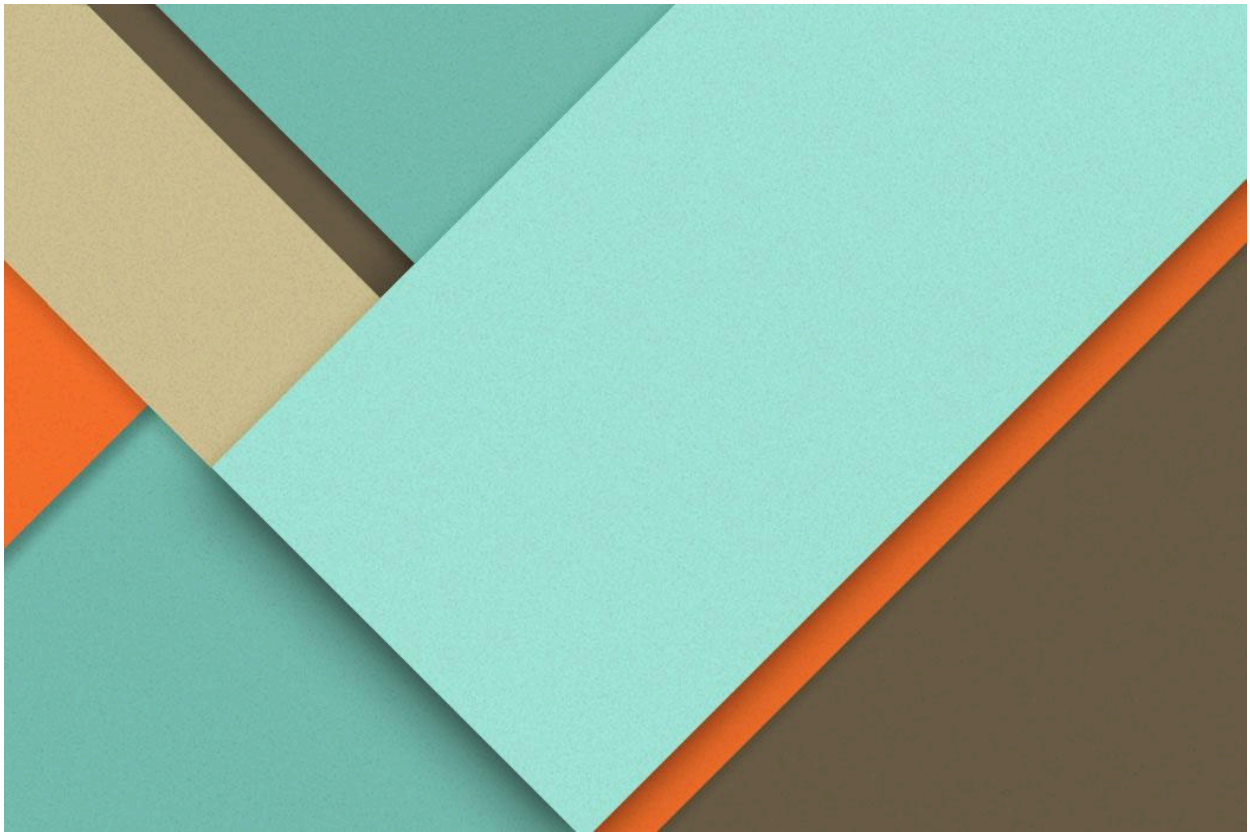


# Cahier des Charges Fonctionnel

## Send-It

03.03.2025

Mélanie TROUTIER & Yulian GUINAND
BTS SIO

Version 1.0

# AUTEURS

| Nom | Rôle | Département |
|---|---|---|
| Yulian GUINAND | Développeur | BTS SIO Option : SLAM |
| Mélanie TROUTIER | Développeur | BTS SIO Option : SLAM |

# HISTORIQUE DES DOCUMENTS

| Date | Version | Description du document | Auteur du document |
|---|---|---|---|
| 03.03.2025 | 1.0 | Première version du Cahier des Charges | GUINAND Yulian<br>TROUTIER Mélanie |

# APPROBATIONS

| Date d'approbation | Version approuvée | Rôle de l'approbateur | Approbateur |
|---|---|---|---|
| | | | |

