Cahier des charges



Thomas Genin - Nicola Piemontese - Valentin Tournier - Thomas Violent

CPE Lyon – 5ICS

Table des matières

[I Définition du projet 2](#_Toc150788716)

[II Objectifs du projet 2](#_Toc150788717)

[III Périmètre 2](#_Toc150788718)

[IV Rôles 2](#_Toc150788719)

[V Cas d’utilisation 3](#_Toc150788720)

[VI Spécifications fonctionnels 3](#_Toc150788721)

[VII Contraintes techniques 3](#_Toc150788722)

[VIII Planning prévisionnel 4](#_Toc150788723)

[IV Maquettes 4](#_Toc150788724)

## I Définition du projet

Le projet s’inscrit dans une volonté d’étendre l’offre de détection des mails malveillants en proposant un outil similaire au fonctionnement de VirusTotal qui permettrait entre autres, l’analyse de mail sur demande. L’idée a émergé sur un besoin, celui pour un analyste SOC de détecter simplement et automatiquement le caractère suspicieux (ou non d’un mail) au travers de différents facteurs.

## II Objectifs du projet

Nous voulons développer une solution permettant une analyse complète des emails. Celle-ci permettra d’avoir une analyse approfondie afin de détecter les emails malveillants à partir de plusieurs indicateurs.

Les utilisateurs pourront soumettre des emails à la solution et un score de risque sera calculé afin de définir si celui est malveillant.

## III Périmètre

Cette application se destine principalement au professionnel de la cybersécurité, notamment les analystes SOC.

En effet, les analystes doivent régulièrement étudier des emails tagués comme potentiellement malveillant par les systèmes de détections automatisée ou signalés par les utilisateurs. Le site permet de réaliser une analyse approfondie de manière automatique.

## IV Rôles

## V Cas d’utilisation

* Les utilisateurs doivent pouvoir uploader sur l’interface web un email à analyser
* Les utilisateurs doivent pouvoir visualiser le résultat de l’analyse
* Les utilisateurs doivent pouvoir lancer l’analyse d’un email via l’API
* Les utilisateurs doivent pouvoir visualiser l’historique et les résultats des précédentes analyses.

## VI Spécifications fonctionnels

* Le site web doit être en anglais
* Les utilisateurs
  + Doivent pour s’authentifier
  + Voir leur mails/fichier envoyés par le passé
  + Soumettre un mail ou un fichier qui sera scanné par l’appli
  + Obtenir un retour de l’application, qui se traduira par un score de fiabilité concernant les mails envoyés
* Analyse des mails
  + Les métadonnées des emails doivent être vérifiés à chaque scan, incluant :
    - La réputation du nom de domaine et de l’adresse IP.
    - Certains headers spécifiques (SPF, DKIM)
  + Le contenu des emails doit passer une batterie de tests :
    - Fautes d’orthographes
    - Caractères dans une police non conventionnel
    - Liens web et toutes autres url présentes
      * Vérifier la réputation du domaine et l’adresse IP derrière le nom de domaine,
      * Vérifier si le lien écrit est bien le lien voulu (par exemple, si un site web amène vers google.fr, s’assurer qu’il n’est pas écrit googIe.fr)
    - Similaire à d’ancien mail reconnus comme spam
    - Typosquattage (par exemple un l et un I)
* Analyse des fichiers
  + La potentielle présence de code malveillant doit être vérifiée, incluant :
    - Une analyse complète (Virus total)
    - L’ouverture des fichiers dans des sandbox dédiées (si applicable)
  + Le type de document doit être vérifié
  + Si le fichier est un contenant (un fichier zip par exemple), le contenu de chaque fichier doit être vérifié.

