Le droit d’accès à leurs données personnelles sont permis en actionnant le burger et sélectionnant les paramètres ce qui amène sur la page contenant leurs données personnelles.

Le droit de rectification des données inexactes est respectée car on peut apercevoir un bouton modifie qui permet à l’utilisateur de modifier toutes les informations le concernant.

Pour le droit de l’effacement de leurs données ( droit à l’oubli ) nous avons une icône poubelle qui permet à l’utilisateur de supprimer son compte avec une confirmation afin d’être conforme au RGPD.

Afin de sécuriser les demandes de l’utilisateur et de les rendre simple et efficace, nous avons choisi de valider chaque choix de l’utilisateur par une confirmation avec utilisation de mot de passe ( nous aurions pu aussi utiliser une authentification par e-mail/numéro de téléphone.

Afin de minimiser la collecte et l’utilisation des données perso, nous avons mis en place sur le côté au droit de la page paramètre, un icone partager où l’utilisateur doit premièrement choisir quelle donnée personnelle il souhaite partager, ce qui permet de minimiser la collecte des données personnelles.

Afin d’assurer les mesures de sécurité intégré pour protéger les données perso des utilisateurs, nous demandons une confirmation par double authentification à la suite de l’appuie de l’icone partager si le code appuyer est bon cela nous affichera le choix des données partager sinon nous dirons à l’utilisateur qu’il reste 2 essais de mdp suite à ses essais le compte sera déconnecté toute en envoyant un mail au compte utilisateur pour le prévenir de ce qui est arrivé.

Nous avons choisi afin de protéger l’utilisateur la double authentification pour respecter le RGPD et afin d’amener la meilleure sécurité au client.

Exercice 3 : A)

Afin d’implémenter un système de login sur notre app web, nous allons choisir la solution monologue pour langage PHP.

Afin d’ajouter monologue, nous allons simplement exécuter la commande dans notre terminale composer require monologue /monologue [il faut impérativement avoir composer pour pouvoir effectuer cette commande].

Nous devons maintenant configurer monologue afin de capturer les info requises tels que les requête HTTP, les tentatives de connexion, les modifications de données et les actions critiques des utilisateurs.

Pour faire cela, il va falloir créer plusieurs canaux de login et des gestionnaires selon le type de log.

Ces canaux filtreront les logins selon leurs types et selon leurs types ils iront dans différents gestionnaires.

Afin de stocker de manière sécurisée les logs nous configurons la solution log stash qui nous permettra d’écouter les logs entrants sur un port spécifique qui nous permettra de filtrer et formater ses logs entrants et seront envoyer vers élastique search. Il faudra bien configurer les indexes et les mappings appropriées pour stocker les logs.

Pour surveiller les logs et détecter les modèles anormaux puisque nous sommes sur windows, nous allons utiliser l’outil appelé : window advanced firewall security qui permet de surveiller les logs du parefeu windows et de prendre des mesures en réponse aux activités suspectes.

B)

Nous devons développer un script d’analyse des logs qui a comme objectif :

* lire et analyse les logs stockées
* identifier les accès anormaux, les tentative de connexion échoué en grand nombre ou accéder aux données sensible
* Évaluer l’étendu de la violation
* Classer l’incident au fonction de la sévérité