# TABLE DES MATIÈRES

**AUTEURS** ..... **1**
**HISTORIQUE DES DOCUMENTS** ..... **2**
**APPROBATIONS** ..... **3**
**TABLE DES MATIÈRES** ..... **4**
**1. CONTEXTE ET DÉFINITION DU PROJET** ..... **5**
**2. OBJECTIFS DU PROJET** ..... **6**
**3. PÉRIMÈTRE DU PROJET** ..... **7**
**4. DESCRIPTION FONCTIONNELLE DES BESOINS** ..... **9**
1. Gestion des fichiers ..... 9
1.1 Dépôt de fichiers ..... 9
1.2 Restrictions et validations ..... 10
2. Partage et gestion des liens ..... 11
2.1 Génération des liens sécurisés ..... 11
2.2 Options avancées pour utilisateurs connectés ..... 11
3. Téléchargement et accès aux fichiers ..... 12
3.1 Processus de téléchargement ..... 12
3.2 Sécurité et restrictions ..... 13
4. Gestion des utilisateurs ..... 15
4.1 Inscription et connexion ..... 15
4.2 Espace utilisateur ..... 17
5. Gestion des fichiers sur le serveur ..... 19
5.1 Stockage et suppression automatique ..... 19
5.2 Optimisation et performance ..... 20
6. Sécurité et protection des données ..... 20
6.1 Chiffrement et sécurité des fichiers ..... 20
6.2 Protection contre les attaques ..... 21
7. Expérience utilisateur et interface ..... 23
7.1 Design et ergonomie ..... 23
7.2 Notifications et suivi ..... 25
**5. ENVELOPPE BUDGÉTAIRE** ..... **27**
Coût de la main-d'œuvre ..... 27
**6. Délais** ..... **28**
Planning prévisionnel ..... 28
Diagramme de Gantt prévisionnel ..... 29

# 1. CONTEXTE ET DÉFINITION DU PROJET

En 2021, environ 319.6 milliards d'e-mails étaient envoyés et reçus chaque jour. Ce chiffre devrait dépasser 392.5 milliards en 2026. Pour les comptes de messagerie Internet tels que Outlook.com ou Gmail, la taille limite des e-mails est de 20 mégaoctets (Mo), ce qui correspond à 8 ou 9 images en format jpeg. Pour les comptes Exchange (e-mail professionnel), la limite de taille de courrier par défaut est de 10 Mo (selon une étude de Microsoft). Cela pose de nombreux problèmes, notamment une limitation du nombre de pièces jointes que l'on peut envoyer, ou encore une sécurisation limitée des données, ... Étant donné l'accroissement du nombre d'e-mails envoyés ou reçus sur les dernières années, il serait intéressant de trouver une solution viable pour pouvoir augmenter la capacité des fichiers envoyés.

## 2. OBJECTIFS DU PROJET

Nous voulons permettre à des utilisateurs, se connectant ou non à Send-It, de déposer des fichiers de pratiquement tous les types (exceptés les .php).

Les utilisateurs non connectés auront accès à un espace de maximum 2 Go et les utilisateurs se connectant pourront déposer jusqu'à 15 Go de fichiers.

Dans le but de ne pas laisser des fichiers encombrer Send-It pour une durée indéterminée, nous les laisserons accessibles durant une semaine après l'ajout des fichiers sur le système. Autrement dit, les fichiers se supprimeront d'eux-mêmes au bout de 7 jours.

L'utilisateur devra fournir à la personne à laquelle il souhaite transmettre un ou plusieurs fichiers, un lien et un mot de passe générés aléatoirement. L'utilisateur aura la possibilité de mettre en ligne plusieurs dossiers de fichiers et de créer des liens par dossier, par fichiers ou par groupement de fichiers (en sélectionnant les fichiers concernés).

L'utilisateur non connecté n'aura pas la possibilité de créer des dossiers de fichiers et d'organiser ceux-ci, il pourra juste mettre un maximum de 10 fichiers dans un formulaire et de générer un lien et un mot de passe, eux aussi disponible pendant une durée de 1 heure. Un décompte apparaît pour lui signifier la durée pendant laquelle ils sont disponibles. A la fin de l'heure, si les fichiers n'ont pas été téléchargés, ils se suppriment automatiquement.

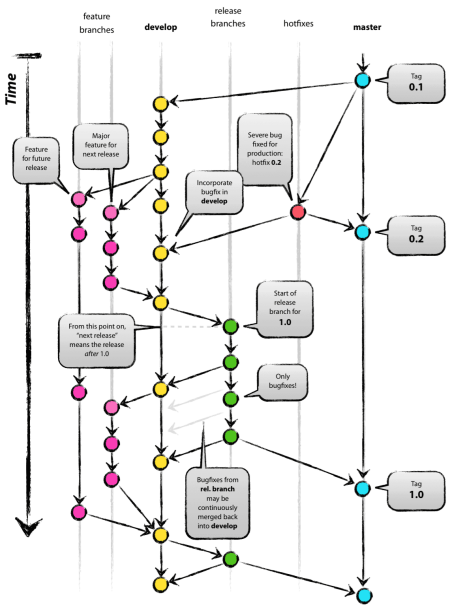
# 3. PÉRIMÈTRE DU PROJET

Le projet repose sur une architecture bien définie, choisie pour optimiser la performance et garantir une expérience utilisateur fluide et sécurisée.

