelles, nous avons pu créer et administrer notre base de données sans difficulté majeure. PhpMyAdmin s'est révélé être un outil essentiel, simplifiant grandement les tâches de gestion de la base de données, telles que la création de tables, l'insertion de données et la gestion des utilisateurs.

Table	Action	Lignes	Type	Interclassement	Taille	Perte
<input type="checkbox"/> appartenance	★ Parcourir Structure Rechercher Insérer Vider Supprimer	10	MylSAM	utf8mb4_0900_ai_ci	5,2 kio	72 o
<input type="checkbox"/> association	★ Parcourir Structure Rechercher Insérer Vider Supprimer	3	MylSAM	utf8mb4_0900_ai_ci	4,0 kio	-
<input type="checkbox"/> bonnespratiques	★ Parcourir Structure Rechercher Insérer Vider Supprimer	7	MylSAM	utf8mb4_0900_ai_ci	2,3 kio	20 o
<input type="checkbox"/> exigencesmdp	★ Parcourir Structure Rechercher Insérer Vider Supprimer	0	MylSAM	utf8mb4_0900_ai_ci	1,0 kio	-
<input type="checkbox"/> logs	★ Parcourir Structure Rechercher Insérer Vider Supprimer	340	MylSAM	utf8mb4_0900_ai_ci	24,3 kio	-
<input type="checkbox"/> motscles	★ Parcourir Structure Rechercher Insérer Vider Supprimer	2	MylSAM	utf8mb4_0900_ai_ci	2,0 kio	-
<input type="checkbox"/> phases	★ Parcourir Structure Rechercher Insérer Vider Supprimer	2	MylSAM	utf8mb4_0900_ai_ci	2,0 kio	-
<input type="checkbox"/> programmes	★ Parcourir Structure Rechercher Insérer Vider Supprimer	6	MylSAM	utf8mb4_0900_ai_ci	2,1 kio	-
<input type="checkbox"/> utilisateurs	★ Parcourir Structure Rechercher Insérer Vider Supprimer	3	MylSAM	utf8mb4_0900_ai_ci	2,4 kio	-
9 tables	Somme	373	MylSAM	utf8mb4_0900_ai_ci	45,5 kio	92 o

Figure 5 : Gestion des tables PhpMyAdmin

L'utilisation de PhpMyAdmin nous a permis de valider facilement les requêtes SQL écrites dans nos fichiers PHP. En effet, l'interface de requête SQL de PhpMyAdmin a été un atout précieux pour tester et déboguer les requêtes avant de les intégrer dans notre code. Cette approche nous a assuré que les opérations sur la base de données seraient exécutées correctement par notre application, réduisant ainsi les erreurs potentielles et augmentant l'efficacité de notre développement.

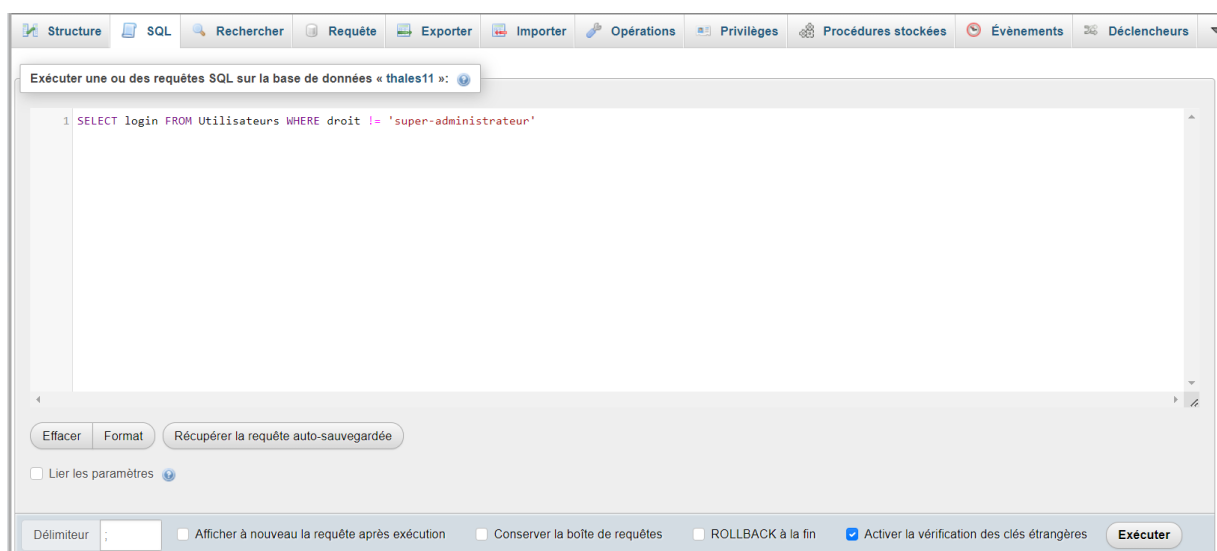


Figure 6 : Exemple de test de requête via PhpMyAdmin

En résumé, PhpMyAdmin a été indispensable pour gérer notre base de données tout au long du projet. Il nous a permis de centraliser la gestion des données et de garantir l'intégrité et la cohérence de notre base de données.

4.2. Python

Dans le cadre de ce projet, nous avons utilisé Python principalement pour l'exportation des données en fichiers PDF et Excel. L'expérience acquise en Python grâce aux cours et aux exercices pratiques nous a permis de manipuler les données de manière efficace et de générer des fichiers structurés pour les utilisateurs.

Les bibliothèques Python telles que pandas, xlsxwriter, et fpdf ont été particulièrement utiles. pandas nous a permis de structurer et de manipuler les données de manière flexible, tandis que xlsxwriter et fpdf ont facilité la création de documents Excel et PDF. Cette automatisation a permis d'accélérer le processus de génération de rapports.

```
def export_to_pdf(bp_details, user_id):
    """Export best practice details to a PDF file."""
    file_path = "bp_details.pdf"
    if os.path.exists(file_path):
        os.remove(file_path)

    pdf = FPDF()
    pdf.add_page()

    current_date = datetime.now().strftime("%d/%m/%Y")
    pdf.set_font("Arial", size=12)
    pdf.cell(0, 10, txt=f"Date: {current_date}", ln=1, align="L")
    pdf.cell(0, 10, txt=f"ID: {user_id}", ln=1, align="L")

    # Column titles
    pdf.set_font("Arial", 'B', 12)
    pdf.cell(40, 10, 'Programme', 1)
    pdf.cell(40, 10, 'Phase', 1)
    pdf.cell(80, 10, 'Item', 1)
    pdf.cell(40, 10, 'Mots clés', 1)
    pdf.cell(40, 10, 'Appliqué', 1)
    pdf.ln()

    # Data rows
    pdf.set_font("Arial", size=12)
    for bp in bp_details:
        pdf.cell(40, 10, bp['Programme'], 1)
        pdf.cell(40, 10, bp['Phase'], 1)
        pdf.cell(80, 10, bp['Item'], 1)
        pdf.cell(40, 10, bp['Mots clés'], 1)
        pdf.cell(40, 10, bp['Appliqué'], 1)
        pdf.ln()

    pdf.output(file_path)
```

Figure 7 : Exemple de code Python utilisé pour générer un PDF

En résumé, Python nous a servi à :

- Récupérer les informations des bonnes pratiques (Bps) à partir des IDs exportés.
- Mettre en forme ces informations dans des fichiers PDF et Excel.

Cela a permis de répondre efficacement aux besoins des utilisateurs en leur fournissant des rapports détaillés et bien organisés.

4.3. HTML et CSS

HTML et CSS ont été utilisés pour développer l'interface utilisateur de notre application web, permettant aux utilisateurs de naviguer et d'interagir facilement avec les différentes fonctionnalités de notre système. Les connaissances acquises en HTML et CSS à travers nos cours ont été mises en pratique pour créer une interface propre, réactive et conviviale.

Le HTML a servi à structurer le contenu de nos pages web, définissant les éléments de base tels que les formulaires, les tableaux et les sections de contenu. Le CSS, quant à lui, a été utilisé pour styliser ces éléments, assurant une présentation visuelle cohérente et attrayante.

```
<div id="mainContainer">
  <div id="bpContainer">
    <div id="bpList">Please select criteria to display best practices.</div>
  </div>
  <div id="logsContainer" onclick="redirectToLogsPage()">
    <div id="logsList"></div>
  </div>
</div>
<form action="exportData.php" method="post">
  <input type="hidden" name="action" value="export">
  <input type="hidden" name="bpIds" id="exportBpIds"> <!-- IDs will be injected by JS -->
  <select name="format">
    <option value="excel">Excel</option>
    <option value="pdf">PDF</option>
  </select>
  <button type="submit">Export</button>
  <button type="button" onclick="redirectToDuplicationPage()">Duplicate Selected BPs</button>
  <button type="button" onclick="redirectToDeactivatePage()">Deactivate Selected BPs</button>
  <button type="button" onclick="redirectToDeletePage()">Permanently Delete Selected BPs</button>
</form>
```

Figure 8 : Exemple de formulaire html

```
/* Body Styles */
body {
  font-family: Arial, sans-serif;
  margin: 0;
  padding: 0;
  background-color: #f4f4f4;
}

/* Header Styles */
header {
  background-color: #007bff;
  color: white;
  padding: 10px 0;
  width: 100%;
  display: flex;
  flex-wrap: wrap;
  justify-content: space-between;
  align-items: center;
  position: relative; /* Added for absolute positioning of the menu */
}

header .menu-burger {
  background: none;
  border: none;
  color: white;
  font-size: 24px;
  margin-left: 20px;
  cursor: pointer;
}
```

Figure 9 : Exemple de code CSS

En somme, l'utilisation de HTML et CSS a été cruciale pour le succès de notre projet. Ces technologies nous ont permis de créer une interface utilisateur attrayante et fonctionnelle, facilitant l'interaction des utilisateurs avec notre application.

