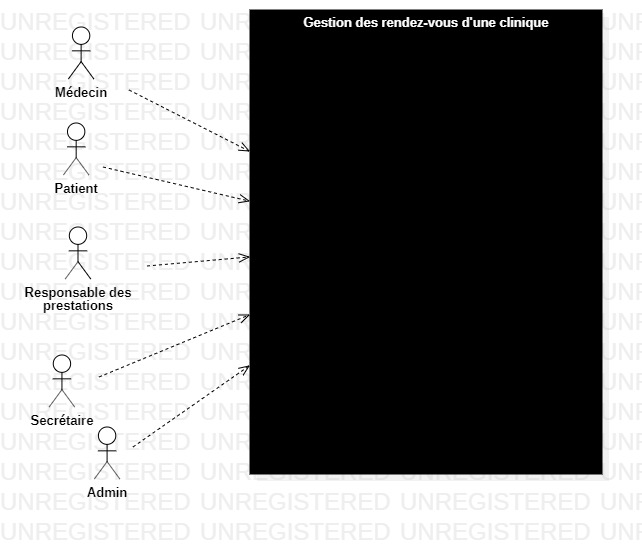
**Projet Java : Gestion des rendez-vous d’une clinique**

1. Vues des besoins
2. Diagramme de contexte

* Acteurs :
  + Médecin
  + Patient
  + Secrétaire
  + Responsable des prestations
* Fonctionnalités par acteurs

|  |  |
| --- | --- |
| Acteurs | Fonctionnalités |
| Médecin | * Se connecter * Planifier rendez-vous * Lister rendez vous * Rechercher dossier médical par patient * Voir les détails d’une consultation * Lister ses consultations * Filtrer consultation * Annuler consultation |
| Patient | * Se connecter * Créer un compte * Faire une demande de rendez-vous * Lister ses rendez-vous, consultations, prestations |
| Secrétaire | * Se connecter * Valider un rendez-vous |
| Responsable des prestations | * Lister des prestations * Voir les détails d’une prestation * Filtrer les prestations par date * Annuler les prestations * Se connecter |
| Admin | * Créer profile |