Du côté **Backend**, la technologie utilisée est PHP 8.4, un langage robuste et performant, sans framework particulier afin de garder un maximum de flexibilité et de contrôle sur l'architecture du projet. Le choix de **PostgreSQL** pour la base de données repose sur sa stricte gestion des types de données, ce qui assure une meilleure intégrité et une expérience plus fiable dans la manipulation des informations.

Le **Frontend** est développé en **JavaScript**, sans recourir à jQuery, afin de maximiser la légèreté du code et de favoriser une approche moderne et native du langage. Le **framework CSS** utilisé est **Tailwind CSS**, choisi non seulement pour sa flexibilité mais aussi pour l'expérience plus poussée de l'équipe avec cet outil par rapport à Bootstrap. Cela permet une personnalisation plus fine et une gestion plus intuitive des styles.

Le **versioning** du projet est géré via **Gitea**, et un processus GitFlow est mis en place pour assurer une organisation optimale du code. Ainsi, la branche **development** est dédiée au développement en cours, tandis que la branche **main** est réservée à la version prête à être déployée. Des branches pour **features**, **releases** et **hotfixes** permettent de gérer efficacement les évolutions et les corrections urgentes. Un point crucial du flux de travail est l'interdiction formelle de commettre directement sur les branches **main** et **development**, garantissant ainsi l'intégrité du code dans les phases finales et de développement. Enfin, pour le suivi du cycle de vie du code, un système est mis en place avec **Gitea Actions** et **SonarQube** pour l'analyse qualité du code, afin de prévenir les bugs et garantir un code propre. Voir le schéma d'exemple :



Concernant les **contraintes techniques**, l'un des défis majeurs est la gestion des **performances**, notamment la gestion des transferts volumineux et l'optimisation des **requêtes SQL** pour éviter les ralentissements. Pour le stockage des fichiers, une solution dédiée a été mise en place, avec des fichiers hébergés sur un **serveur dédié**, fourni par Ethan Le Touzic. La **sécurité** du projet est également une priorité : des mécanismes de protection contre les **injections SQL** et les attaques **brute-force** sont implémentés. De plus, les fichiers sont protégés par des mots de passe et des **liens temporaires**, garantissant ainsi leur confidentialité et leur accessibilité contrôlée. Enfin, les fichiers devront être supprimés lorsque la date d'expiration est dépassée.

Pour le suivi du projet, la méthode **ScrumBan** est utilisée. Cette approche hybride permet de combiner la rigueur de Scrum avec la flexibilité du Kanban. Des **réunions hebdomadaires** d'environ 30 minutes sont organisées pour faire le point sur l'avancée des tâches et ajuster les priorités. Les critères de la notion de "**Done**" sont clairement définis : chaque fonctionnalité doit être validée, testée et fusionnée avant de pouvoir être considérée comme terminée, ce qui garantit un développement de qualité tout au long du cycle de vie du projet.

# 4. DESCRIPTION FONCTIONNELLE DES BESOINS

## 1. Gestion des fichiers

### 1.1 Dépôt de fichiers

Le système de dépôt de fichiers doit permettre aux utilisateurs, qu'ils soient connectés ou non, d'envoyer des fichiers de manière simple, sécurisée et efficace. Les fonctionnalités diffèrent en fonction du statut de l'utilisateur.

Utilisateurs non connectés

Les utilisateurs non authentifiés peuvent déposer des fichiers avec les restrictions suivantes :

- **Taille maximale** : La taille totale des fichiers ne doit pas dépasser **5 Go**.
- **Formats acceptés** : Tous les formats de fichiers sont pris en charge, à l'exception des fichiers exécutables (tel que php, py, exe, bat...) pour des raisons de sécurité.
- **Génération de lien et de mot de passe** :
    - Une fois les fichiers déposés, le système génère automatiquement un **lien de téléchargement unique** et un **mot de passe aléatoire**.
    - Ces informations doivent être communiquées au destinataire pour accéder aux fichiers.
    - Une interface claire affichera ces détails avec un bouton de copie facilitant le partage ainsi que la date et l'heure de l'expiration.
- **Expiration et suppression automatique** :
    - Les fichiers restent disponibles pendant **7 jours** par défaut. Il est possible de modifier cette durée.
    - Dès lors que la date d'expiration des fichiers est dépassée, ils sont automatiquement supprimés du serveur.