4.4. PHP

PHP a été le langage principal utilisé pour le développement côté serveur de notre application. Il a permis de gérer la logique de l'application, de traiter les requêtes des utilisateurs et de communiquer avec la base de données. Les compétences en PHP acquises au cours du deuxième semestre ont été essentielles pour construire une application robuste et sécurisée.

Grâce à PHP, nous avons pu implémenter des fonctionnalités critiques telles que l'authentification des utilisateurs, la gestion des sessions et l'exécution de requêtes SQL. Par exemple, nous avons utilisé PHP pour gérer la suppression des bonnes pratiques, en veillant à ce que les opérations soient effectuées de manière sécurisée et efficace. Nous avons validé les données d'entrée et géré les interactions complexes entre les utilisateurs et le serveur.

```
<?php
// Start the session
session_start();
// Include database connection and logger functions
require 'db_connection.php';
include 'logger.php';

// Log the access to the page
logAction('deleteBPs.php', 'Page accessed');

// Check user access rights
if (!isset($_SESSION['login']) || ($_SESSION['droit'] != 'administrateur' && $_SESSION['droit'] != 'super-administrateur')) {
    $message = "<p class='error'>Access denied. You must be logged in as an administrator.</p>";
    logAction('deleteBPs.php', 'Access denied');
    exit;
}

$message = '';
$bpIds = $_GET['bpIds'] ?? '';

// Handle form submission
if ($_SERVER['REQUEST_METHOD'] === 'POST' && isset($_POST['confirm'])) {
    if ($_POST['confirm'] === 'yes') {
        // Convert the BP IDs string to an array
        $bpIdsArray = explode(',', $bpIds);
        $placeholders = implode(',', array_fill(0, count($bpIdsArray), '?'));

        // Delete from Appartenance table
        $queryAppartenance = "DELETE FROM Appartenance WHERE numAppart IN ($placeholders)";
        $stmt = $conn->prepare($queryAppartenance);
        $stmt->bind_param(str_repeat('i', count($bpIdsArray)), ...$bpIdsArray);
        $stmt->execute();
        $stmt->close();

        // Log the successful deletion
        logAction('deleteBPs.php', "Appartenance entries deleted: " . implode(', ', $bpIdsArray));
    }
}
```

Figure 10 : Partie du Code PHP pour la Gestion de la Suppression des Bonnes Pratiques

En conclusion, PHP a joué un rôle central dans le développement de notre application. Il a permis de créer une architecture backend solide, garantissant la performance et la sécurité de notre système tout en offrant une expérience utilisateur fluide et intuitive.

4.5. Javascript

JavaScript a été utilisé pour améliorer l'interactivité et la réactivité de notre application web. Nous avons intégré des scripts JavaScript pour gérer les événements utilisateur, charger dynamiquement les données et manipuler le contenu de la page en temps réel.

Par exemple, nous avons utilisé JavaScript pour charger les options de filtres et les bonnes pratiques à partir du serveur, en fonction des sélections de l'utilisateur. Cela a permis de rendre l'application plus réactive et intuitive. De plus, nous avons utilisé JavaScript pour afficher les journaux des activités des utilisateurs, offrant ainsi une meilleure traçabilité des actions.

```
// Function to load filter options from the server
function loadFilterOptions() {
  fetch('loadFilters.php')
    .then(response => response.json())
    .then(data => {
      // Clear current options
      phaseFilter.innerHTML = '<option value="">All Phases</option>';
      programFilter.innerHTML = '<option value="">All Programs</option>';

      // Use a Set to ensure uniqueness
      const phaseSet = new Set();
      const programSet = new Set();

      // Populate phase filter options
      data.phases.forEach(phase => {
        if (!phaseSet.has(phase.numPhases)) {
          phaseFilter.innerHTML += `<option value="${phase.numPhases}">${phase.nomPhase}</option>`;
          phaseSet.add(phase.numPhases);
        }
      });

      // Populate program filter options
      data.programs.forEach(program => {
        if (!programSet.has(program.numProg)) {
          programFilter.innerHTML += `<option value="${program.numProg}">${program.nomProgramme}</option>`;
          programSet.add(program.numProg);
        }
      });
    })
    .catch(error => {
      console.error('Error loading filter options:', error);
    });
}
```

Figure 11 : Fonction Javascript qui charge les options du filtre

En somme, JavaScript a enrichi notre application web en ajoutant des fonctionnalités dynamiques et interactives. Il a permis de créer une interface utilisateur plus réactive et conviviale, répondant aux attentes modernes des utilisateurs.

4.6. Échange d'Informations entre les Pages et Sécurité du Site

Échange d'Informations entre les Pages :

Sessions PHP :

Pour échanger des informations entre les différentes pages de notre application web, nous avons principalement utilisé les sessions PHP. Les sessions nous permettent de stocker et de récupérer des informations utilisateur de manière sécurisée et persistante à travers plusieurs pages.

```
<?php
// Start the session
session_start();
// Check if the user has a role set in the session
if (isset($_SESSION['droit'])) {
    // Display options available to all logged-in users
    echo '<li><a href="enregistrerbp.html">Créer une BP</a></li>';
    echo '<li><a href="view_logs.php">Logs</a></li>';

    // Display options for administrators and super-administrators
    if ($_SESSION['droit'] == 'administrateur' || $_SESSION['droit'] == 'super-administrateur') {
        echo '<li><a href="Page-modif-mdp-user.php">Modifier MDP Utilisateur</a></li>';
        echo '<li><a href="manageDeactivatedBPs.php">Gérer les Bonnes Pratiques Désactivées</a></li>';
    }

    // Display options only for super-administrators
    if ($_SESSION['droit'] == 'super-administrateur') {
        echo '<li><a href="deleteAccount.php">Supprimer un compte</a></li>';
        echo '<li><a href="modifyUserRole.php">Modifier les droits d\'un compte</a></li>';
        echo '<li><a href="createAccount.php">Créer un compte</a></li>';
    }
}
```

Figure 12 : Exemple d'utilisation de sessions PHP

Fichiers CSV :

Ensuite, pour ce qui est de l'exportation des données, nous avons choisi d'utiliser des fichiers CSV. On écrit donc les id des différentes bonnes pratiques choisies dans un fichier CSV, qui est ensuite traité par un script Python pour générer des rapports au format PDF ou Excel. Ce fichier est stocké dans un dossier avec des permissions `chmod 777` pour permettre sa suppression ou son écrasement lors des prochains exports.

```
// Define the path for the CSV file on the web server
$csvFile = '/var/www/html/export_data.csv';

// Open the CSV file for writing
$handle = fopen($csvFile, 'w');

// Add a header row to the CSV file
fputcsv($handle, ['numBP']);

// Write the data to the CSV file
foreach ($numBPs as $numBP) {
    fputcsv($handle, [$numBP]);
}

// Close the file handle
fclose($handle);

// Log the creation of the CSV file
logAction('exportBP.php', 'CSV file created: ' . $csvFile);

// Redirect to the confirmation page with the file path and format as query parameters
header("Location: confirmExport.php?file=" . urlencode($csvFile) . "&format=" . urlencode($format));

// Exit the script
exit;
?>
```

Figure 13 : Exemple de création d'un fichier CSV pour échanger des données

```
def main(format, csv_file):
    """Main function to execute the export process."""
    bp_ids = []
    # Read the list of BP IDs from the provided CSV file
    with open(csv_file, newline='') as csvfile:
        reader = csv.reader(csvfile)
        bp_ids = [row[0] for row in reader]

    # Fetch BP details and active user ID
    bp_details = fetch_bp_details(bp_ids)
    user_id = fetch_active_user()

    # Export data to the specified format
    if format == 'excel':
        export_to_excel(bp_details, user_id)
    elif format == 'pdf':
        export_to_pdf(bp_details, user_id)
    else:
        print(f"Unknown format: {format}")

if __name__ == "__main__":
    # Command-line arguments for format and CSV file
    format = sys.argv[1]
    csv_file = sys.argv[2]
    main(format, csv_file)
```

Figure 14 : Exemple d'ouverture d'un fichier CSV pour échanger des données

Vérification de la présence du fichier :

Pour garantir que les fichiers ne s'accumulent pas, nous supprimons les anciens fichiers avant de créer de nouveaux exports.

```
def export_to_pdf(bp_details, user_id):
    """Export best practice details to a PDF file."""
    file_path = "bp_details.pdf"
    if os.path.exists(file_path):
        os.remove(file_path)
```

Figure 15 : Exemple de suppression d'un fichier pour éviter les doublons

Sécurisation du Site :

Requêtes Préparées :

Pour prévenir les injections SQL, nous utilisons des requêtes préparées avec des placeholders « ? » pour les paramètres. Cela empêche les attaquants d'injecter du code malveillant dans les requêtes SQL.

```
// Create a new MySQLi connection
$conn = new mysqli($servername, $dbUsername, $dbPassword, $dbname);

// Check if the connection was successful
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// Prepare an SQL statement to insert the new user
$stmt = $conn->prepare("INSERT INTO Utilisateurs (login, mdp, droit, prenom, nom, bloque, tentative_login, statut) VALUES (?, ?, ?, ?, ?, 0, 0, 'inactif')");
// Bind parameters to the SQL statement
$stmt->bind_param("sssss", $username, $hashedPassword, $role, $prenom, $nom);
// Execute the SQL statement
if ($stmt->execute()) {
    // If successful, set a success message and log the account creation
    $errorMessage = "Account successfully created.";
    logAction('createAccount.php', 'Account created for user: ' . $username);
} else {
    // If there was an error, set an error message and log the failure
    $errorMessage = "Error creating account: " . $stmt->error;
    logAction('createAccount.php', 'Account creation error for user: ' . $username . ', Error: ' . $stmt->error);
}

// Close the statement and connection
$stmt->close();
$conn->close();
```

Figure 16 : Exemple de requête préparée avec des placeholders "?"

