DOCUMENTATION TECHNIQUE

Table des matières

[Spécifications Techniques 2](#_Toc117460306)

[Diagramme de cas d’utilisation : 3](#_Toc117460307)

[Diagrammes de séquences : 4](#_Toc117460308)

[DS1 Ajout d’une structure : 4](#_Toc117460309)

[DS2 Modification des droits : 5](#_Toc117460310)

[DS3 Sélection des partenaires actifs : 7](#_Toc117460311)

[Diagramme de classe : 8](#_Toc117460312)

[Mesures de sécurité : 8](#_Toc117460313)

[L’authentification : 8](#_Toc117460314)

[Les autorisations : 8](#_Toc117460315)

[Protection contre les injections : 8](#_Toc117460316)

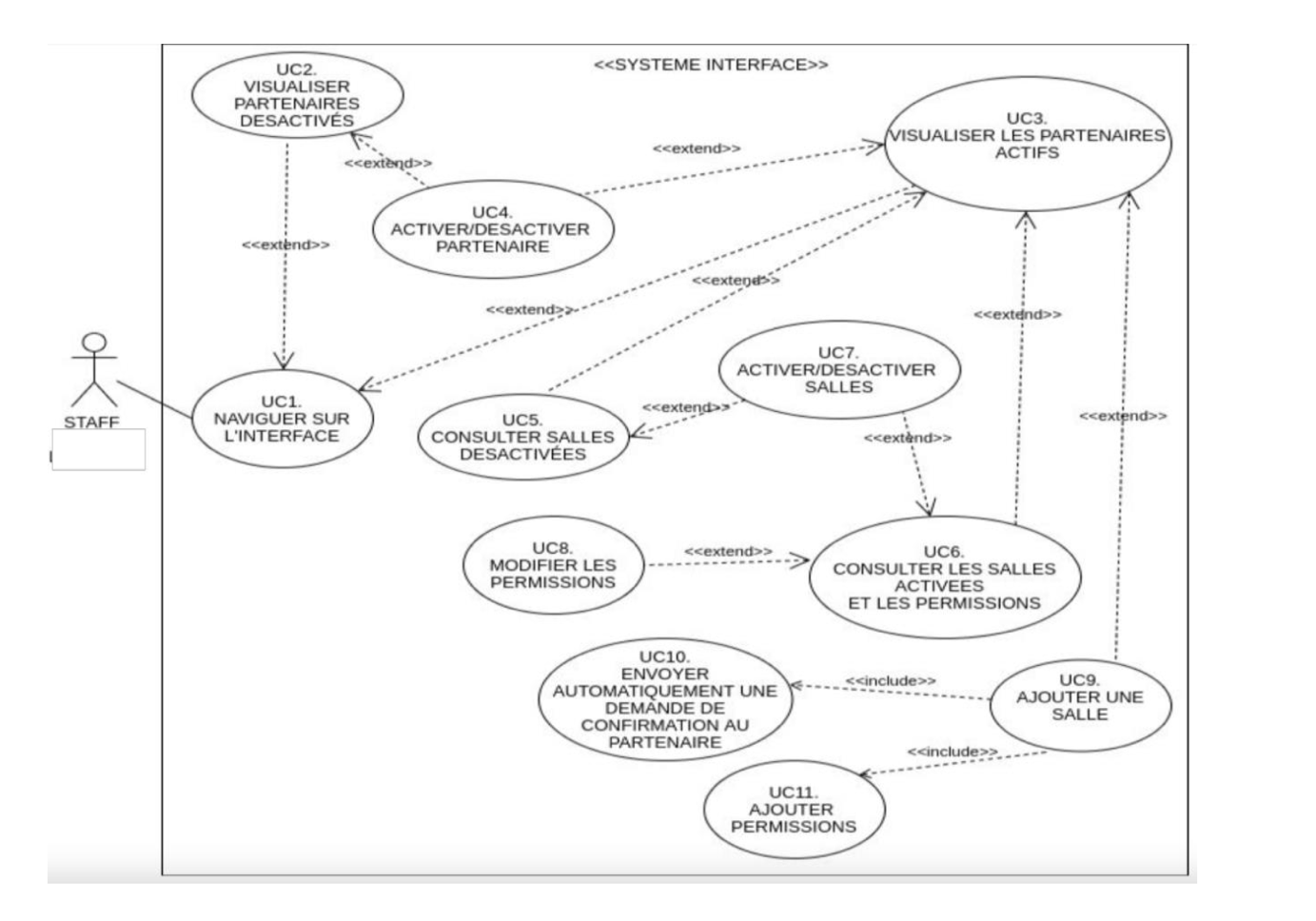
[Les formulaires : 9](#_Toc117460317)