Utilisateurs connectés

Les utilisateurs authentifiés disposent de fonctionnalités avancées leur permettant une meilleure gestion de leurs fichiers :

- **Taille maximale autorisée** : Les utilisateurs connectés peuvent téléverser jusqu'à **15 Go** de fichiers.
- **Organisation des fichiers** :
    - Possibilité de créer et de gérer des **dossiers** pour classer les fichiers.
    - Interface permettant de renommer ou supprimer les fichiers et dossiers.
- **Génération automatique de lien et mot de passe** :
    - Comme pour les utilisateurs non connectés, le système génère un **lien sécurisé** et un **mot de passe aléatoire** pour protéger l'accès aux fichiers.
    - L'utilisateur peut choisir de **modifier le mot de passe** manuellement.
- **Expiration et suppression automatique** :
    - Les fichiers restent disponibles pour une durée de **7 jours** par défaut. Il est également possible de modifier cette durée.
    - Avant expiration, une **notification** (interface) peut être envoyée à l'utilisateur pour l'informer que ses fichiers seront bientôt supprimés.
    - Après la période de 7 jours, les fichiers sont **définitivement supprimés** du serveur.

## 1.2 Restrictions et validations

Afin d'assurer un bon fonctionnement du système et d'éviter tout problème technique ou abus, plusieurs mécanismes de restriction et de validation sont mis en place lors du dépôt des fichiers :

- **Vérification du type de fichier avant dépôt** :
    - Le système s'assure que les fichiers déposés respectent les formats autorisés.
    - Les fichiers exécutables (tel que **php)** sont automatiquement refusés.
- **Vérification de l'espace disponible sur le serveur** :
    - Avant l'acceptation d'un fichier, le système vérifie qu'il y a suffisamment d'espace de stockage disponible.
    - En cas d'insuffisance, un message d'erreur informe l'utilisateur que le dépôt n'est pas possible.
- **Affichage d'un message d'erreur détaillé en cas d'échec du dépôt** :
    - En cas de problème (fichier non autorisé, taille trop grande, espace insuffisant, etc.), un message explicite est affiché à l'utilisateur.
    - Des recommandations peuvent être proposées pour résoudre l'erreur (ex. : réduire la taille des fichiers, réessayer ultérieurement).

## 2. Partage et gestion des liens

### 2.1 Génération des liens sécurisés

Le système garantira un accès sécurisé aux fichiers partagés grâce à la génération de liens protégés et temporaires.

- **Génération automatique d'un lien unique** :
  - Dès qu'un fichier ou un groupe de fichiers est déposé, le système génère automatiquement un **lien de téléchargement unique**.
  - Ce lien est propre à chaque dépôt et ne peut pas être deviné ou reproduit par un tiers.
- **Association d'un mot de passe aléatoire** :
  - Pour renforcer la sécurité, un **mot de passe aléatoire** est automatiquement attribué à chaque lien généré.
  - L'utilisateur doit transmettre ce mot de passe au destinataire pour lui permettre d'accéder aux fichiers.
  - L'utilisateur connecté a la possibilité de modifier le mot de passe attribué.
- **Expiration du lien en fonction du type d'utilisateur** :
  - Pour **tous les utilisateurs**, le lien expire **7 jours** après sa création.
  - Une fois expiré, le lien devient inactif et les fichiers associés sont supprimés définitivement du serveur.

### 2.2 Options avancées pour utilisateurs connectés

Les utilisateurs connectés bénéficient de plusieurs options avancées leur permettant de mieux contrôler les fichiers partagés :

- **Révocation anticipée d'un lien** :
  - Un utilisateur connecté peut **révoquer un lien de partage** avant son expiration.
  - Une fois révoqué, le lien devient immédiatement inactif et les fichiers ne sont plus accessibles.
- **Renouvellement du lien avec un nouveau mot de passe** :
  - L'utilisateur a la possibilité de **générer un nouveau lien avec un mot de passe différent** sans avoir à téléverser à nouveau les fichiers.
  - L'ancien lien devient alors obsolète, empêchant tout accès non autorisé aux fichiers précédemment partagés.
- **Journalisation des accès aux fichiers** :
  - Chaque accès aux fichiers est enregistré avec **la date, l'adresse IP et l'identifiant utilisateur (si disponible)**.
  - L'utilisateur connecté peut consulter ces logs pour vérifier qui a accédé à ses fichiers et à quel moment.

# 3. Téléchargement et accès aux fichiers

## 3.1 Processus de téléchargement

Le téléchargement des fichiers suit un processus structuré et sécurisé afin de garantir une expérience fluide pour l'utilisateur tout en assurant la confidentialité des données.

### 3.1.1 Accès aux fichiers via le lien et le mot de passe

- Une fois un fichier ou un dossier partagé, l'utilisateur recevant le lien doit **saisir le mot de passe** généré aléatoirement lors du dépôt.
- Si le mot de passe est correct, l'utilisateur accède à une interface lui permettant de télécharger les fichiers associés.
- Si le mot de passe est incorrect, un message d'erreur s'affiche et une **tentative erronée est comptabilisée**.