## VII Contraintes techniques

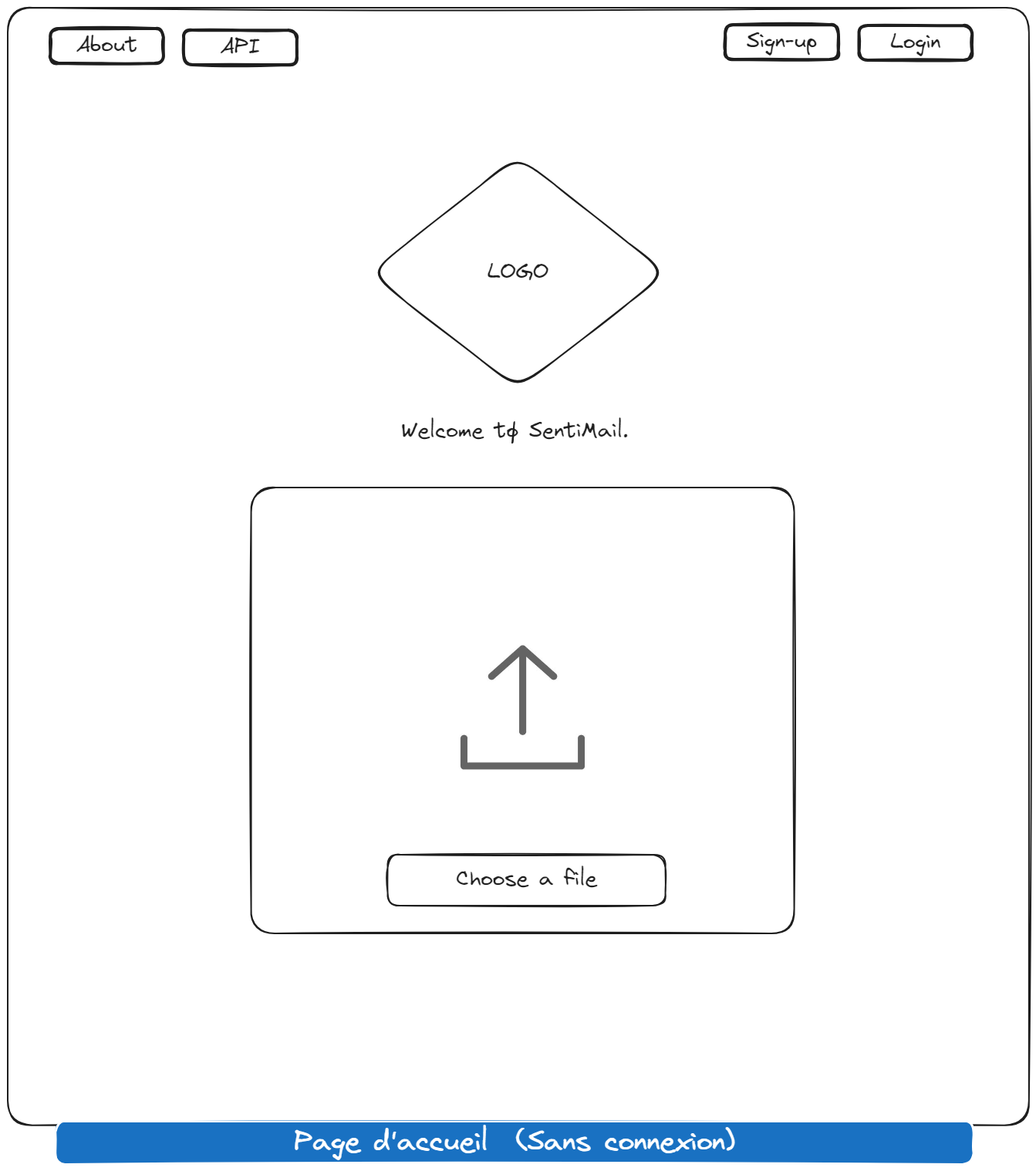
* La solution doit utiliser une base de données relationnelle ou noSQL.
* Le stockage des mails et des pièces jointes doit se faire sur un bucket S3 ou équivalent.
* La solution doit pouvoir être déployé sur une plateforme cloud de type Kubernetes
* La solution doit pouvoir supporter une certaine montée en charge (~50 requêtes par secondes, burst éventuel à 200)
* La solution doit utiliser une architecture de type micro-service afin de faciliter l’évolution et la maintenance des différents services. Une tolérance aux pannes est attendu sur les micro-services.