[Le protocole HTTPS : 9](#_Toc117460318)

# Spécifications Techniques

|  |  |
| --- | --- |
| Local | Production |
| Serveur :  MAMP   * MySQL (5.7.34) * APACHE * PHP (8.0.8) | Serveur :  02switch   * MariaDB (10.3.36) * Apache * PHP(8.0.24) |
| Front :   * HTML 5 * CSS 3 * Bootstrap 5 * Jquery 3.2 | |
| Back :   * Composer (2.4.1) * Symfony (5.4.12) * Bundles :   - DebugBundle  - DoctrineBundle  - DoctrineMigrationsBundle  - FrameworkBundle  - MakerBundle  - MonologBundle  - SecurityBundle  - SensioFrameworkExtraBundle  - TwigBundle  - TwigExtraBundle  - WebProfilerBundleWebProfilerBundle  - WebpackEncoreBundle | |

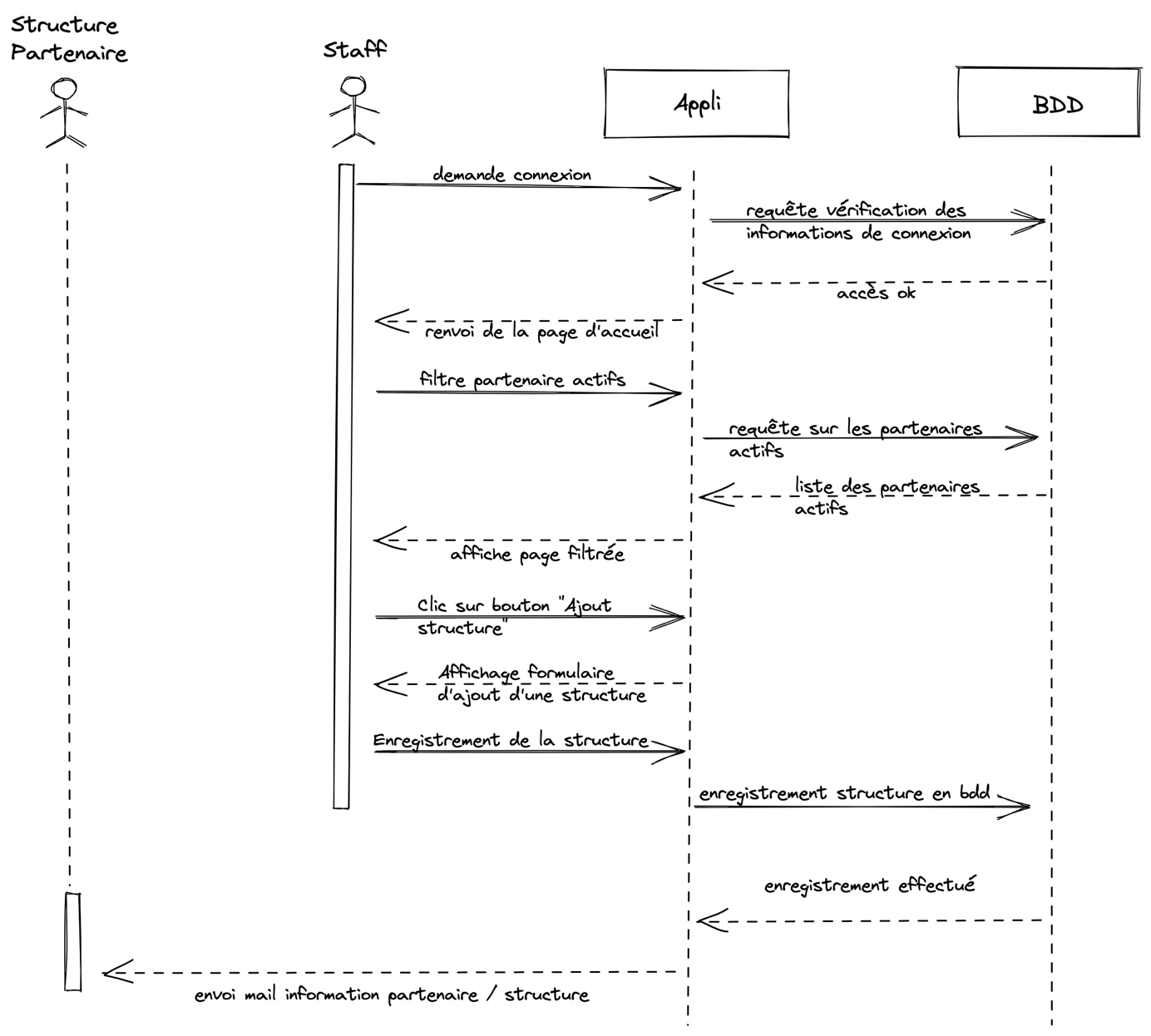
# Diagramme de cas d’utilisation :