Vérification de la Connexion :

Nous vérifions systématiquement si un utilisateur est connecté avec les bons droits avant de lui permettre d'accéder aux pages sécurisées. Si l'utilisateur n'est pas connecté ou n'a pas le niveau de privilège nécessaire, il est redirigé vers la page de connexion avec un message d'erreur approprié.

```
// Check if the user has super-administrator rights
if (!isset($_SESSION['login']) || $_SESSION['droit'] != 'super-administrateur') {
    // Display an error message and log the access denial
    echo "<p class='error'>Access denied. You must be logged in as a super-administrator.</p>";
    logAction('createAccount.php', 'Access denied');
    exit;
}
```

Figure 17 : Exemple de vérification de connexion

Chiffrement des Mots de Passe :

Les mots de passe des utilisateurs sont stockés de manière sécurisée en utilisant la fonction hash de PHP, qui applique l'algorithme de hachage sha256.

```
$hashedPassword = hash('sha256', $password); // Hashage du mot de passe avec SHA-256
```

Figure 18 : Exemple de chiffrement de mot de passe

Blocage après Tentatives de Connexion Échouées :

Pour prévenir les attaques par force brute, nous avons mis en place un mécanisme de blocage des comptes après trois tentatives de connexion infructueuses.

```
else {
    // Verify the password
    if (password_verify($password, $row['mdp'])) {
        // Set session variables for successful login
        $_SESSION['login'] = $username;
        $_SESSION['droit'] = $row['droit'];
        // Update the user status to active and reset login attempts
        $updateStmt = $conn->prepare("UPDATE Utilisateurs SET tentative_login = 0, statut = 'actif' WHERE login = ?");
        $updateStmt->bind_param("s", $username);
        $updateStmt->execute();
        $updateStmt->close();
        logAction('login.php', 'Successful login for user: ' . $username);
        // Redirect to the index page
        header("Location: index.php");
        exit();
    }
}
```

Figure 19 : Exemple d'incrémement du nombre de tentative de connexion effectuée

Messages d'Erreur et Redirections :

En cas de tentative d'accès non autorisé, des messages d'erreur clairs sont affichés et l'utilisateur est redirigé vers la page appropriée. Cela empêche les utilisateurs non autorisés d'accéder aux pages sécurisées.

```
echo "<p class='error'>Access denied. You must be logged in as a super-administrator.</p>";
logAction('createAccount.php', 'Access denied');
exit;
```

Figure 20 : Exemple de message d'erreur et de redirection

5. Assurance Qualité Logicielle

5.1. Plan de validation

Création de Compte

Tableau 11 : Plan de validation sur la Création de compte

No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks
i	Entrer dans le site et créer un compte avec une adresse mail valide (modèle « x@yy.zz »).	Si cette adresse n'a pas pour modèle « x@yy.zz », elle ne sera pas considérée comme valide.	Pass	

No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks
ii	Vérifier que l'identifiant est unique, sans accent et fait entre 8 et 30 caractères composés uniquement de lettres et de chiffres.	Si les caractéristiques de l'identifiant ne sont pas respectées, il ne sera pas valide.	Fail	L'application échoue à vérifier l'unicité des identifiants.
iii	Vérifier que le mot de passe ne contient pas l'identifiant, ni d'accent, et respecte les exigences de caractères (au moins q lettres majuscules, p autres minuscules et r caractères spéciaux).	Si les caractéristiques du mot de passe ne sont pas respectées, il ne sera pas valide.	Pass	
iv	Configurer les paramètres n, p, q, r uniquement par un utilisateur de type « Administrateur », avec un mot de passe scripté et blocage après 3 tentatives échouées.	Si ces 4 paramètres ne sont pas configurables par un administrateur, il y a un problème de droits du compte.	Fail	L'application échoue à bloquer un compte après 3 tentatives de connexion infructueuses.
v	Vérifier qu'il est possible de se connecter avec uniquement l'identifiant et le mot de passe, et qu'il est possible de se déconnecter et changer de mot de passe lorsqu'il est oublié.	S'il est impossible de se connecter avec ces deux paramètres, il y a un problème dans le portail de connexion.	Pass	

Comptes Utilisateurs

Tableau 12 : Plan de validation sur les comptes utilisateurs

No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks
i	Vérifier qu'un simple utilisateur voit moins de fonctionnalités que les administrateurs, opérateurs et super administrateurs.	Si un simple utilisateur voit autant de fonctionnalités que les autres types d'utilisateurs, il y a un problème de droits.	Pass	
ii	Tester la page des bonnes pratiques pour voir, sélectionner par phases ou rechercher par mots-clés, et exclure programmes, phases ou mots-clés.	S'il est impossible de trier et exclure les bonnes pratiques, il y a un problème avec l'application CHECKLIST.	Pass	
iii	Vérifier que l'application CHECKLIST peut écrire, dupliquer ou supprimer des bonnes pratiques pour une phase donnée.	Si l'application CHECKLIST ne peut pas réaliser ces actions, il y a un problème avec l'application CHECKLIST.	Fail	L'application échoue à supprimer des bonnes pratiques pour un programme donné et une phase donnée.

Comptes Administrateur/Super-Administrateur

Tableau 13 : Plan de validation sur les comptes administrateurs et super-administrateurs

No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks
i	Vérifier que les administrateurs peuvent supprimer/ajouter des comptes opérateurs, contrôler les accès des utilisateurs, afficher la	Si les administrateurs ne peuvent pas réaliser toutes ces actions, il	Pass	

No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks
	liste des utilisateurs, retrouver un utilisateur par son identifiant, voir l'historique des interactions, modifier le mot de passe des comptes verrouillés, créer/supprimer des programmes et bonnes pratiques, et effectuer les actions d'un opérateur.	y a un problème avec les droits du compte.		
ii	Vérifier que le super-administrateur peut donner/retirer des droits aux administrateurs, changer leurs droits, supprimer des comptes administrateurs, ne pas être bloqué par l'application CHECKLIST et effectuer les actions d'un administrateur.	Si le super-administrateur ne peut pas réaliser toutes ces actions, il y a un problème avec les droits du compte.	Pass	

Assistance Technique

Tableau 14 : Plan de validation sur l'assistance technique

No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks
i	Vérifier que l'application CHECKLIST permet d'accéder à un guide d'utilisation.	Si l'application CHECKLIST ne peut pas réaliser ces actions, il y a un problème avec les redirections.	Pass	
ii	Vérifier que l'application CHECKLIST mémorise les événements (Informations, Warnings, Alarmes) et les actions des utilisateurs (date, utilisateur, profil, type d'événement, description de l'action).	Si l'application CHECKLIST ne peut pas mémoriser ces données, il y a un problème avec la base de données.	Pass	

Interface Web

Tableau 15 : Plan de validation sur l'interface web

No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks
i	Vérifier que l'interface web possède un onglet d'accueil avec une barre de recherche pour les mots-clés, un onglet d'assistance technique, un onglet programme, et qu'elle s'adapte aux différentes tailles d'écrans.	Si l'interface web n'affiche et n'exécute pas toutes ces actions, il y a un problème dans le CSS, PHP ou HTML.	Pass	
ii	Vérifier que lors de la suppression d'une bonne pratique, d'un compte ou d'un programme, un message de confirmation est affiché, que l'interface web ouvre une page d'authentification, que la page de connexion possède un bouton pour créer un compte, que la page de création de compte affiche une erreur si les conditions ne sont pas respectées, que les identifiants sont cryptés dans la base de données, et qu'il y a des boutons supplémentaires pour les administrateurs ou super-administrateurs.	Si l'interface web ne réalise pas toutes ces actions, il y a un problème dans le CSS, PHP ou HTML.	Pass	

5.2. Procédures de test

Objectif de la Section :

Cette section décrit le cadre général des procédures de test mises en place pour vérifier les différentes fonctionnalités de l'application CHECKLIST. Les détails spécifiques de chaque test individuel sont fournis en annexe pour éviter d'alourdir le rapport principal.

Description Générale :

Les procédures de test sont conçues pour garantir que l'application fonctionne correctement et répond aux exigences spécifiées. Chaque procédure de test inclut des objectifs clairs, des étapes détaillées, et des critères de réussite ou d'échec. Les tests couvrent les principales fonctionnalités telles que la création, la duplication et la suppression de bonnes pratiques, ainsi que la gestion des comptes utilisateurs et administrateurs.

Structure Générale des Procédures de Test :

Objectif des Tests : Vérifier le bon fonctionnement des différentes fonctionnalités du site.

Prérequis :

- Conditions initiales nécessaires avant l'exécution des tests.
- Environnement de test configuré.

Étapes de Test : Instructions détaillées pour l'exécution des tests.

Données d'entrée : Informations nécessaires pour réaliser les tests.

Résultats Attendus : Critères pour déterminer le succès ou l'échec des tests.

Organisation des Procédures de Test : Les détails complets de chaque procédure de test, y compris les étapes spécifiques, les données d'entrée et les résultats attendus, sont documentés en annexe.

5.3. Rapports de test et fiches d'anomalies

Objectif de la Section :

Cette section résume les résultats des tests effectués et décrit les anomalies détectées durant le processus de test. Les rapports de test détaillés et les fiches d'anomalies complètes sont inclus en annexe pour permettre une analyse approfondie sans surcharger le rapport principal.

Description Générale :

Les rapports de test fournissent une vue d'ensemble des activités de test réalisées, incluant le statut de chaque test (réussi ou échoué) et les remarques pertinentes. Les fiches d'anomalies documentent chaque problème rencontré, incluant une description détaillée, les étapes pour reproduire l'anomalie, et son impact sur le système.

Contenu des Rapports de Test et Fiches d'Anomalies :

Rapports de Test :

Résumé des Activités de Test : Liste des tests exécutés avec leur statut.

Observations : Remarques sur les résultats des tests.

Analyse des Résultats : Évaluation globale des performances de l'application.

Fiches d'Anomalies :

Identification de l'Anomalie : Numéro unique et description de l'anomalie.

Conditions de Découverte : Contexte et environnement du test.

Étapes de Reproduction : Instructions pour reproduire l'anomalie.

Impact : Effets de l'anomalie sur le système et les utilisateurs.