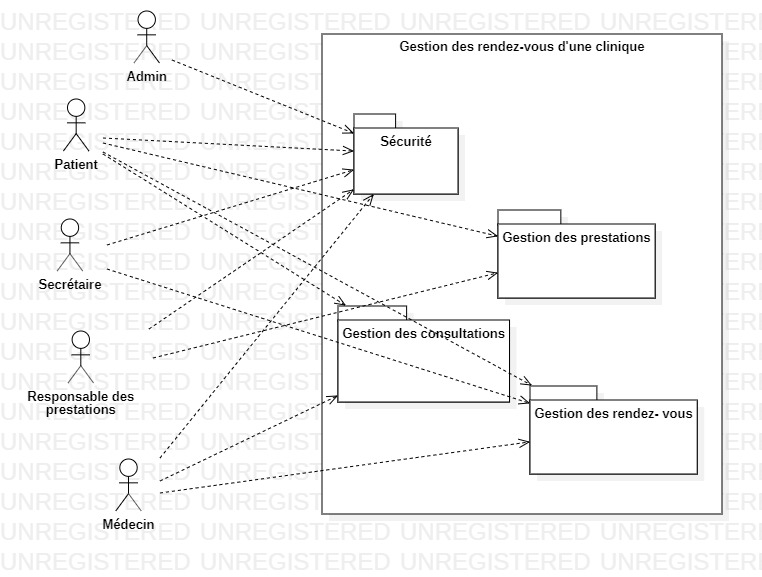


**Diagramme de Contexte**

1. Diagramme de Package

* Fonctionnalités par package

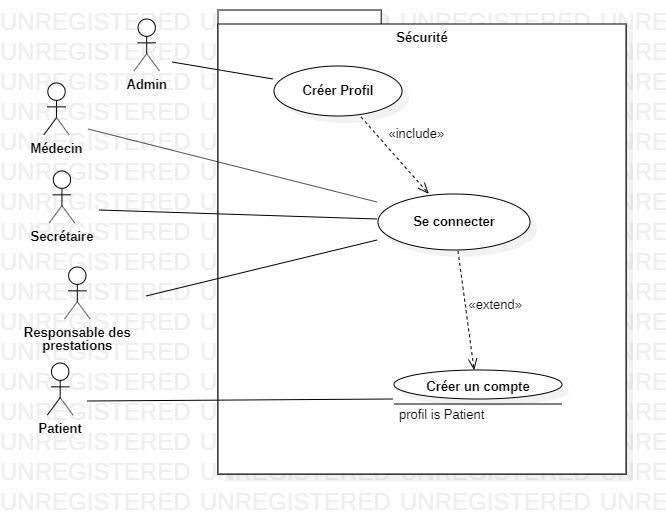
|  |  |
| --- | --- |
| Package | Fonctionnalité |
| Sécurité | * Créer compte * Se connecter * Créer profile |
| Gestion des prestations | * Lister des prestations * Voir les détails d’une prestation * Filtrer les prestations par date * Annuler les prestations |
| Gestion des consultations | * Voir les détails d’une consultation * Lister ses consultations * Filtrer consultation * Annuler consultation |
| Gestion des rendez vous | * Faire une demande de rendez-vous * Lister rendez-vous * Planifier rendez-vous * Rechercher dossier médical par patient |