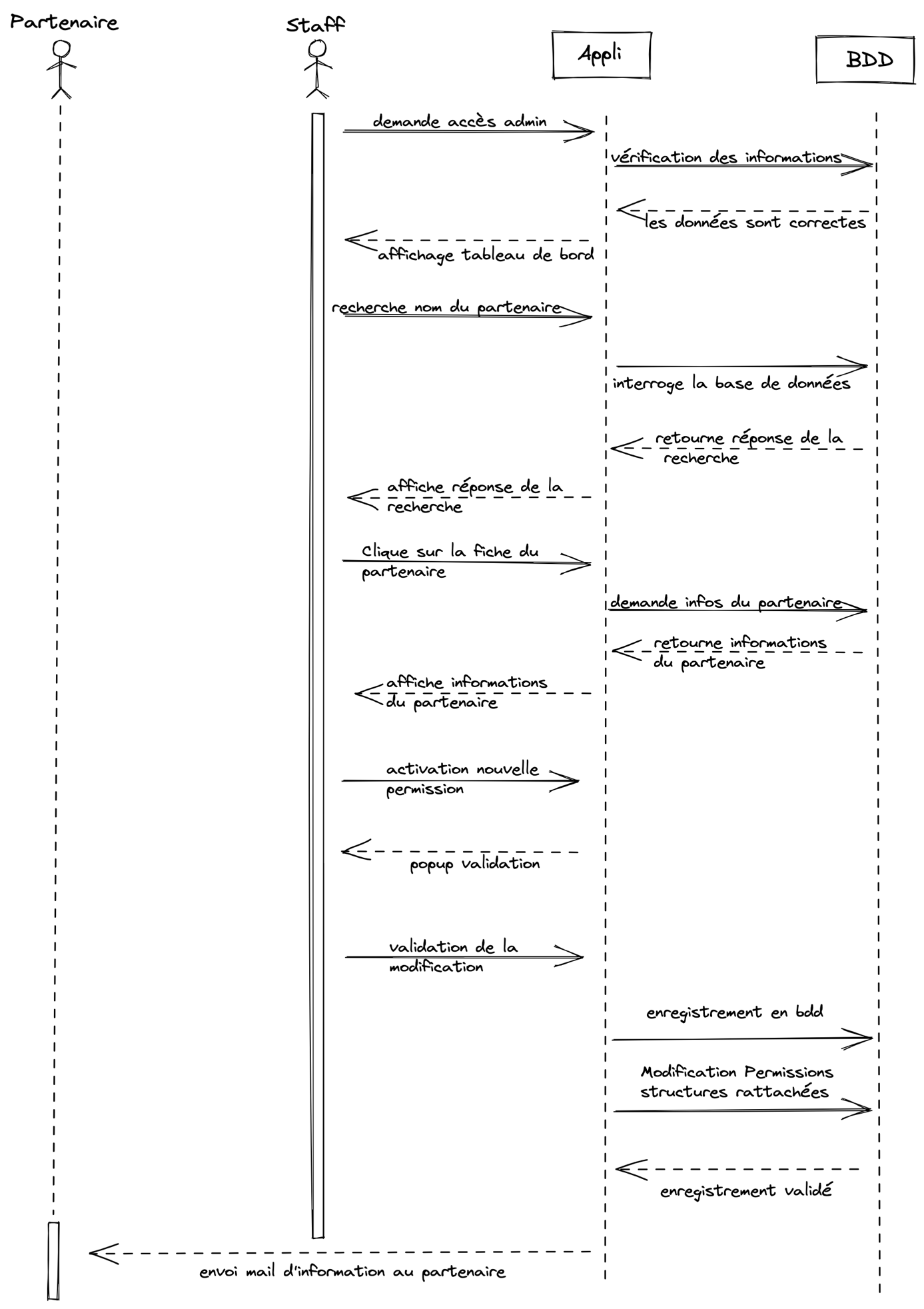
* UC1. Naviguer sur l'interface.
* UC2. Visualiser partenaires désactivés.
* UC3. Visualiser les partenaires actifs.
* UC4. Activer/désactiver partenaire.
* UC5. Consulter structures désactivées.
* UC6. Consulter les structures activées et les permissions.
* UC 7. Activer/Désactiver structures.
* UC8. Modifier les permissions.
* UC9. Ajouter une salle.
* UC10. Envoyer automatiquement une demande de confirmation au partenaire.
* UC11. Ajouter permissions.