Statut de l'Anomalie : Suivi du statut (nouvelle, en cours, résolue, fermée).

Utilisation des Rapports et Fiches d'Anomalies : Ces documents sont utilisés pour suivre la progression des tests, identifier et prioriser les problèmes à corriger, et fournir des informations précises aux développeurs pour résoudre les anomalies.

Les détails complets des rapports de test et fiches d'anomalies sont disponibles en annexe.

6. Gestion de Configuration des Logiciels

6.1. Utilisation de GitHub

Pour assurer une gestion efficace du code source et permettre une collaboration fluide entre les membres de l'équipe, nous avons utilisé GitHub en conjonction avec Visual Studio Code (VSCode). Voici comment nous avons procédé :

Installation de Git :

Chaque membre de l'équipe a installé Git sur son environnement de développement local. Pour ce faire, les étapes suivantes ont été suivies :

- Téléchargement de Git depuis le site officiel : <https://git-scm.com/downloads>
- Installation de Git en suivant les instructions spécifiques au système d'exploitation utilisé (Windows, MacOS, Linux).

Configuration de Git :

Après l'installation, chaque membre a configuré Git avec ses informations personnelles :

- `git config --global user.name "Bastien Labeste"` (par exemple)
- `git config --global user.email "bastien.labeste@etu.unice.fr"`

Clonage du Dépôt GitHub :

Le dépôt GitHub du projet a été cloné sur les environnements locaux de chaque membre :

- `git clone https://github.com/BastienLab/Projet_Thales11.git`

Utilisation de Visual Studio Code (VSCode) :

Pour faciliter l'édition et la gestion du code, nous avons utilisé Visual Studio Code (VSCode). Chaque membre a lié le dossier contenant les fichiers du site web à VSCode en utilisant la commande suivante dans le terminal de VSCode :

- `cd / Users/Bastien/Desktop/BUT1 R&T/SAE15_thales-P2`
- `code .`

VSCode offre une intégration Git qui permet de gérer les commits, les branches et les pushes directement depuis l'interface.

Commit et Push des Changements via VSCode :

À chaque fois qu'un membre faisait des modifications, il les enregistrait (commit) avec un message explicatif et poussait (push) les changements sur le dépôt GitHub. Les étapes suivantes étaient suivies dans VSCode :

- Sélection des fichiers à commit : Via l'onglet "Source Control", les fichiers modifiés sont sélectionnés.
- Ajout d'un message de commit : Un message décrivant les modifications est ajouté.
- Commit et Push : Les changements sont commit et poussés vers le dépôt distant en cliquant sur les icônes correspondantes dans l'interface de VSCode.

Mise à Jour du Fichier README.md :

Nous avons maintenu à jour le fichier README.md pour fournir des informations détaillées sur le projet, les étapes de configuration, et les instructions d'utilisation. Ce fichier est essentiel pour quiconque souhaitant comprendre et utiliser notre projet.

6.2. Gestion des tags

Pour gérer les différentes versions du projet, nous avons utilisé le système de tags de Git. Cela nous a permis de marquer des points importants de l'avancement du projet et de créer des versions distinctes du code source. Voici comment nous avons procédé :

Création de Versions :

À chaque étape significative du développement, nous avons créé un dossier de version (par exemple, V0.5, V1.0, V1.5, VF). Ces versions représentaient des jalons clés dans le développement du projet.

Création de Tags :

Pour chaque version, nous avons créé un tag dans Git afin de marquer cette version dans le dépôt. Les tags nous ont permis de retrouver facilement une version spécifique du projet. Voici les commandes utilisées pour créer un tag :

- `git tag -a V0.5 -m "Version 0.5: Description des changements"`
- `git push origin V0.5`

Messages de Commit :

À chaque commit, nous avons ajouté un message explicatif décrivant les modifications apportées. Cela a permis aux autres membres de comprendre les changements effectués et de suivre l'évolution du projet de manière détaillée. Exemple de message de commit :

- `git commit -m "Ajout de la fonctionnalité X et correction du bug Y"`

Consultation des Tags :

Pour consulter les tags existants et leurs descriptions, les membres pouvaient utiliser la commande suivante :

- `git tag -n`

7. Annexes

7.1. Annexe A : Procédures de Test

Tableau 16 : Liste des procédures de test

Numéro de Vérification	Nom de la Procédure	Version de l'Issue	Date Début	Date Fin	Test Report Executed by	Test Report Check & Approved by	Résultat de la Vérification
1	Création de compte	0.5	2024-04-26	2024-04-30	Salhi Morgan	Labeste Bastien	Fail
2	Création de compte	0.5	2024-04-26	2024-04-30	Salhi Morgan	Labeste Bastien	Pass
3	Création de compte	0.5	2024-04-26	2024-04-30	Salhi Morgan	Labeste Bastien	Fail
4	Création de compte	0.5	2024-04-26	2024-04-30	Salhi Morgan	Labeste Bastien	Pass
5	Création de compte	0.5	2024-04-26	2024-04-30	Salhi Morgan	Labeste Bastien	Pass
6	Utilisateur	0.5	2024-04-26	2024-04-30	Salhi Morgan	Labeste Bastien	Pass
7	Utilisateur	0.5	2024-04-26	2024-04-30	Salhi Morgan	Labeste Bastien	Pass
8	Utilisateur	0.5	2024-04-26	2024-04-30	Salhi Morgan	Labeste Bastien	Fail
9	Administrateur	0.5	2024-04-26	2024-04-30	Salhi Morgan	Labeste Bastien	Pass
10	Administrateur	0.5	2024-04-26	2024-04-30	Salhi Morgan	Labeste Bastien	Pass
11	Administrateur	0.5	2024-04-26	2024-04-30	Salhi Morgan	Labeste Bastien	Pass
12	Création de compte	VF	2024-06-01	2024-06-01	Salhi Morgan	Labeste Bastien	Pass
13	Création de compte	VF	2024-06-01	2024-06-01	Salhi Morgan	Labeste Bastien	Pass

Projet Checklist

14	Utilisateur	VF	2024-06-01	2024-06-01	Salhi Morgan	Labeste Bastien	Pass
15	Super-administrateur	0.5	2024-04-26	2024-04-30	Salhi Morgan	Labeste Bastien	Fail
16	Super-administrateur	0.5	2024-04-26	2024-04-30	Salhi Morgan	Labeste Bastien	Pass
17	Super-administrateur	VF	2024-06-01	2024-06-07	Salhi Morgan	Labeste Bastien	Pass

7.2. Annexe B : Rapports de Test

Tableau 17 : Rapport de test 1

<Product> Verification / Validation Report				<product id>
Verification-Number: 1		Test Procedure : Création de compte Issue : version 0.5 Short Description : Test exigence - création de comptes Date : 2024-04-30 Author: SALHI Morgan	at Date, Time: Start: 2024-04-26 End: 2024-04-30	Verification Result: Fail
Requirement-Number: (CHECKLIST-compte-210) (CHECKLIST-compte-220) (CHECKLIST-compte-230)	Open NCRs:		Test Report Executed by: Salhi Morgan Test Report Check & Approved by: Labeste Bastien	<div><div>FAIL</div><div>PASS</div></div>
Test Description : Différents tests sur la création de compte, ces tests concernent : (CHECKLIST-compte-210) ; (CHECKLIST-compte-220) ; (CHECKLIST-compte-230)				
Test Preparation:				
No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks
i	Vérifier que l'identifiant ne peut pas contenir d'accent lors de la création du compte. (CHECKLIST-compte-210)	Message d'erreur affiché « The login does not meet the requirements.». Compte non enregistré dans la BDD.	Pass	
ii	Vérifier que l'identifiant est unique (CHECKLIST-compte-220)	Message d'erreur affiché « The login does not meet the requirements.». Compte non enregistré dans la BDD.	Fail	

iii	Vérifier que la création d'un compte est impossible avec un identifiant inférieur à 8 caractères et supérieur à 30. (CHECKLIST-compte-230)	Message d'erreur affiché « The login does not meet the requirements.». Compte non enregistré dans la BDD.	Pass	
Test Execution:				
1	Test de création de compte avec l'identifiant suivant « ifkgé » (CHECKLIST-compte-210)	Message d'erreur affiché « The login does not meet the requirements.». Compte non enregistré dans la BDD.	Pass	
2	Test de création de compte l'identifiant suivant « morgan ». (CHECKLIST-compte-220)	Message d'erreur affiché « The login does not meet the requirements.Compte non enregistré dans la BDD.	Fail	
3	Test de création de compte avec un identifiant inférieur à 8 caractères : « morgan » ; et supérieur à 30 caractères : « abababababababababababababa » (CHECKLIST-compte-230)	Message d'erreur affiché « The login does not meet the requirements.». Compte non enregistré dans la BDD.	Pass	
Execution duration: X hours				
Summary:				
La plupart des tests effectués sont fonctionnels et concordent bien avec les exigences, nous devons modifier les exigences sur le site pour que la (CHECKLIST-compte-220) soit respectée.				