## VIII Planning prévisionnel

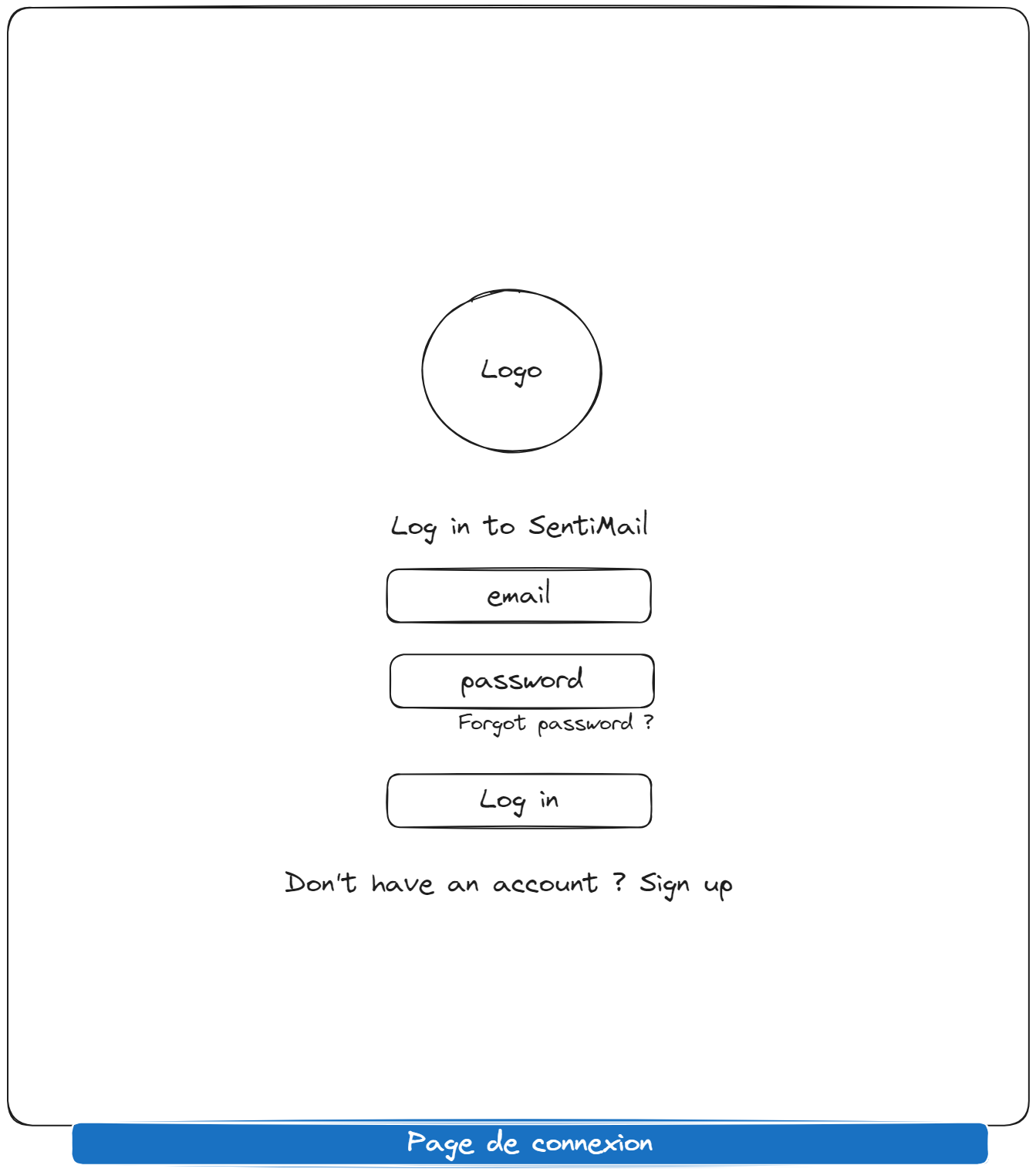


## IV Maquettes

**Page N°1 : Page d’accueil**