### 3.1.2 Interface utilisateur et aperçu des fichiers

L'interface de téléchargement est conçue pour être **simple, rapide et intuitive** :

- Affichage de la **liste des fichiers disponibles** avec leur nom, taille et date d'expiration.
- **Aperçu intégré** pour les fichiers compatibles :
    - Images (JPEG, PNG, GIF, WebP, SVG).
- Affichage de la **date d'expiration** des fichiers.

### 3.1.3 Options de téléchargement

L'utilisateur a plusieurs possibilités pour récupérer ses fichiers :

- **Téléchargement individuel** : chaque fichier peut être téléchargé séparément en cliquant sur le bouton dédié.

- **Téléchargement en lot** : l'utilisateur peut appuyer sur un bouton principal pour télécharger tous les fichiers qui seront alors compressés sous format **ZIP** avant d'être téléchargés.

### 3.1.4 Indicateur de progression du téléchargement

- Lors du téléchargement, une **barre de progression** s'affiche pour montrer l'avancement du transfert.

---

## 3.2 Sécurité et restrictions

### 3.2.1 Vérification du mot de passe avant l'accès aux fichiers

- Lorsqu'un utilisateur tente d'accéder à un fichier via un lien, il doit entrer le **mot de passe unique** généré lors du partage.

- Le mot de passe est chiffré et comparé de manière sécurisée pour éviter toute faille d'accès.

- Si le mot de passe est valide, l'accès aux fichiers est accordé et l'utilisateur peut procéder au téléchargement.

### 3.2.2 Limitation des tentatives erronées

Pour éviter les attaques par **brute-force**, un mécanisme de **sécurité renforcée** est mis en place :

- Après **5 tentatives incorrectes**, l'accès au lien est temporairement bloqué pendant **15 minutes**.
- Si 3 nouveaux échecs sont détectés après ce délai, un **blocage définitif** du lien est appliqué.
- L'émetteur des fichiers (si connecté) est informé du blocage et peut générer un nouveau lien.

### 3.2.3 Alerte en cas d'accès suspect (uniquement pour utilisateurs connectés)

- Si un utilisateur connecté partage un fichier et qu'un accès suspect est détecté (**nombre élevé de tentatives infructueuses**), une **alerte** lui est envoyée.
- L'alerte contient :
    - L'IP et l'emplacement approximatif du tentatif d'accès.
    - Le nombre de tentatives échouées.
    - Une option pour **révoquer immédiatement le lien de partage**.
- Un journal des accès est disponible dans l'espace utilisateur pour suivre toutes les connexions effectuées sur les fichiers partagés.

# 4. Gestion des utilisateurs

La gestion des utilisateurs est un élément important du projet, permettant aux personnes disposant d'un compte de bénéficier de fonctionnalités avancées et d'un espace personnel sécurisé pour gérer leurs fichiers. Cette section détaille l'ensemble des fonctionnalités liées à l'inscription, la connexion et l'espace utilisateur.

---

## 4.1 Inscription et connexion

### 4.1.1 Création de compte

L'inscription à la plateforme est nécessaire pour bénéficier d'un espace de stockage étendu (15 Go) et de fonctionnalités avancées (organisation des fichiers, historique, renouvellement de liens, etc.).

- L'utilisateur doit renseigner les informations suivantes :
    - **Adresse e-mail valide** (utilisée pour l'authentification et les notifications).
    - **Mot de passe sécurisé** respectant les critères suivants :
        - Minimum **12 caractères**.
        - Contient **au moins une lettre majuscule** et une **lettre minuscule**.
        - Contient **au moins un chiffre**.
        - Contient **au moins un caractère spécial** (@, #, $, etc.).

#### 4.1.2 Connexion au compte

Pour accéder à son compte, l'utilisateur doit fournir :

- **Son adresse e-mail**.
- **Son mot de passe**.

Après saisie des informations :

- Le système vérifie si les identifiants sont corrects.
- En cas d'échec :
    - Un message d'erreur indique qu'il y a une erreur dans les informations saisies.
    - Après **5 tentatives infructueuses**, le compte est **temporairement bloqué** pendant 15 minutes pour éviter les attaques par force brute.

#### 4.1.3 Double authentification (2FA) - Optionnelle

L'utilisateur peut activer la **double authentification (2FA)** pour sécuriser davantage son compte.

- Une fois activée, à chaque connexion :
    - L'utilisateur saisit son e-mail et mot de passe.
    - Un code temporaire (OTP) est envoyé par **e-mail**.
    - L'utilisateur doit entrer ce code pour finaliser la connexion.