# Diagrammes de séquences :

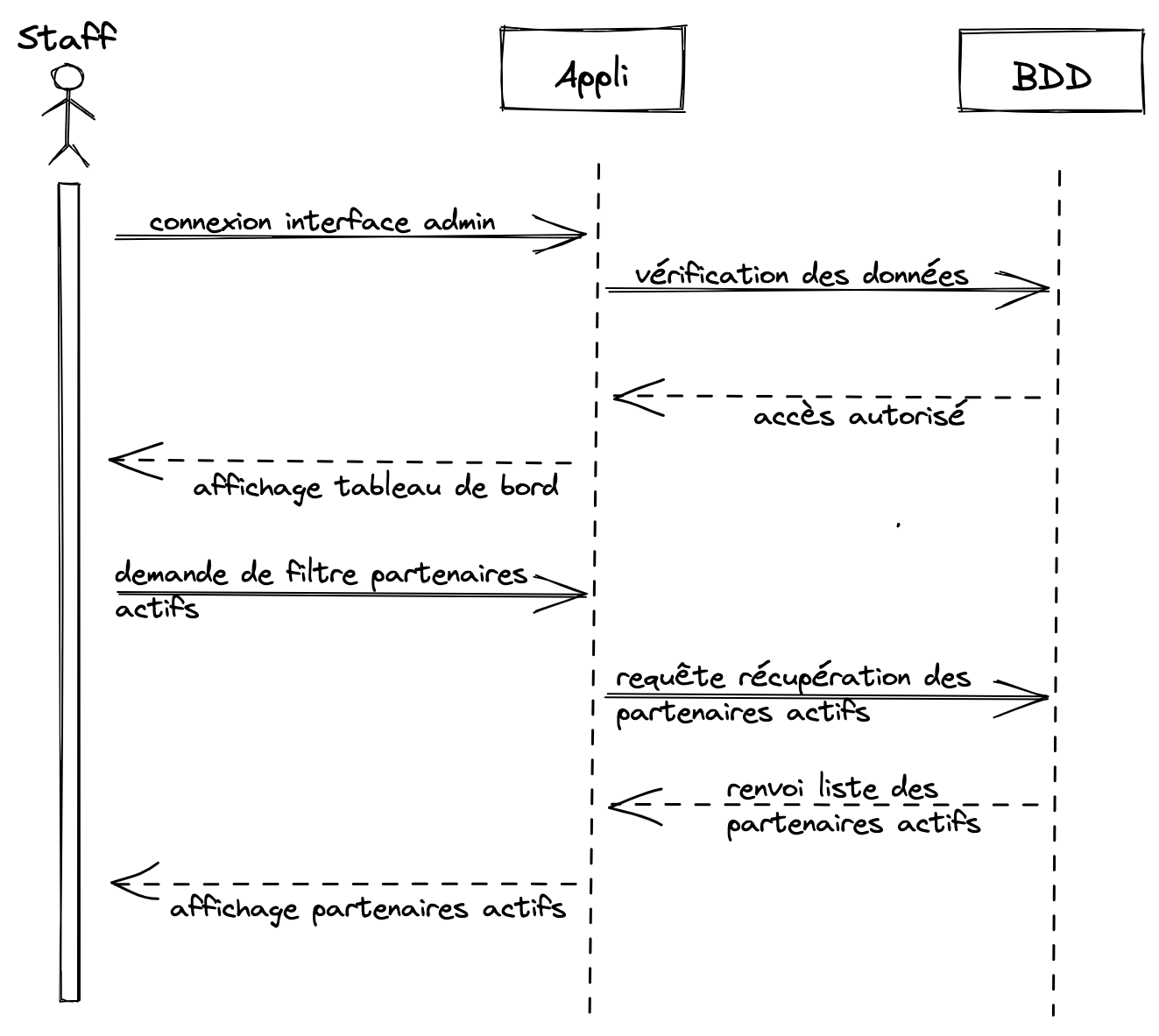
## DS1 Ajout d’une structure :