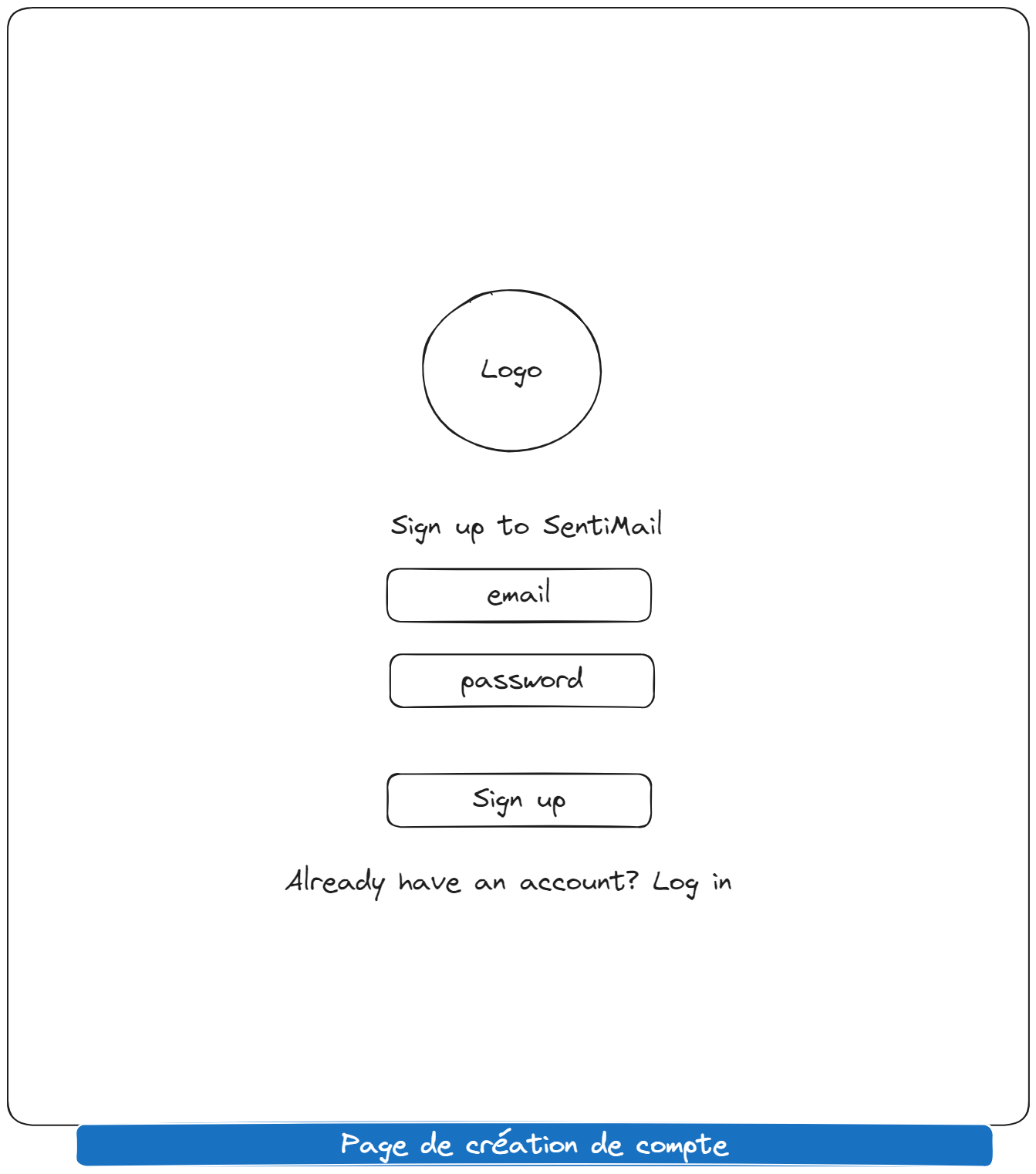


**Page N°2 – Page de connexion**



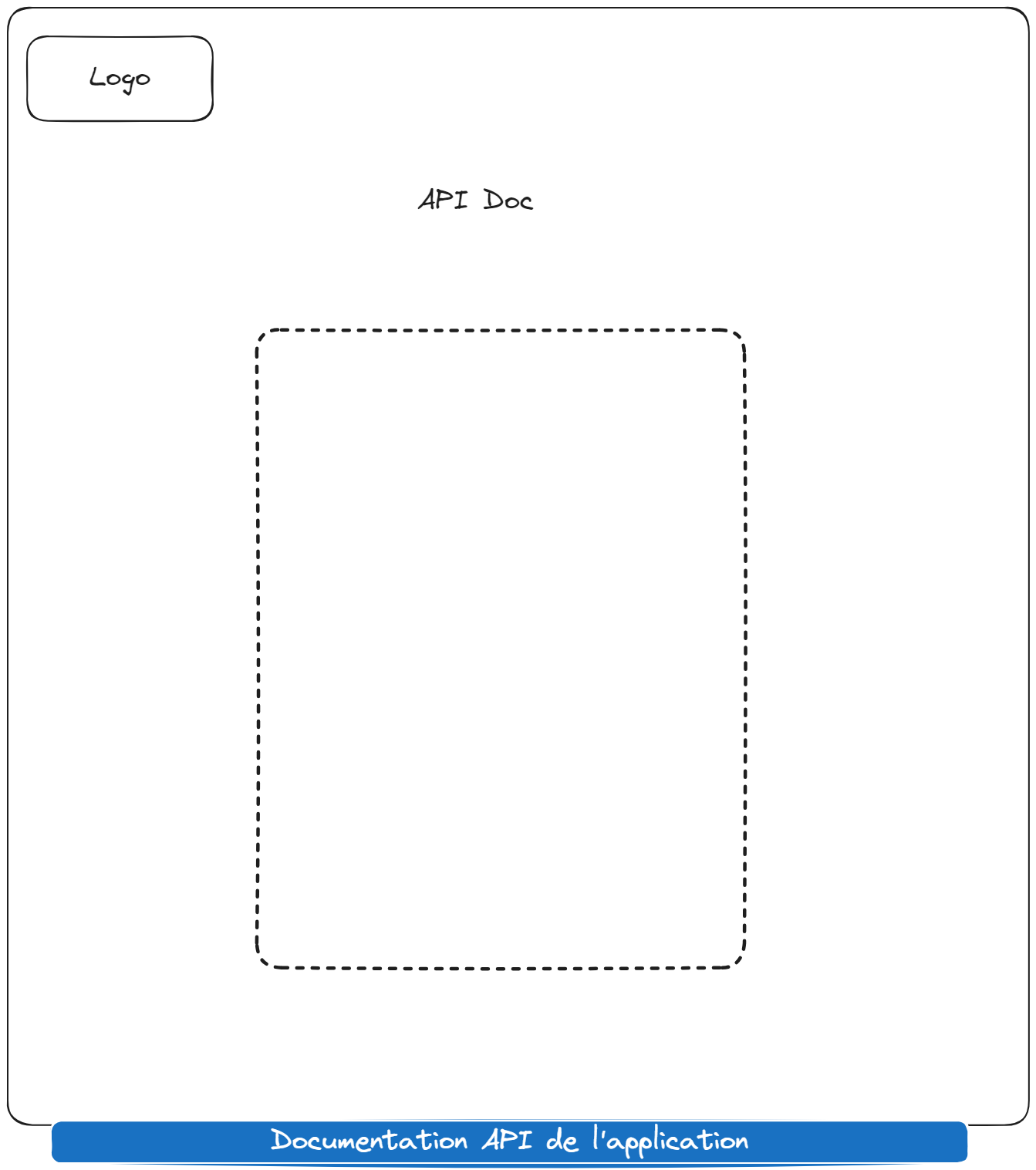