#### 4.1.4 Réinitialisation du mot de passe

En cas d'oubli du mot de passe, l'utilisateur peut demander sa réinitialisation :

1. **Saisie de l'e-mail associé au compte**.
2. **Envoi d'un e-mail sécurisé** contenant un lien de réinitialisation (valable 15 minutes).
3. **L'utilisateur clique sur le lien** et saisit un **nouveau mot de passe** respectant les règles de sécurité.
4. **Confirmation et connexion possible avec le nouveau mot de passe**.
5. En cas de plusieurs demandes de réinitialisation en peu de temps, un **blocage temporaire** est appliqué pour éviter les abus.

---

### 4.2 Espace utilisateur

L'espace utilisateur permet aux membres connectés de gérer leurs fichiers de manière sécurisée et organisée.

#### 4.2.1 Tableau de bord

Le tableau de bord est la **page principale** après connexion et affiche :

- **La liste des fichiers déposés**, classés par date d'ajout (du plus récent au plus ancien).
- **L'état des fichiers**, indiquant si :
    - Le fichier est encore **actif** (nombre de jours restants avant suppression).
    - Le fichier a déjà été **téléchargé** ou non.
    - Le lien généré est encore **valide** ou **expiré**.
- Un **moteur de recherche** permet de filtrer les fichiers par **nom**, **date de dépôt** ou **état**.
- Un **système de tags ou catégories** permet de mieux organiser les fichiers.

#### 4.2.2 Suppression manuelle d’un fichier avant expiration

L’utilisateur peut supprimer un fichier avant la suppression automatique :

- Un bouton **"Supprimer"** est disponible pour chaque fichier.
- Une **confirmation** est demandée avant suppression définitive.
- Une **notification** informe l’utilisateur que la suppression a bien été effectuée.
- Si un fichier supprimé était encore lié à un partage, son **lien devient immédiatement invalide**.

#### 4.2.3 Historique des fichiers partagés et téléchargés

L’historique permet de suivre les activités liées aux fichiers :

- **Fichiers partagés** :
    - Date et heure de création du lien.
    - Nombre de fois où le fichier a été téléchargé.
    - Dernière date de téléchargement.
    - Statut du lien (**actif** ou **expiré**).
- **Fichiers téléchargés** :
    - Liste des fichiers téléchargés par l’utilisateur (depuis son compte ou via un lien reçu).
    - Informations sur la date et l’heure du téléchargement.
- **Export possible** des données en format **CSV** pour archivage.

## 5. Gestion des fichiers sur le serveur

L'infrastructure de stockage et de gestion des fichiers sur Send-It repose sur un **serveur dédié** sécurisé. L'optimisation des performances et la suppression automatique des fichiers expirés sont des éléments essentiels pour garantir une utilisation fluide et éviter l'accumulation inutile de données.

### 5.1 Stockage et suppression automatique

Stockage des fichiers

- Tous les fichiers uploadés par les utilisateurs sont stockés sur un **serveur dédié** distinct du serveur applicatif afin d'optimiser la charge et d'améliorer la scalabilité du système.

Les fichiers sont organisés dans une arborescence structurée sous la forme suivante :

```
/storage/
    ├── {upload_id}/
        ├── fichier1.pdf
        ├── fichier2.jpg
```

- Chaque fichier est stocké avec un **identifiant unique (UUID)** pour éviter toute collision de nom et garantir l'unicité.

**Suppression automatique des fichiers expirés**

- Un **cron job** est exécuté toutes les heures pour analyser les fichiers stockés et détecter ceux dont la période de rétention est dépassée.
- Les fichiers seront supprimés **automatiquement après 7 jours**, qu'ils aient été téléchargés ou non.

**Logs et traçabilité des suppressions**

- Chaque suppression de fichier est **journalisée dans un fichier de logs** pour assurer un suivi et faciliter d'éventuelles analyses en cas d'incident.

Format du log :

```
[2025-03-28 14:05:32] Suppression fichier: {fichier_UUID} | Utilisateur: {user_id} | Raison: Expiration
```

- Une table en base de données stocke les métadonnées des fichiers supprimés pendant **30 jours** avant leur purge définitive.

### 5.2 Optimisation et performance

**Compression des fichiers**

- Une **compression automatique** est appliquée aux fichiers volumineux pour **réduire l'espace disque utilisé** sans altérer la qualité.
- Types de compression appliqués en fonction du format :
    - **Images** (JPEG, PNG) → Compression sans perte via `mozjpeg` et `pngquant`.
    - **Documents PDF** → Optimisation via `Ghostscript` pour réduire la taille des fichiers.

## 6. Sécurité et protection des données

La sécurité des fichiers et des données des utilisateurs est une priorité absolue dans le cadre du projet **Send-It**. Une série de mesures avancées est mise en place pour assurer la protection des fichiers stockés, la confidentialité des liens de partage, et la sécurisation des accès contre les attaques malveillantes.