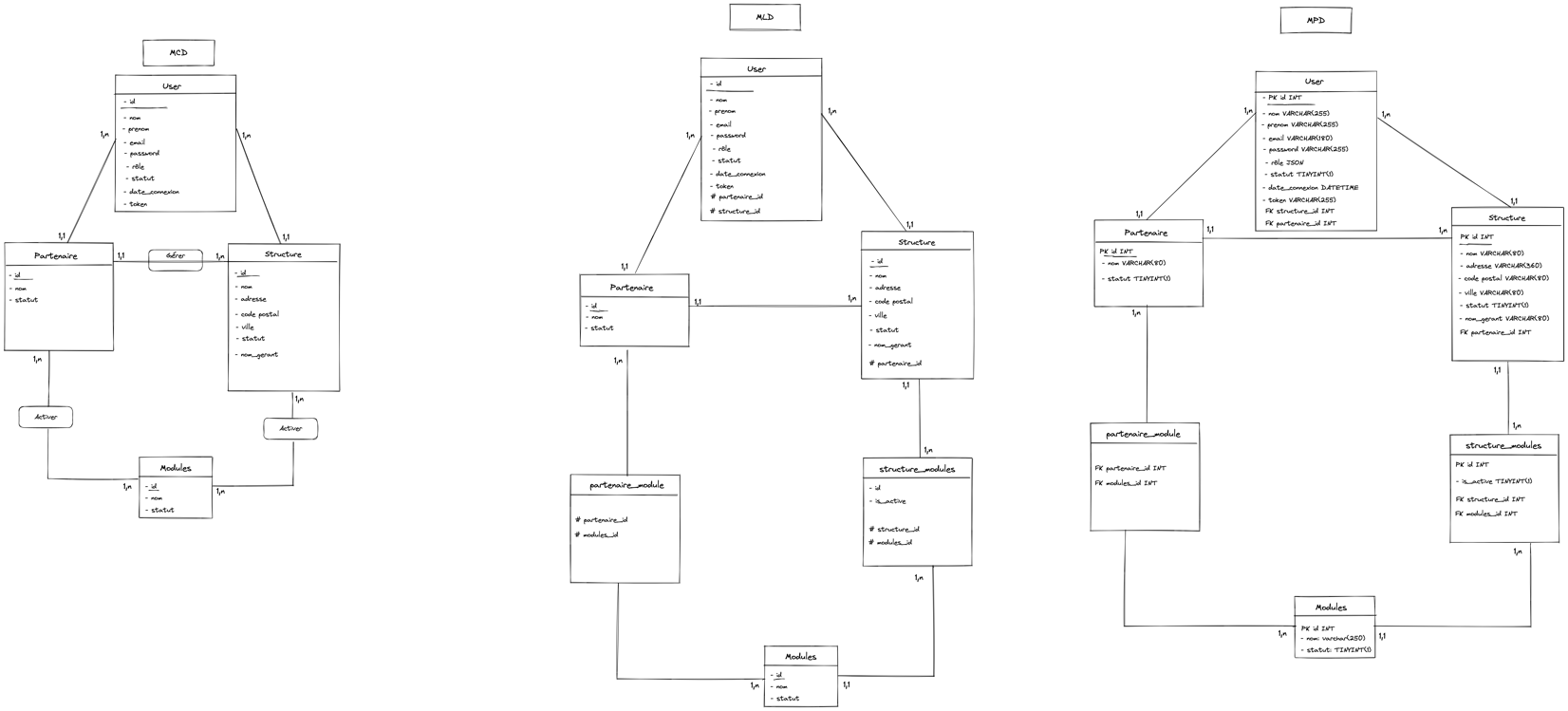
## DS2 Modification des droits :



## DS3 Sélection des partenaires actifs :



# Diagramme de classe :



# Mesures de sécurité :

L’utilisation de Symfony et particulièrement du bundle Security offre un panel d’outils visant à gérer et renforcer la sécurité de l’application.

## L’authentification :

- Renforcement de la politique de sécurité lors de la création du mot de passe par l’utilisateur

- Les mots de passe sont chiffrés en base de données

## Les autorisations :

- Gestion des droits utilisateurs, et restriction des accès à certaines pages de l’application

## Protection contre les injections :

- Validation obligatoire des données rentrées par l’utilisateur, grâce au système de « contraintes » de Symfony

- L’utilisation de l’ORM Doctrine pour gérer les interactions avec la base de données protège l’application des injections SQL, grâce au système de requêtes préparées.

- L’utilisation du moteur de rendu TWIG protège l’application contre le « Cross site scripting » (XSS), grâce à sa syntaxe entre double accolades, qui permet l’échappement des données.

## Les formulaires :

- Symfony ajoute une protection contre le « Cross site request forgery » (CSRF), en implémentant un token lors de la validation d’un formulaire.

## Le protocole HTTPS :

- Le service HTTPS activé vient renforcer la sécurité des échanges.