**Page N°3 – Page de création de compte**



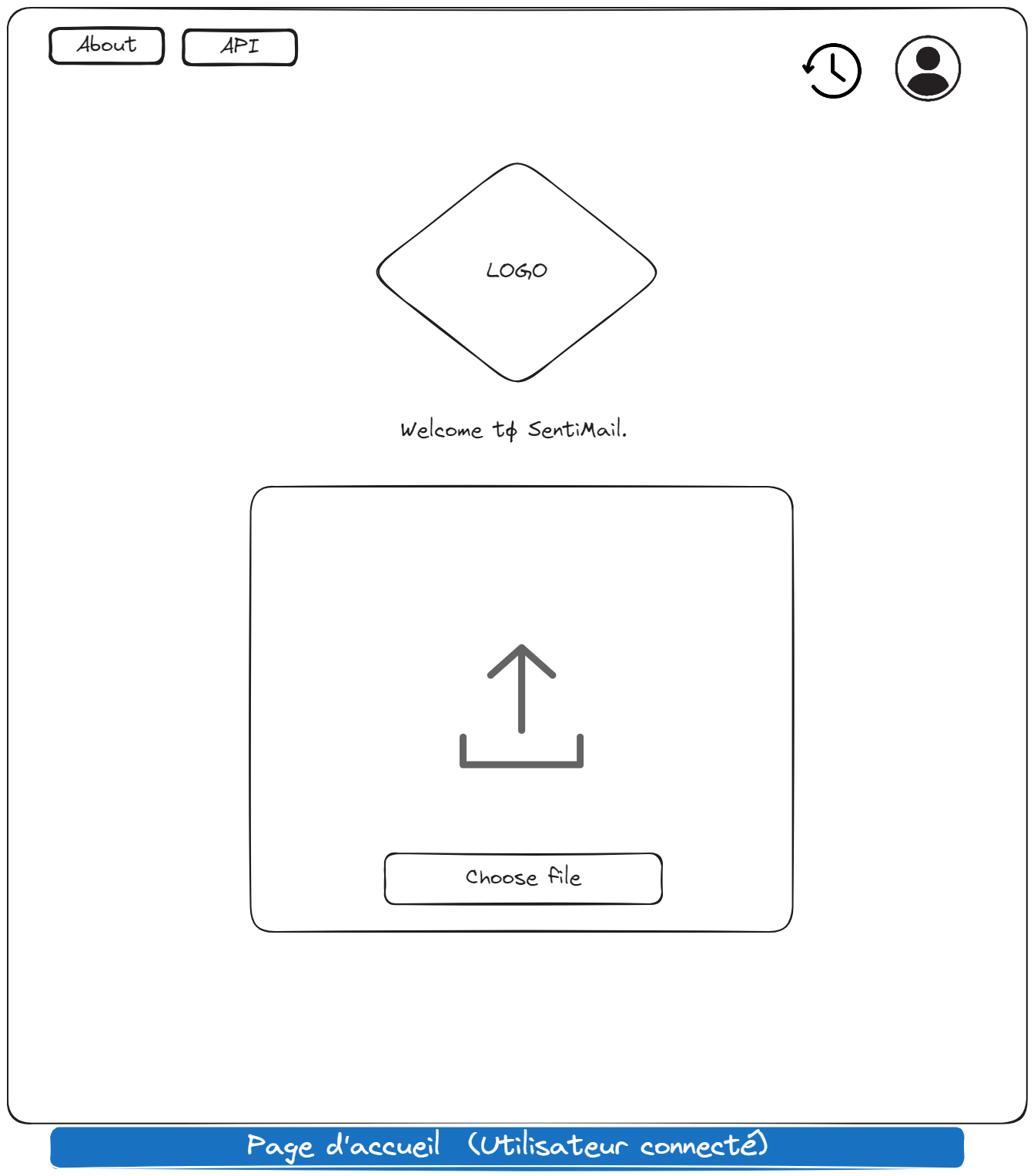
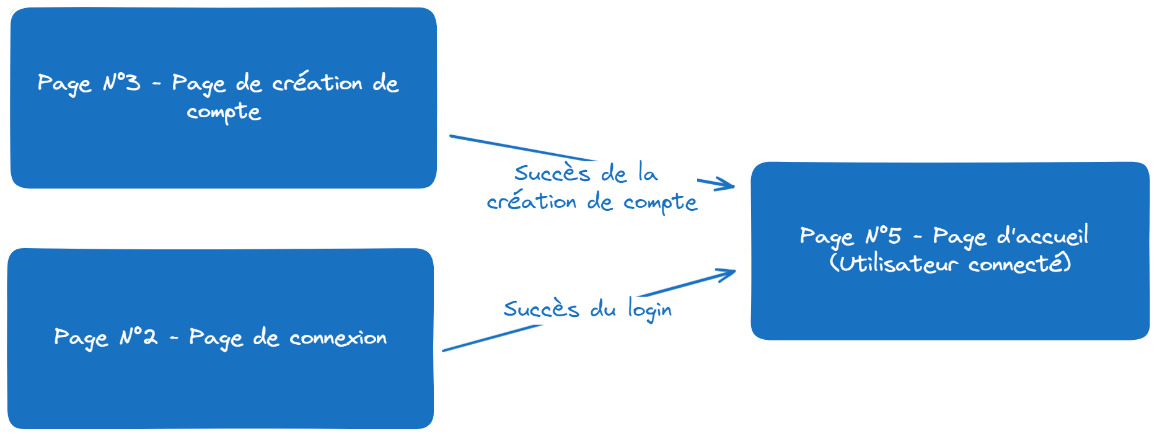