---

### 6.1 Chiffrement et sécurité des fichiers

Chiffrement des liens et mots de passe générés

- Chaque lien de partage généré contient un **jeton unique** encodé en base64 et signé numériquement.
- Les mots de passe associés aux fichiers sont **hachés avec bcrypt** avant d'être stockés en base de données, empêchant leur récupération même en cas de fuite.
- Pour protéger les liens de partage contre des attaques de type **brute-force**, la plateforme impose une **restriction stricte sur le nombre de tentatives d'accès**, déclenchant un verrouillage temporaire après plusieurs erreurs.

---

## 6.2 Protection contre les attaques

Détection et blocage des attaques brute-force

- Un **système de limitation de requêtes** est mis en place pour prévenir les attaques visant à tester plusieurs combinaisons de mots de passe ou de liens.

- Après **5 tentatives erronées sur un fichier protégé par mot de passe**, l'accès est temporairement **bloqué pendant 15 minutes** pour cette adresse IP.

- Pour les utilisateurs connectés, si plusieurs tentatives de connexion échouent, l'accès au compte est **verrouillé** et un e-mail de vérification est envoyé.

- De plus, si un utilisateur (connecté ou non) tente de télécharger plusieurs fois en peu de temps, son accès sera refusé pendant 30 minutes.

Protection contre les injections SQL et XSS

- Toutes les requêtes SQL sont préparées et paramétrées pour éviter les **attaques par injection SQL**.

- Un système de validation et d'échappement des entrées est appliqué sur tous les formulaires pour empêcher les **attaques XSS (Cross-Site Scripting)**.

- Les en-têtes de sécurité sont configurés correctement :

    - **Content-Security-Policy (CSP)** : empêche l'exécution de scripts malveillants.

    - **X-Frame-Options: DENY** : empêche l'inclusion du site dans un iframe malveillant.

    - **X-XSS-Protection: 1; mode=block** : active la protection anti-XSS des navigateurs.

Mise en place d’un système de logs et alertes pour surveiller les activités suspectes

- Tous les accès aux fichiers, tentatives de connexion et actions sensibles sont **journalisés** dans une base de données sécurisée.

- Un **système d’alerte en temps réel** est déployé pour détecter les comportements suspects :

    - Tentatives répétées d’accès à un fichier ou un compte utilisateur.

    - Connexions depuis des adresses IP inhabituelles ou suspectes.

    - Téléchargement massif de fichiers en un court laps de temps.

- Des **logs détaillés** sont générés et stockés en lecture seule pendant une durée de **6 mois** avant suppression automatique.

- Un tableau de bord d’administration permet d’afficher des **statistiques de sécurité** et de **bloquer des IP malveillantes** si nécessaire.

# 7. Expérience utilisateur et interface

L'interface utilisateur (UI) et l'expérience utilisateur (UX) doivent garantir une utilisation fluide, intuitive et agréable, tout en maximisant l'accessibilité et la performance.

## 7.1 Design et ergonomie

L'objectif principal du design est de **proposer une interface claire, intuitive et moderne** qui s'adapte aux besoins des utilisateurs tout en garantissant une navigation rapide et efficace.

Interface responsive et accessible sur mobile

L'application **doit être totalement responsive**, c'est-à-dire parfaitement utilisable sur **ordinateurs, tablettes et smartphones**, quelle que soit la taille de l'écran.

- **Approche Mobile First** : Développement prioritaire pour mobile, avec un affichage optimisé pour les petits écrans avant d'être adapté aux écrans plus larges.
- **Gestion fluide du redimensionnement** : Utilisation de **Tailwind CSS** pour garantir un affichage optimal sur toutes les résolutions (breakpoints bien définis : sm, md, lg, xl).
- **Éléments tactiles optimisés** : Boutons et liens suffisamment grands pour être facilement cliquables sur mobile (minimum 44x44 px).
- **Navigation simplifiée** : Menu accessible et déroulant sur mobile, avec icônes bien définies pour une compréhension immédiate.
- **Performances optimisées** : Chargement rapide des pages et des fichiers, évitant les latences et améliorant l'expérience utilisateur sur les connexions mobiles.

Thème clair/sombre pour le confort visuel

L'interface doit proposer **deux modes d'affichage** pour s'adapter aux préférences et au confort visuel de l'utilisateur :

- **Mode clair** : Fond blanc avec textes sombres, idéal pour une utilisation en journée.
- **Mode sombre** : Fond noir ou gris foncé avec textes clairs, réduit la fatigue oculaire lors d'une utilisation prolongée, notamment la nuit.