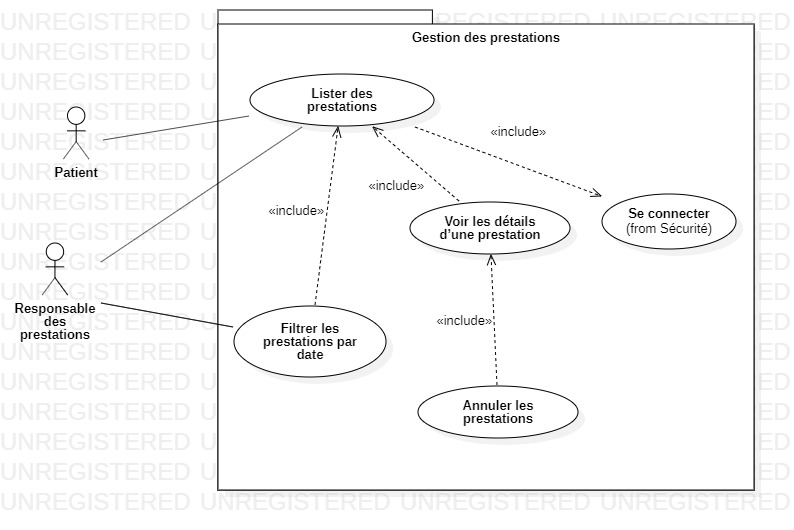
**Diagramme de Package**

1. Diagramme de Use case

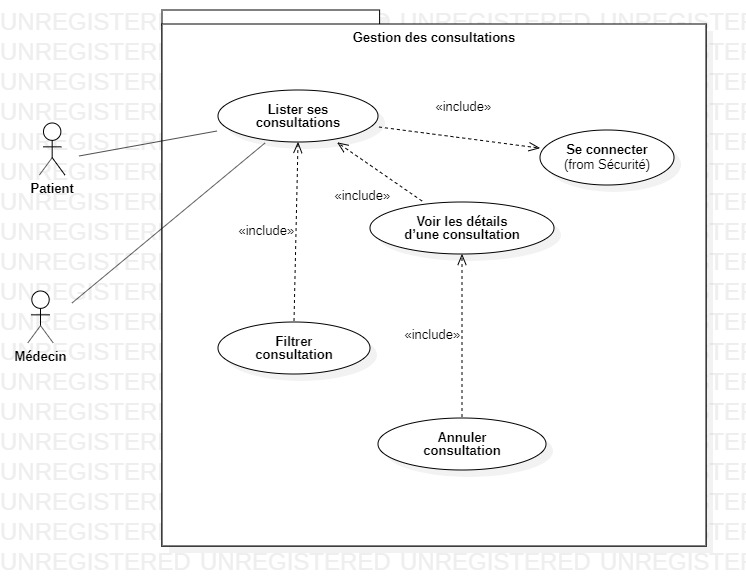
* Sécurité



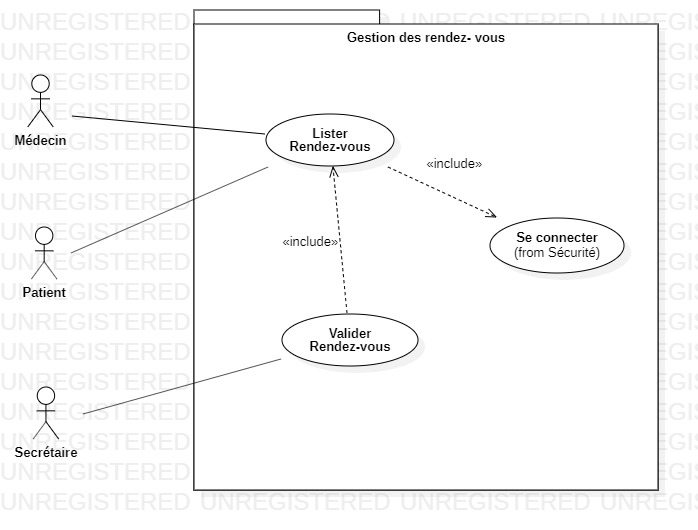
* Gestion des prestations



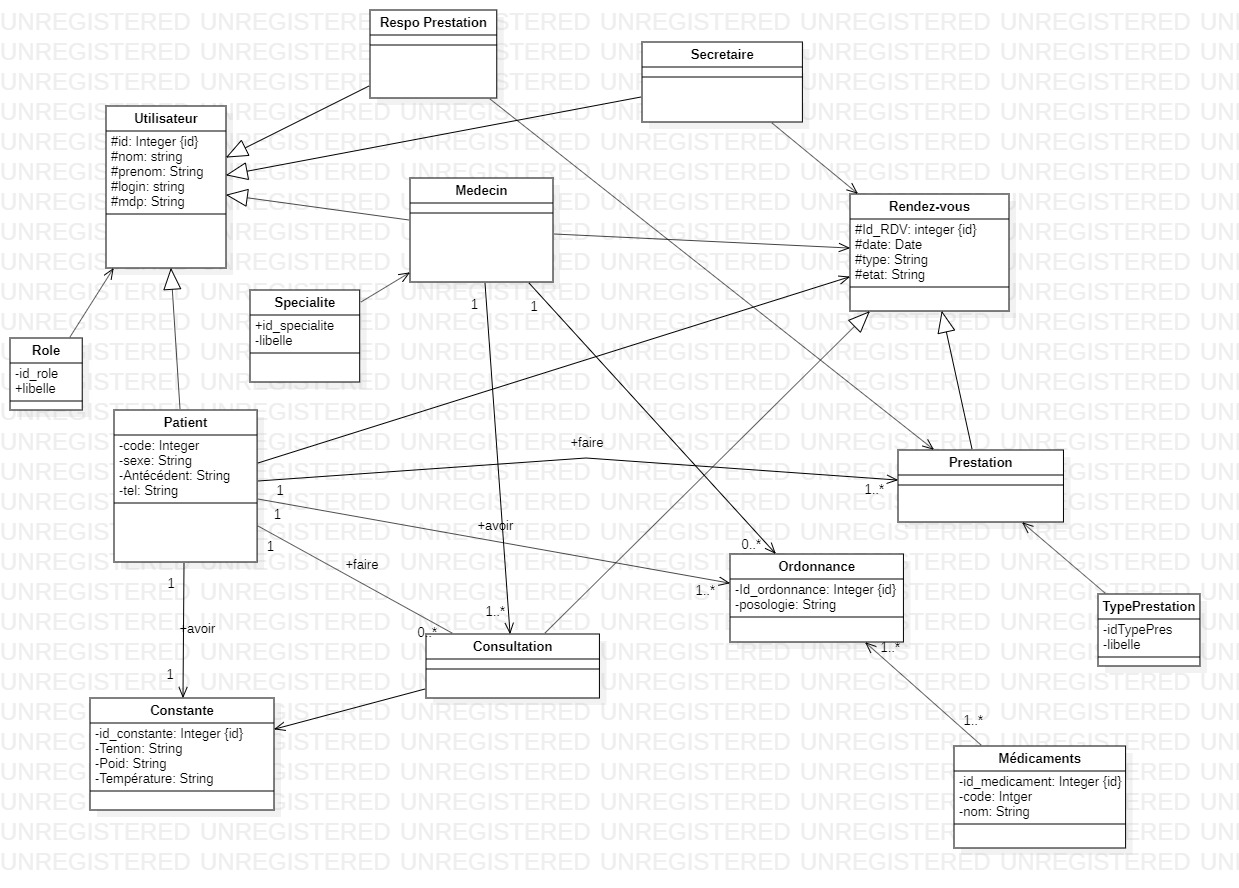
* Gestion des consultations



* Use case diagramme Gestion des rendez-vous



1. Diagramme de classe



1. Diagramme de séquence

* Se connecter