**Page N°4 – Page de documentation d’API**

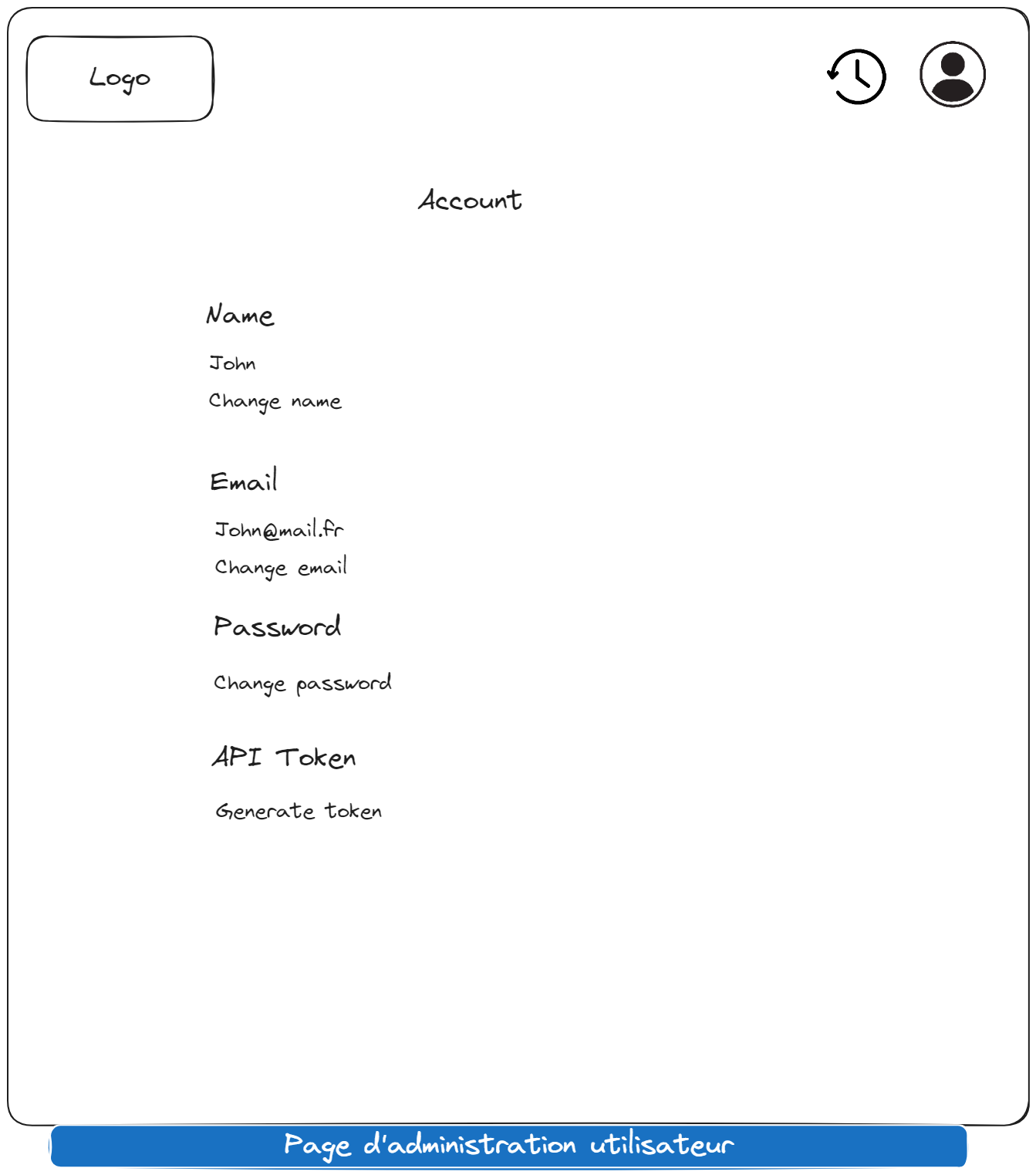


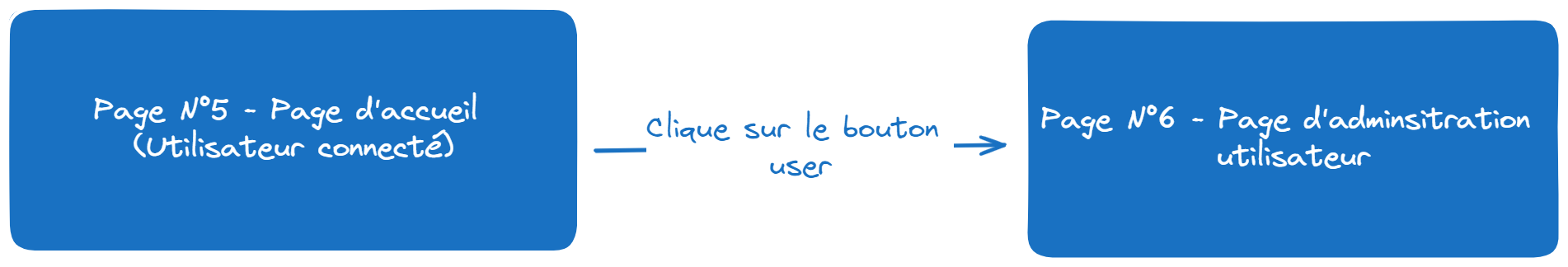


**Page N°5 – Page d’accueil (Utilisateur connecté)**

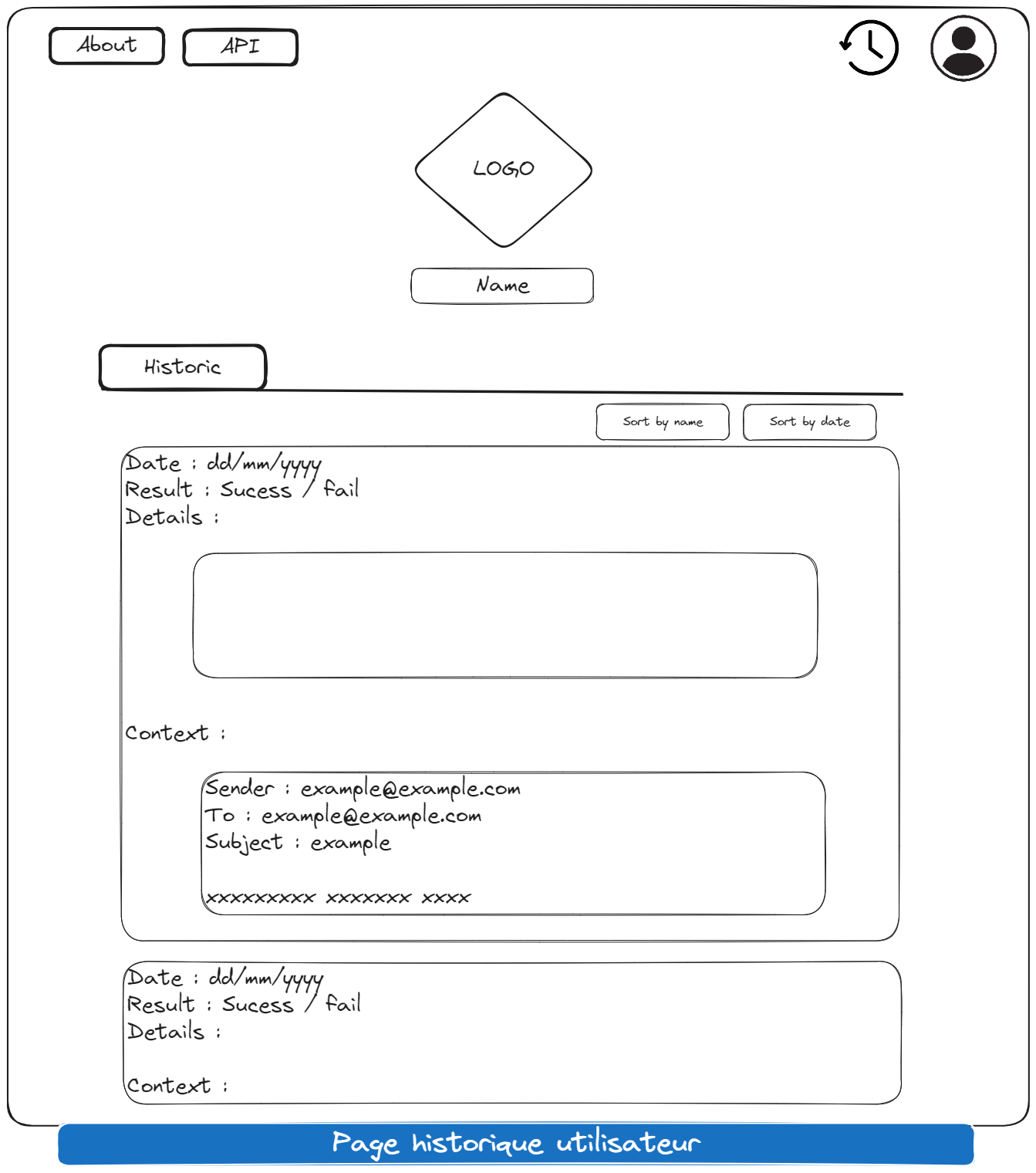