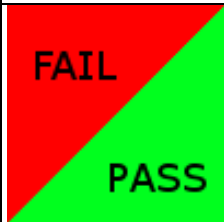
Tableau 18 : Rapport de test 2

<Product> Verification / Validation Report				<product id>
Verification-Number: 2		Test Procedure: Création de compte	at Date, Time: Start: 2024-04-26 End: 2024-04-30	Verification Result: Pass
Requirement-Number: (CHECKLIST-compte-240) (CHECKLIST-compte-250) (CHECKLIST-compte-260 à 290)	Open NCRs:	Issue : version 0.5 Short Description: Test exigence – création de comptes Date : 2024-04-30 Author: SALHI Morgan	Test Report Executed by: SALHI Morgan Test Report Check & Approved by: Labeste Bastien	<div>FAIL</div> <div>PASS</div>
Test Description: <test detailed description>				
Test Preparation:				
No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks
i	Vérifier que le mot de passe ne peut pas contenir le nom d'utilisateur lors de la création du compte. (CHECKLIST-compte-240)	Message d'erreur affiché « The password does not meet the requirements.». Compte non enregistré dans la BDD.	Pass	
ii	Vérifier que le mot de passe ne peut pas contenir d'accent lors de la création du compte ou d'un changement de celui-ci. (CHECKLIST-compte-250)	Message d'erreur affiché The password does not meet the requirements.». Compte non enregistré dans la BDD.	Pass	

Projet Checklist

iii	Vérifier que le mot de passe contient les caractères spécifiés lors de la création ou modification du mot de passe (CHECKLIST-compte-260 à 290)	Message d’erreur affiché The password does not meet the requirements.». Compte non enregistré dans la BDD.	Pass	
Test Execution:				
1	Test de création de compte avec le mot de passe suivant : « Morgan2024 !»	Message d’erreur affiché « The password does not meet the requirements.». Compte non enregistré dans la BDD.	Pass	
2	Test de création de compte avec le mot de passe suivant : « iut2024thalès ! »	Message d’erreur affiché « The password does not meet the requirements.». Compte non enregistré dans la BDD.	Pass	
3	Test de création de compte avec le mot de passe suivant : « morgan123 »	Message d’erreur affiché « The password does not meet the requirements.». Compte non enregistré dans la BDD.	Pass	
Execution duration: 1 hours				
Summary:				
Toutes les exigences sont respectées sans problèmes. La création du compte ne peut se faire si les exigences concernant le mot de passe ne sont pas respectées.				

Tableau 19 : Rapport de test 3

<Product> Verification / Validation Report					<product id>
Verification-Number: 3		Test Procedure: Création de compte Issue: version 0.5 Short Description: Test exigences – création de comptes Date: 2024-04-30 Author: SALHI Morgan		at Date, Time: Start: 2024-04-26 End: 2024-04-30 Test Report Executed by: Salhi Morgan Test Report Check & Approved by: Labeste Bastien	Verification Result: Fail
Requirement-Number: (CHECKLIST-compte-300) (CHECKLIST-compte-310) (CHECKLIST-compte-320)	Open NCRs:				
Test Description:					
Différents tests sur la création de compte, ces tests concernent : (CHECKLIST-compte-300) ; (CHECKLIST-compte-310); (CHECKLIST-compte-320)					
Test Preparation:					
No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks	
i	Vérifier que le mot de passe est stocké sous forme chiffrée (CHECKLIST-compte-300)	Le mot de passe doit être caché lorsque l'on va sur la page de connexion.	Pass		
ii	Vérifier que le compte est bloqué après 3 tentatives de connexion infructueuses (CHECKLIST-compte-310)	Le message suivant est affiché : « Too many failed login attempts! Please try again later.»	Fail		
iii	Vérifier que les paramètres n, p, q, r sont configurables uniquement par un Administrateur ou Super-administrateur (CHECKLIST-compte-320)	Impossible d'y accéder, certains boutons ne s'affichent et ne sont accessibles que pour un type d'utilisateur.	Pass		

Test Execution:				
1	Test pour mot de passe chiffré : Déconnexion d'un compte puis redirection sur la page de connexion. (CHECKLIST-compte-300)	Le mot de passe est chiffré.	Pass	
2	Test pour connexion échoué : connexion à un compte avec un mot de passe ne correspondant pas à l'identifiant. (CHECKLIST-compte-310)	Le message suivant est affiché : « Too many failed login attempts! Please try again later.»	Fail	
3	Test pour configuration des paramètres n, p, q, r : tentative d'accéder au menu administrateur depuis un compte utilisateur. (CHECKLIST-compte-320)	Impossible d'y accéder, certains boutons ne s'affichent et ne sont accessibles que pour un type d'utilisateur.	Pass	
Execution duration: 1 hour				
Summary:				
La plupart des tests effectués sont fonctionnels et concordent bien avec les exigences, nous devons modifier les exigences sur le site pour que la (CHECKLIST-compte-310) soit respectée.				

Tableau 20 : Rapport de test 4

<Product> Verification / Validation Report				<product id>	
Verification-Number: 4		Test Procedure: Création de compte	at Date, Time: Start: 2024-04-26 End: 2024-04-30	Verification Result: Pass	
Requirement-Number: (CHECKLIST-Auth-400) (CHECKLIST-Auth-410) (CHECKLIST-Auth-420)	Open NCRs:	Issue: version 0.5 Short Description: Test exigence – Inscription, connexion, déconnexion Date: 2024-04-30 Author: SALHI Morgan	Test Report Executed by: Salhi Morgan Test Report Check & Approved by: Labeste Bastien	<div><div>FAIL</div><div>PASS</div></div>	
Test Description:					
Différents tests sur l’inscription, la connexion et la déconnexion ces tests concernent : (CHECKLIST-Auth-400) ; (CHECKLIST-Auth-410) ; (CHECKLIST-Auth-420)					
Test Preparation:					
No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks	
i	Vérifier que l'application permet l'inscription avec un identifiant et un mot de passe (CHECKLIST-Auth-400)	Inscription avec identifiant et mot de passe	Pass		
ii	Vérifier que l'application permet la connexion avec un identifiant et un mot de passe (CHECKLIST-Auth-410)	Connexion avec identifiant et mot de passe	Pass		
iii	Vérifier que l'application permet la déconnexion (CHECKLIST-Auth-420)	Déconnexion possible	Pass		
Test Execution:					

Projet Checklist

1	Test pour inscription sur le site : création de compte avec un login valide « lut2024Projet » et mot de passe valide « Thales !7groupe11 ». (CHECKLIST-Auth-400)	Inscription avec identifiant et mot de passe.	Pass	
2	Test pour connexion sur le site : connexion avec un login valide « lut2024Projet » et mot de passe valide « Thales !7groupe11 ». (CHECKLIST-Auth-410)	Connexion avec identifiant et mot de passe	Pass	
3	Test pour déconnexion du site : Clique sur le bouton « Déconnexion »	Déconnexion réussie, renvoie sur la page de login	Pass	
Execution duration: 1 hours				
Summary:				
Toutes les exigences sont respectées sans problèmes. L'inscription, la connexion ainsi que la déconnexion sont des paramètres fonctionnels.				

Tableau 21 : Rapport de test 5

<Product> Verification / Validation Report					<product id>
Verification-Number: 5		Test Procedure: Authentication		at Date, Time: Start: 2024-04-26 End: 2024-04-30	Verification Result: Pass
Requirement-Number: (CHECKLIST-Auth-430) (CHECKLIST-Auth-440)	Open NCRs:	Issue: version 0.5 Short Description: Test exigence – authentication Date: 2024-04-30 Author: SALHI Morgan		Test Report Executed by: Salhi Morgan Test Report Check & Approved by: Labeste Bastien	<div>FAIL</div> <div>PASS</div>
Test Description: Différents tests sur la création de compte, ces tests concernent : (CHECKLIST-Auth-430) ; (CHECKLIST-Auth-440)					
Test Preparation:					
No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks	
i	Vérifier que l'application permet de modifier le mot de passe de son compte (CHECKLIST-Auth-430)	La modification du mot de passe à bien été réalisé.	Pass		
ii	Vérifier que l'application limite l'accès à certaines fonctionnalités en fonction du type d'utilisateur. (CHECKLIST-Auth-440)	Il est impossible de voir et d'accéder au menu administrateur en tant qu'utilisateur	Pass		
Test Execution:					
1	Test pour modifier le mot de passe de son compte avec le mot de passe suivant : « ProjetMorgan212 ! »	La modification du mot de passe à bien été réalisé.	Pass		

Projet Checklist

2	Test pour avoir accès au menu administrateur avec un compte utilisateur	Il est impossible de voir et d'accéder au menu administrateur en tant qu'utilisateur	Pass	
Execution duration: 1 hours				
Summary:				
Toutes les exigences sont respectées sans problèmes. L'utilisateur n'a pas accès au menu administrateur et le mot de passe peut être modifié si désirer.				

Tableau 22 : Rapport de test 6

<Product> Verification / Validation Report					<product id>
Verification-Number: 6		Test Procedure: Utilisateur		at Date, Time: Start: 2024-04-26 End: 2024-04-30	Verification Result: Pass
Requirement-Number: (CHECKLIST-Utilisateur-500) (CHECKLIST-Utilisateur-510) (CHECKLIST-Utilisateur-520)	Open NCRs:	Issue: version 0.5 Short Description: Test exigence – trie bonne pratiques Date: 2024-04-30 Author: SALHI Morgan		Test Report Executed by: Salhi Morgan Test Report Check & Approved by: Labeste Bastien	<div>FAIL</div> <div>PASS</div>
Test Description:					
Différents tests sur la parte utilisateur avec le tri et sélection des bonnes pratiques, ces tests concernent : (CHECKLIST-Utilisateur-500) ; (CHECKLIST-Utilisateur-510) ; (CHECKLIST-Utilisateur-520)					
Test Preparation:					
No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks	
i	Vérifier que l'application permet de sélectionner un programme pour voir toutes les « bonnes pratiques » associées (CHECKLIST-Utilisateur-500)	Programme sélectionné et bonnes pratiques affichées	Pass		
ii	Vérifier que l'application permet de sélectionner une phase pour voir toutes les « bonnes pratiques » associées (CHECKLIST-Utilisateur-510)	Phase sélectionnée et bonnes pratiques affichées	Pass		
iii	Vérifier que l'application permet d'écrire des mots-clés pour lister les « bonnes pratiques » correspondantes (CHECKLIST-Utilisateur-520)	Mots-clés utilisés pour lister les bonnes pratiques	Pass		
Test Execution:					

Projet Checklist

1	Test pour sélectionner un programme et y trouver les bonnes pratiques associées : Création de plusieurs bonnes pratiques avec une d'elle qui est associé aux bonnes pratiques. (CHECKLIST-Utilisateur-500)	Programme sélectionné et bonnes pratiques affichées	Pass	
2	Test pour sélectionner une phase et y trouver les bonnes pratiques associées : Création de plusieurs phases avec une d'elle qui est associé aux bonnes pratiques. (CHECKLIST-Utilisateur-510)	Phase sélectionnée et bonnes pratiques affichées	Pass	
3	Test pour trie des bonnes pratiques via la recherche de mots-clés : Création de bonnes pratiques qui contiennent des mots clés puis recherche dans la barre de recherche en tapant des mots clés de bonnes pratiques. (CHECKLIST-Utilisateur-520)	Les bonnes pratiques s'affichent en fonction du mot clé tapé.	Pass	
Execution duration: 1 hours				
Summary:				
Toutes les exigences sont respectées sans problèmes. Le tri et la sélection des bonnes pratiques se fait en fonction des paramètres entrés par l'utilisateur.				