Barre de recherche pour retrouver des fichiers rapidement

Afin d'améliorer l'ergonomie et d'offrir une navigation efficace, une **barre de recherche avancée** est intégrée :

- **Recherche instantanée** : Suggestions affichées en temps réel dès la saisie d'un mot-clé.
- **Filtrage par type de fichier** : Possibilité de restreindre la recherche aux images, documents, vidéos, etc.
- **Filtrage par date d'envoi** : Recherche des fichiers déposés sur une période donnée.
- **Indicateurs visuels** : Affichage de **balises** (tags) sur les fichiers pour mieux les distinguer (ex. : "Partagé", "En attente de téléchargement", "Expiré").
- **Icônes et prévisualisation** : Affichage d'un aperçu (icône ou miniature) des fichiers dans les résultats.

## 7.2 Notifications et suivi

Pour assurer un suivi efficace des fichiers partagés et optimiser l'expérience utilisateur, un **système de notifications et d'alertes** est mis en place.

Notification e-mail lors du partage d'un fichier (optionnelle)

Lorsqu'un fichier est partagé via un lien, l'expéditeur peut **choisir d'envoyer une notification par e-mail** au destinataire :

- **Contenu de l'e-mail** :
    - Nom de l'expéditeur et e-mail associé.
    - Liste des fichiers partagés et leur taille.
    - Lien de téléchargement et mot de passe associé.
    - Date d'expiration des fichiers.
- **Personnalisation** :
    - Possibilité d'ajouter un **message personnalisé** à l'e-mail.
    - Option d'activer/désactiver cette fonctionnalité lors du partage.
- **Sécurité** :
    - Lien chiffré et unique généré dans l'e-mail pour éviter toute interception.
    - **Protection anti-spam** pour éviter l'envoi d'e-mails non sollicités.
- **Gestion des envois** :
    - Envoi des e-mails via un service SMTP sécurisé.

Avertissement avant l’expiration d’un fichier

Afin d’éviter la perte accidentelle de fichiers, un **système d’alerte est mis en place** pour informer l’utilisateur avant qu’un fichier n’expire :

- **Méthodes d’alerte** :
    - **Bannière d’alerte sur l’interface** : Affichage d’un message coloré indiquant que les fichiers expireront bientôt.
    - **Compte à rebours dynamique** sur la page de téléchargement.
- **Options avancées pour utilisateurs connectés** :
    - Possibilité de **prolonger** la durée de stockage des fichiers (dans la limite des 7 jours).
    - Option de **re-génération d’un lien** avec une nouvelle durée de validité.
    - Journalisation des rappels envoyés dans l’espace utilisateur.

# 5. ENVELOPPE BUDGÉTAIRE

Le développement du projet **Send-It** nécessitera **80 heures de travail**, réparties entre **deux développeurs**, soit **40 heures chacun**.

## Coût de la main-d'œuvre

Le salaire minimum en France étant de **11,88 € brut/heure**, le coût total de développement peut être estimé comme suit :

$$80\,\text{heures} \times 11,88\,€/\text{h} = 950,40€$$

## 6. Délais

Le projet **Send-It** suit un calendrier défini avec une **deadline finale fixée au 22 août** et une **présentation orale prévue le 11 juillet**. Pour assurer une gestion efficace du temps et garantir la livraison dans les délais, un **planning détaillé** avec des livrables intermédiaires a été établi.

---

### Planning prévisionnel

| Phase | Tâches principales | Début | Fin | Livrables |
|---|---|---|---|---|
| **Phase 1 : Analyse & Conception** | Rédaction du cahier des charges<br><br>Choix des technologies<br><br>Planification du développement | 01/04 | 14/04 | Cahier des charges validé |
| **Phase 2 : Développement Backend** | Création de la base de données PostgreSQL<br><br>Développement des API en PHP<br><br>Sécurisation des endpoints | 15/04 | 10/05 | API fonctionnelle et sécurisée |

| | | | | |
|---|---|---|---|---|
| **Phase 3 : Développement Frontend** | Création de l'interface utilisateur<br><br>Implémentation des fonctionnalités<br><br>Tests unitaires | 11/05 | 05/06 | Interface web opérationnelle |
| **Phase 4 : Tests & Optimisation** | Tests fonctionnels et de performance<br><br>Correction des bugs<br><br>Amélioration UX/UI | 06/06 | 30/06 | Version stable testée |
| **Phase 5 : Préparation de l'oral** | Rédaction du support de présentation<br><br>Préparation de la démonstration | 01/07 | 10/07 | Présentation prête |
| **Phase 6 : Corrections et Finalisation** | Intégration des retours de l'oral<br><br>Derniers ajustements et tests | 12/07 | 22/08 | Version finale |

## Diagramme de Gantt prévisionnel

Un **diagramme de Gantt prévisionnel** permet de visualiser la répartition des tâches dans le temps et d'anticiper les risques de retard. Il se situe sur Jira.