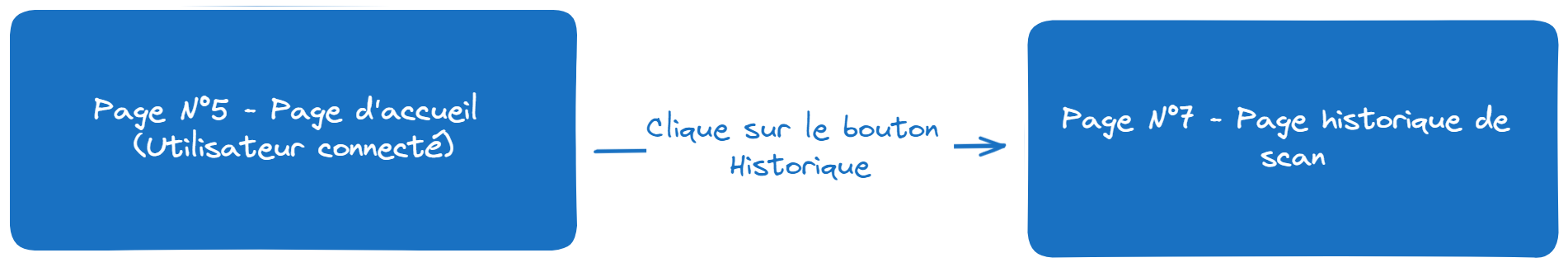
**Page N°6 – Page d’administration utilisateur**



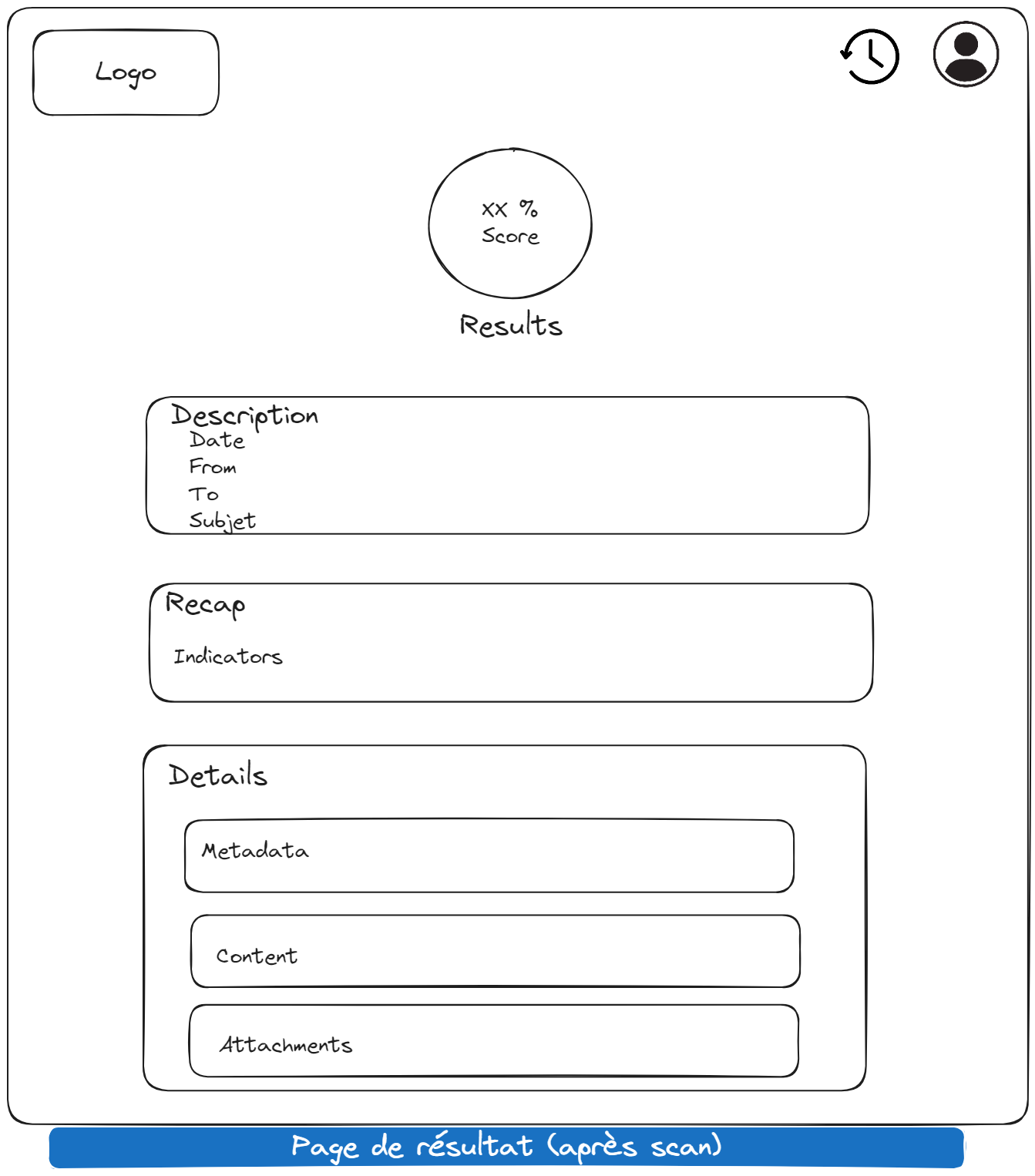
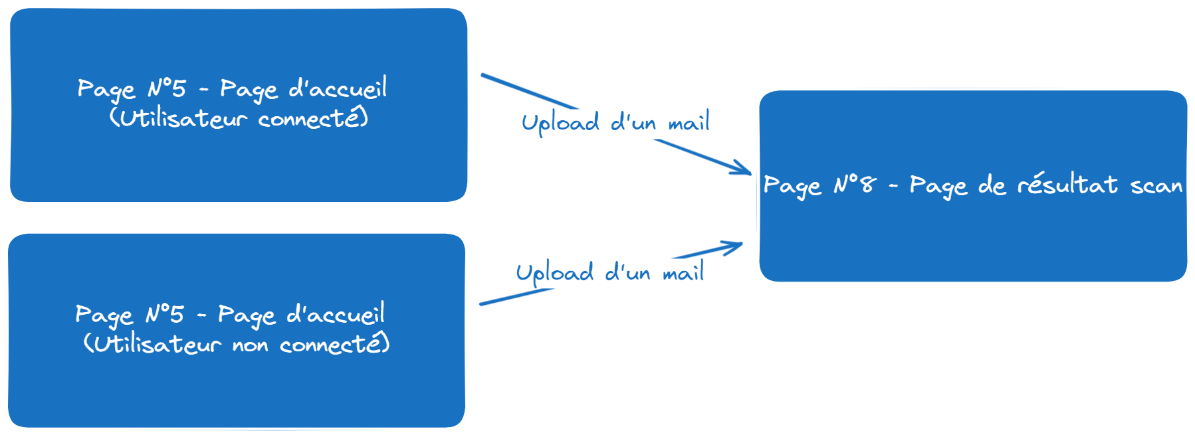