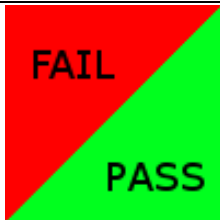
Tableau 23 : Rapport de test 7

<Product> Verification / Validation Report					<product id>
Verification-Number: 7		Test Procedure: Utilisateur Issue: version 0.5 Short Description: Test exigence – suppression de variables et affichage bonnes pratiques Date: 2024-04-30 Author: SALHI Morgan	at Date, Time: Start: 2024-04-26 End: 2024-04-30		Verification Result: Pass
Requirement-Number: (CHECKLIST-Utilisateur-530) (CHECKLIST-Utilisateur-540) (CHECKLIST-Utilisateur-550)	Open NCRs:		Test Report Executed by: Salhi Morgan Test Report Check & Approved by: Labeste Bastien		<div><div>FAIL</div><div>PASS</div></div>
Test Description: Différents tests sur la création de compte, ces tests concernent : (CHECKLIST-Auth-430) ; (CHECKLIST-Auth-440)					
Test Preparation:					
No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks	
i	Vérifier que l'application permet d'exclure un ou plusieurs programmes pour afficher les « bonnes pratiques » restantes (CHECKLIST-Utilisateur-530)	Programmes exclus et bonnes pratiques restantes affichées	Pass		
ii	Vérifier que l'application permet d'exclure une ou plusieurs phases pour afficher les « bonnes pratiques » restantes (CHECKLIST-Utilisateur-540)	Phases exclues et bonnes pratiques restantes affichées	Pass		

Projet Checklist

iii	Vérifier que l'application permet d'exclure un ou plusieurs mots-clés pour afficher les « bonnes pratiques » restantes (CHECKLIST-Utilisateur-550)	Mots-clés exclus et bonnes pratiques restantes affichées	Pass	
Test Execution:				
1	Test pour l'exclusion de de programmes avec un affichage des bonnes pratiques restantes : création de plusieurs programmes reliés à différentes bonnes pratiques, puis suppression de certains de ces programmes. (CHECKLIST-Utilisateur-530)	Exclusion de ces programmes et affichage des bonnes pratiques restantes fonctionnel.	Pass	
2	Test pour l'exclusion de phases avec un affichage des bonnes pratiques restantes : création de plusieurs phases reliées à différentes bonnes pratiques, puis suppression de certaines de ces phases. (CHECKLIST-Utilisateur-540)	Exclusion de ces phases et affiches des bonnes pratiques restantes fonctionnels.	Pass	
3	Test pour l'exclusion des mots-clés avec un affichage des bonnes pratiques restantes : création de plusieurs mots-clés reliés à différentes bonnes pratiques, puis suppression de certains de ces mots-clés. (CHECKLIST-Utilisateur-550)	Exclusion de ces mots-clés et affichage des bonnes pratiques restants fonctionnels.	Pass	
Execution duration: 1 hours				
Summary:				
Toutes les exigences sont respectées sans problèmes. La suppression des programmes, phases et mots-clés ont un impactent sur le nombre de bonnes pratiques affichés qui s'effacent si elles étaient reliées à l'un de ces 3 paramètres.				

Tableau 24 : Rapport de test 8

<Product> Verification / Validation Report				<product id>	
Verification-Number: 8		Test Procedure: Utilisateur	at Date, Time: Start: 2024-04-26 End: 2024-04-30	Verification Result: Fail	
Requirement-Number: (CHECKLIST-Utilisateur-560) (CHECKLIST-Utilisateur-570) (CHECKLIST-Utilisateur-580)	Open NCRs:	Issue: version 0.5 Short Description: Test exigence – création, duplication et suppression de bonnes pratiques Date: 2024-04-30 Author: SALHI Morgan	Test Report Executed by: Salhi Morgan Test Report Check & Approved by: Labeste Bastien		
Test Description:					
Différents tests sur les bonnes pratiques en les créant, les dupliquant et les supprimant : (CHECKLIST-Utilisateur-560) ; (CHECKLIST-Utilisateur-570) ; (CHECKLIST-Utilisateur-580)					
Test Preparation:					
No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks	
i	Vérifier que l'application permet d'écrire de nouvelles bonnes pratiques pour un programme donné et une phase donnée (CHECKLIST-Utilisateur-560)	Nouvelles bonnes pratiques écrites	Pass		
ii	Vérifier que l'application permet de dupliquer des bonnes pratiques d'un autre programme pour un programme donné et une phase donnée (CHECKLIST-Utilisateur-570)	Bonnes pratiques dupliquées	Pass		

iii	Vérifier que l'application permet de supprimer des bonnes pratiques pour un programme donné et une phase donnée (CHECKLIST-Utilisateur-580)	Bonnes pratiques supprimées	Fail	
Test Execution:				
1	Test pour écrire une bonne pratique : association à un programme ainsi qu'à une phase. (CHECKLIST-Utilisateur-560)	Nouvelles bonnes pratiques écrites	Pass	
2	Test pour dupliquer une bonne pratique : clique sur le bouton « dupliquer » à côté de la bonne pratique. (CHECKLIST-Utilisateur-570)	Bonnes pratiques dupliquées	Pass	
3	Test pour supprimer une bonne pratique. (CHECKLIST-Utilisateur-580)	Erreur dans la suppression de la bonne pratique, l'erreur vient de la base de données.	Fail	
Execution duration: 1 hours				
Summary:				
Les tests concernant les deux premiers tests sont fonctionnels, cependant le dernier test est un échec, la bonne pratique ne se supprime pas.				

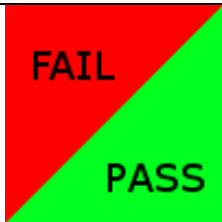
Tableau 25 : Rapport de test 9

<Product> Verification / Validation Report					<product id>
Verification-Number: 9		Test Procedure: Administrateur		at Date, Time: Start: 2024-04-26 End: 2024-04-30	Verification Result: Pass
Requirement-Number: (CHECKLIST-admin-600) (CHECKLIST-admin-610) (CHECKLIST-admin-620)	Open NCRs:	Issue: version 0.5 Short Description: Test exigence – Vérification des droits des administrateurs Date: 2024-04-30 Author: SALHI Morgan		Test Report Executed by: Salhi Morgan Test Report Check & Approved by: Labeste Bastien	<div><div>FAIL</div><div>PASS</div></div>
Test Description:					
Différents tests sur les droits et pouvoir des administrateurs, ces tests concernent : (CHECKLIST-admin-600) ; (CHECKLIST-admin-610) ; (CHECKLIST-admin-620)					
Test Preparation:					
No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks	
i	Vérifier que l'application possède des comptes administrateurs (CHECKLIST-admin-600)	Comptes administrateurs présents	Pass		
ii	Vérifier que les comptes administrateurs peuvent supprimer/ajouter des comptes opérateurs (CHECKLIST-admin-610)	Comptes opérateurs peuvent être ajoutés/supprimés	Pass		
iii	Vérifier que les comptes administrateurs peuvent contrôler les accès des utilisateurs (CHECKLIST-admin-620)	Accès des utilisateurs contrôlés par les administrateurs	Pass		
Test Execution:					

Projet Checklist

1	Test pour vérifier la présence d'administrateurs sur le site : (CHECKLIST-admin-600)	Comptes administrateurs présents	Pass	
2	Test pour ajouter/supprimer des comptes opérateurs : Connexion sur un compte administrateur, accès au menu au menu administrateur et création/suppression de compte de type opérateur. (CHECKLIST-admin-610)	Comptes opérateurs peuvent être ajoutés/supprimés. Message suivant affichés en cas de création : « The user account has been successfully created. » Message suivant en cas de suppression de compte opérateur : The user account has been successfully deleted. ».	Pass	
3	Test pour contrôle des accès sur les comptes de type utilisateur : connexion sur un compte administrateur, accès au menu administrateur,	Accès des utilisateurs contrôlés par les administrateurs.	Pass	
Execution duration: 1 hours				
<p>Summary:</p> <p>Toutes les exigences sont respectées sans problèmes. L'administrateur a tous les pouvoirs qu'il est censé avoir, ils sont tous fonctionnels.</p>				

Tableau 26 : Rapport de test 10

<Product> Verification / Validation Report					<product id>
Verification-Number: 10		Test Procedure: Administrateur		at Date, Time: Start: 2024-04-26 End: 2024-04-30	Verification Result: Pass
Requirement-Number: (CHECKLIST-admin-630) (CHECKLIST-admin-650) (CHECKLIST-admin-660)	Open NCRs:	Issue: version 0.5 Short Description: Test exigence – Vérification des droits des administrateurs Date: 2024-04-30 Author: SALHI Morgan		Test Report Executed by: Salhi Morgan Test Report Check & Approved by: Labeste Bastien	
Test Description: Différents tests sur les droits et pouvoir des administrateurs, ces tests concernent : (CHECKLIST-admin-630) ; (CHECKLIST-admin-650) ; (CHECKLIST-admin-660)					
Test Preparation:					
No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks	
i	Vérifier que les comptes administrateurs peuvent afficher la liste d'utilisateurs par ordre alphabétique (CHECKLIST-admin-630)	Liste d'utilisateurs affichée par ordre alphabétique	Pass		
ii	Vérifier que les comptes administrateurs peuvent voir l'historique d'interactions des utilisateurs (CHECKLIST-admin-650)	Historique d'interactions affiché	Pass		
iii	Vérifier que l'application permet à un utilisateur de type "Administrateur" ou "Super-administrateur" de modifier le mot de passe d'un utilisateur dont le compte est verrouillé (CHECKLIST-admin-660)	Modification du mot de passe possible	Pass		

Projet Checklist

Test Execution:				
1	Test pour tri des utilisateurs par ordre alphabétique pour les administrateurs : Connexion sur un compte administrateur, menu administrateur, liste des utilisateurs. (CHECKLIST-admin-630)	Liste d'utilisateurs affichée par ordre alphabétique	Pass	
2	Test pour afficher l'historique des interactions des utilisateurs : connexion sur un compte administrateur, clique sur l'emplacement à droite de la page d'accueil correspondants aux « logs ». (CHECKLIST-admin-650)	Historique d'interactions affiché	Pass	
3	Test pour modifier le mot de passe d'un compte bloqué : connexion sur un compte administrateur, menu administrateur, liste des utilisateurs, clique sur le compte désiré puis modifié mot de passe. (CHECKLIST-admin-660)	Modification du mot de passe possible	Pass	
Execution duration: 1 hours				
Summary:				
Toutes les exigences sont respectées sans problèmes. L'administrateur a tous les pouvoirs qu'il est censé avoir, ils sont tous fonctionnels.				

Tableau 27 : Rapport de test 11

<Product> Verification / Validation Report					<product id>
Verification-Number: 11		Test Procedure: Administrateur		at Date, Time: Start: 2024-04-26 End: 2024-04-30	Verification Result: Pass
Requirement-Number: (CHECKLIST-admin-670) (CHECKLIST-admin-680) (CHECKLIST-admin-690)	Open NCRs:	Issue: version 0.5 Short Description: Test exigence – Vérification des droits Date: 2024-04-30 Author: SALHI Morgan		Test Report Executed by: Salhi Morgan Test Report Check & Approved by: Labeste Bastien	<div>FAIL</div> <div>PASS</div>
Test Description:					
Différents tests sur les droits et pouvoir des administrateurs, ces tests concernent : (CHECKLIST-admin-670) ; (CHECKLIST-admin-680) ; (CHECKLIST-admin-690)					
Test Preparation:					
No.	Activity Description		Pass/Fail Criteria	Status (Pass, Failed)	Remarks
i	Vérifier que les comptes administrateurs peuvent créer ou supprimer des programmes (CHECKLIST-admin-670)		Programmes créés ou supprimés	Pass	
ii	Vérifier que les comptes administrateurs peuvent supprimer définitivement des bonnes pratiques pour un programme donné et une phase donnée (CHECKLIST-admin-680)		Bonnes pratiques supprimées	Pass	
iii	Vérifier que l'application permet aux "Administrateurs" d'effectuer les mêmes actions qu'un utilisateur "Opérateur" (CHECKLIST-admin-690)		Actions opérateur disponibles pour administrateurs	Pass	
Test Execution:					

Projet Checklist

1	Test pour créer ou supprimer les programmes : connexion sur un compte administrateur, accès à la page des programmes, créer via le menu basique, supprimer en sélectionnant le(s) programme(s) désiré(s). (CHECKLIST-admin-670)	Programmes créés ou supprimés. Message suivant affiché en cas de création : « The program has been created. ». Message suivant en cas de suppression : « The program has been deleted. ».	Pass	
2	Test pour supprimer définitivement une bonne pratique : connexion sur un compte administrateur, accès à la page des bonnes pratiques puis supprimer en sélectionnant celle(s) désirée(s). Accéder à la page des bonnes pratiques puis les supprimer définitivement. (CHECKLIST-admin-680)	Programmes créés ou supprimés. Message suivant affiché en cas de création : « The best practice has been created. ». Message suivant en cas de suppression : « The best practice has been deleted. ».	Pass	
3	Test pour vérifier qu'un administrateur puissent faire les mêmes choses qu'un opérateur : Connexion sur un compte opérateur et administrateur, puis comparaison des fonctionnalités possibles. (CHECKLIST-admin-690)	Actions opérateur disponibles pour administrateurs	Pass	
Execution duration: 1 hours				
Summary:				
Toutes les exigences sont respectées sans problèmes. L'administrateur a tous les pouvoirs qu'il est censé avoir, ils sont tous fonctionnels.				

Tableau 28 : Rapport de test 12

<Product> Verification / Validation Report					<product id>
Verification-Number: <div>12</div>		Test Procedure: Création de compte	System Under Test: <identification>	at Date, Time: Start: 2024-06-01 End: 2024-06-01	Verification Result: Pass
Requirement-Number: (CHECKLIST-compte-220)	Open NCRs:	Issue: version VF Short Description : Test exigences – création de comptes Date: 2024-04-30 Author: SALHI Morgan	Issue SUT: VF	Test Report Executed by: SALHI Morgan Test Report Check & Approved by: Labeste Bastien	<div>FAIL</div> <div>PASS</div>
Test Description :					
Test pour la (CHECKLIST-compte-220) qui ne marchait pas dans la version précédente					
Test Preparation:					
No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks	
i	Vérifier que l'identifiant est unique (CHECKLIST-compte-220)	Message d'erreur affiché « The login does not meet the requirements.». Compte non enregistré dans la BDD.	Pass		
Test Execution:					
1	Test de création de compte l'identifiant suivant « morgan ». (CHECKLIST-compte-220)	Message d'erreur affiché « The login does not meet the requirements.». Compte non enregistré dans la BDD.	Pass		

Execution duration: 10 minutes	
Summary : Désormais la (CHECKLIST-compte-220) est respectée.	

Tableau 29 : Rapport de test 13

<Product> Verification / Validation Report					<product id>
Verification-Number: 13		Test Procedure: Création de compte	System Under Test: <identification>	at Date, Time: Start: 2024-06-01 End: 2024-06-01	Verification Result: Pass
Requirement-Number: (CHECKLIST-compte-310)	Open NCRs:	Issue: version VF Short Description : Test exigences – création de comptes Date: 2024-04-30 Author: SALHI Morgan	Issue SUT: VF	Test Report Executed by: SALHI Morgan Test Report Check & Approved by: Labeste Bastien	<div>FAIL</div> <div>PASS</div>
Test Description :					
Test pour la (CHECKLIST-compte-310) qui ne marchait pas dans la version précédente					
Test Preparation:					
No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks	
i	Vérifier que le compte est bloqué après 3 tentatives de connexion infructueuses (CHECKLIST-compte-310)	Le message suivant est affiché : « Too many failed login attempts! Please try again later.»	Pass		
Test Execution:					
1	Test pour connexion échoué : connexion à un compte avec un mot de passe ne correspondant pas à l'identifiant. (CHECKLIST-compte-310)	Le message suivant est affiché : « Too many failed login attempts! Please try again later.»	Pass		

Execution duration: 10 minutes	
Summary : Désormais la (CHECKLIST-compte-310) est respectée.	

Tableau 30 : Rapport de test 14

<Product> Verification / Validation Report					<product id>
Verification-Number: 14		Test Procedure: Utilisateur	System Under Test: <identification>	at Date, Time: Start: 2024-06-01 End: 2024-06-01	Verification Result: Pass
Requirement-Number: (CHECKLIST-Utilisateur-580)	Open NCRs:	Issue: version VF Short Description : Test exigences – suppression bonne pratique Date: 2024-04-30 Author: SALHI Morgan	Issue SUT: VF	Test Report Executed by: SALHI Morgan Test Report Check & Approved by: Labeste Bastien	<div><div>FAIL</div><div>PASS</div></div>
Test Description :					
Test pour la (CHECKLIST-compte-220) qui ne marchait pas dans la version précédente					
Test Preparation:					
No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks	
i	Vérifier que l'application permet de supprimer des bonnes pratiques pour un programme donné et une phase donnée (CHECKLIST-Utilisateur-580)	Bonnes pratiques supprimées	Pass		
Test Execution:					
1	Test pour supprimer une bonne pratique. (CHECKLIST-Utilisateur-580)	Erreur dans la suppression de la bonne pratique, l’erreur vient de la base de données.	Pass		

Execution duration: 10 minutes	
Summary : Désormais la (CHECKLIST-Utilisateur-580) est respectée.	

Tableau 31 : Rapport de test 15

<Product> Verification / Validation Report					<product id>	
Verification-Number: 15		Test Procedure: Super-administrateur		at Date, Time: Start: 2024-04-26 End: 2024-04-30	Verification Result: Fail	
Requirement-Number: (CHECKLIST-SuperAdmin-700) (CHECKLIST-SuperAdmin-710) (CHECKLIST-SuperAdmin-720)	Open NCRs:	Issue: version 0.5 Short Description: Test exigence – unicité et droits Date: 2024-04-30 Author: Salhi Morgan		Test Report Executed by: Salhi Morgan Test Report Check & Approved by: Labeste Bastien	<div>FAIL</div> <div>PASS</div>	
Test Description:						
Différents tests sur les droits et pouvoir des super-administrateurs, ces tests concernent : (CHECKLIST-SuperAdmin-700) ; (CHECKLIST-SuperAdmin-710) ; (CHECKLIST-SuperAdmin-720)						
Test Preparation:						
No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks		
i	Vérifier que l'application ne peut posséder qu'un seul super-administrateur. (CHECKLIST-SuperAdmin-700)	Un seul super-administrateur présent sur le site.	Fail			
ii	Vérifier qu'un compte super-administrateur peut donner ou retirer les droits d'un compte administrateur. (CHECKLIST-SuperAdmin-710)	Le super-administrateur est capable de donner ou retirer des droits à un administrateur.	Pass			
iii	Vérifier qu'un compte super-administrateur est capable de supprimer un compte administrateur. (CHECKLIST-SuperAdmin-720)	Le super-administrateur est capable de supprimer un compte administrateur	Pass			
Test Execution:						