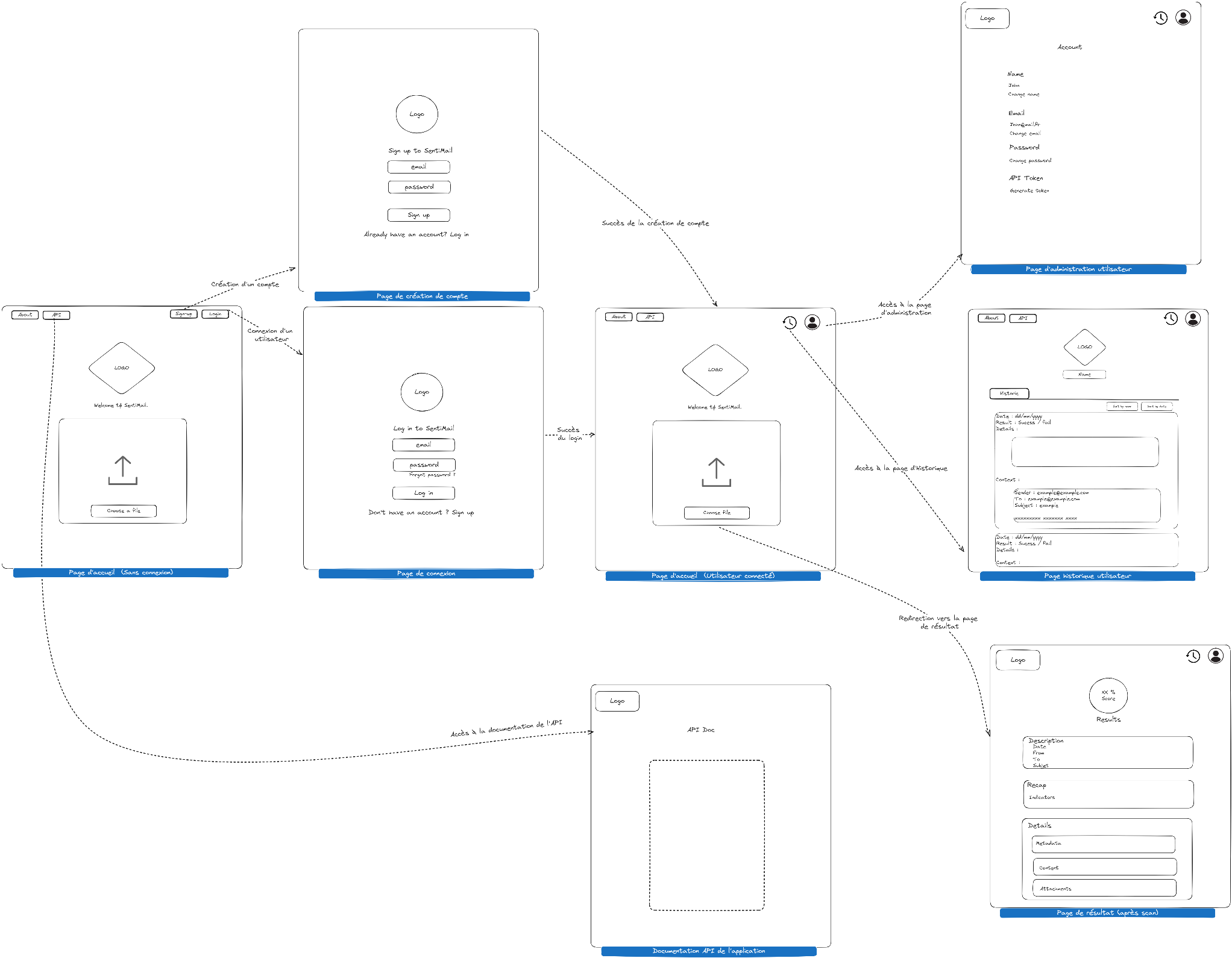
**Page N°7 – Page d’historique scan**





**Page N°8 – Page de résultat scan**





**Maquette Complete de l’application**