Projet Checklist

1	Test pour vérifier qu'un seul super-administrateur est présent sur le site : Connexion, puis vérification de la liste des utilisateurs en triant par type de compte. (CHECKLIST-SuperAdmin-700)	Un seul super-administrateur	Fail	
2	Test pour donner ou retirer des droits à un compte administrateur depuis un compte super-administrateur : connexion super-administrateur puis direction sur la page « modify user role » en sélectionnant l'utilisateur et le rôle désiré. (CHECKLIST-SuperAdmin-710)	Affichage du message suivant : « The account rights have been deleted»	Pass	
3	Test pour supprimer un compte administrateur en tant que super-administrateur : connexion sur un compte super-administrateur puis direction sur la page « delete account ». (CHECKLIST-SuperAdmin-720)	Affichage du message suivant : "The account have been deleted"	Pass	
Execution duration: 1 hours				
Summary:				
Les tests sont quasiment tous fonctionnels, il faut cependant revoir l'unicité du super-administrateur. (CHECKLIST-SuperAdmin-700)				

Tableau 32 : Rapport de test 16

<Product> Verification / Validation Report					<product id>
Verification-Number: 16		Test Procedure: Super-administrateur		at Date, Time: Start: 2024-04-26 End: 2024-04-30	Verification Result: Pass
Requirement-Number: (CHECKLIST-SuperAdmin-740) (CHECKLIST-SuperAdmin-750)	Open NCRs:	Issue: version 0.5 Short Description: Test exigence – pouvoir Date: 2024-04-30 Author: SALHI Morgan		Test Report Executed by: Salhi Morgan Test Report Check & Approved by: Labeste Bastien	<div>FAIL</div> <div>PASS</div>
Test Description: Différents tests sur le compte administrateur pour vérification des pouvoirs, ces tests concernent : (CHECKLIST-SuperAdmin-740) ; (CHECKLIST-SuperAdmin-750)					
Test Preparation:					
No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks	
i	Vérifier que l'application est dans l'incapacité de bloquer le compte super-administrateur. (CHECKLIST-SuperAdmin-740)	L'application est incapable de bloquer le super-administrateur.	Pass		
ii	Vérifier que l'application permet au "Super-Administrateurs" d'effectuer les mêmes actions qu'un utilisateur "Administrateur". (CHECKLIST-SuperAdmin-750)	Le super-administrateur peut effectuer les mêmes actions qu'un administrateur.	Pass		
Test Execution:					
1	Test pour vérifier que le super-administrateur ne peut être bloquer par l'application :		Pass		

2	Test pour vérifier que le super-administrateur puisse faire les mêmes choses qu'un administrateur : Connexion sur un compte super-administrateur et administrateur, puis comparaison des fonctionnalités possibles. (CHECKLIST-admin-690)	Les actions réalisables par l'administrateur peuvent l'être également pour le super-administrateur.	Pass	
Execution duration: 1 hours				
Summary:				
Toutes les exigences sont respectées sans problèmes. Le super-administrateur ne peut être bloqué par l'application et il est en capacité de faire les mêmes actions qu'un administrateur.				

Tableau 33 : Rapport de test 17

<Product> Verification / Validation Report					<product id>
Verification-Number: 17		Test Procedure: Super-administrateur	System Under Test: <identification>	at Date, Time: Start: 2024-06-01 End: 2024-06-07	Verification Result: Pass <div>FAIL PASS</div>
Requirement-Number: (CHECKLIST-SuperAdmin-700)	Open NCRs:	Issue: version VF Short Description : Test exigences – unicité Date: 2024-06-07 Author: SALHI Morgan	Issue SUT: VF	Test Report Executed by: SALHI Morgan Test Report Check & Approved by: Labeste Bastien	
Test Description :					
Test pour la (CHECKLIST-compte-220) qui ne marchait pas dans la version précédente					
Test Preparation:					
No.	Activity Description	Pass/Fail Criteria	Status (Pass, Failed)	Remarks	
i	Vérifier que l’application ne peut posséder qu’un seul super-administrateur. (CHECKLIST-SuperAdmin-700)	Un seul super-administrateur présent sur le site.	Pass		
Test Execution:					
1	Test pour vérifier qu’un seul super-administrateur est présent sur le site : Connexion, puis vérification de la liste des utilisateurs en triant par type de compte. (CHECKLIST-SuperAdmin-700)	Un seul super-administrateur	Pass		

Execution duration: 10 minutes	
Summary : Désormais la (CHECKLIST-SuperAdmin-700) est respectée.	

7.3. Annexe C : Fiches d'Anomalies

Tableau 34 : Fiche d'anomalie 1

Champ	Description
Product Verification / Validation Report ID	1
Verification-Number	8
Test Procedure	Utilisateur
Issue Version	0.5
Short Description	Test exigence – création, duplication et suppression de bonnes pratiques
Date	2024-04-30
Author	SALHI Morgan
Verification Result	Fail
Requirement-Number	(CHECKLIST-Utilisateur-580)
Description de l'Anomalie	L'application échoue à supprimer des bonnes pratiques pour un programme donné et une phase donnée.
Étapes pour Reproduire	1. Connexion sur le compte utilisateur. 2. Accès à la section des bonnes pratiques. 3. Tentative de suppression d'une bonne pratique existante.
Résultat Attendu	La bonne pratique doit être supprimée avec succès.
Résultat Observé	Erreur dans la suppression de la bonne pratique, l'erreur vient de la base de données.
Impact	Les utilisateurs ne peuvent pas supprimer des bonnes pratiques, ce qui affecte la gestion des données.

Tableau 35 : Fiche d'anomalie 2

Champ	Description
Product Verification / Validation Report ID	2
Verification-Number	1
Test Procedure	Création de compte
Issue Version	0.5
Short Description	Test exigence – création de comptes
Date	2024-04-30
Author	SALHI Morgan
Verification Result	Fail
Requirement-Number	(CHECKLIST-compte-220)
Description de l'Anomalie	L'application échoue à vérifier l'unicité des identifiants lors de la création de compte.
Étapes pour Reproduire	1. Tentative de création de compte avec l'identifiant "morgan".
Résultat Attendu	Message d'erreur affiché "Le login ne correspond pas aux exigences". Compte non enregistré dans la BDD.
Résultat Observé	Échec dans la vérification de l'unicité des identifiants.
Impact	Les utilisateurs peuvent créer des comptes avec des identifiants non uniques, ce qui peut causer des conflits dans la base de données.

Tableau 36 : Fiche d'anomalie 3

Champ	Description
Product Verification / Validation Report ID	3
Verification-Number	3
Test Procedure	Création de compte
Issue Version	0.5
Short Description	Test exigences – création de comptes
Date	2024-04-30
Author	SALHI Morgan
Verification Result	Fail
Requirement-Number	(CHECKLIST-compte-310)
Description de l'Anomalie	L'application échoue à bloquer un compte après 3 tentatives de connexion infructueuses.
Étapes pour Reproduire	1. Tentative de connexion à un compte avec un mot de passe incorrect, répétée 3 fois.
Résultat Attendu	Message affiché "Trop de connexions échouées ! Veuillez essayer plus tard."
Résultat Observé	Échec dans le blocage du compte après 3 tentatives de connexion infructueuses.
Impact	Les utilisateurs peuvent tenter de se connecter de manière illimitée, ce qui expose le système à des attaques par force brute.

7.4. Annexe D : Matrice de Traçabilité

Tableau 37 : Matrice de traçabilité

Requirement-Number	Verification-Number	Test Procedure	Issue Version	Test Report Executed by	Test Report Check & Approved by	Résultat de la Vérification
CHECKLIST-Utilisateur-500	6	Utilisateur	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-Utilisateur-510	6	Utilisateur	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-Utilisateur-520	6	Utilisateur	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-Utilisateur-530	7	Utilisateur	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-Utilisateur-540	7	Utilisateur	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-Utilisateur-550	7	Utilisateur	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-Utilisateur-560	8	Utilisateur	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-Utilisateur-570	8	Utilisateur	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-Utilisateur-580	8	Utilisateur	0.5	Salhi Morgan	Labeste Bastien	Fail
CHECKLIST-Utilisateur-580	14	Utilisateur	VF	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-compte-210	1	Création de compte	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-compte-220	1	Création de compte	0.5	Salhi Morgan	Labeste Bastien	Fail
CHECKLIST-compte-230	1	Création de compte	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-compte-240	2	Création de compte	0.5	Salhi Morgan	Labeste Bastien	Pass

Requirement-Number	Verification-Number	Test Procedure	Issue Version	Test Report Executed by	Test Report Check & Approved by	Résultat de la Vérification
CHECKLIST-compte-250	2	Création de compte	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-compte-260 à 290	2	Création de compte	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-compte-300	3	Création de compte	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-compte-310	3	Création de compte	0.5	Salhi Morgan	Labeste Bastien	Fail
CHECKLIST-compte-320	3	Création de compte	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-Auth-400	4	Création de compte	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-Auth-410	4	Création de compte	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-Auth-420	4	Création de compte	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-Auth-430	5	Création de compte	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-Auth-440	5	Création de compte	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-compte-220	12	Création de compte	VF	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-compte-310	13	Création de compte	VF	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-admin-600	9	Administrateur	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-admin-610	9	Administrateur	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-admin-620	9	Administrateur	0.5	Salhi Morgan	Labeste Bastien	Pass

Requirement-Number	Verification-Number	Test Procedure	Issue Version	Test Report Executed by	Test Report Check & Approved by	Résultat de la Vérification
CHECKLIST-admin-630	10	Administrateur	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-admin-650	10	Administrateur	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-admin-660	10	Administrateur	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-admin-670	11	Administrateur	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-admin-680	11	Administrateur	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-admin-690	11	Administrateur	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-admin-700	15	Super-administrateur	0.5	Salhi Morgan	Labeste Bastien	Fail
CHECKLIST-admin-710	16	Super-administrateur	0.5	Salhi Morgan	Labeste Bastien	Pass
CHECKLIST-admin-720	17	Super-administrateur	VF	Salhi Morgan	Labeste Bastien	Pass