|  |  |
| --- | --- |
| ID | F\_REQ\_1 |
| Description | Un client minimum doit pouvoir se connecter |
| Priorité | Must |
| Version | 1.0 |

|  |  |
| --- | --- |
| ID | F\_REQ\_2 |
| Description | L’administrateur doit pouvoir créer, afficher des chansons albums et playlists |
| Priorité | Must |
| Version | 1.0 |

|  |  |
| --- | --- |
| ID | F\_REQ\_3 |
| Description | Le client doit pouvoir afficher les chansons, albums, playlists (stocké sur le serveur) à la console |
| Priorité | Must |
| Version | 1.0 |

|  |  |
| --- | --- |
| ID | F\_REQ\_4 |
| Description | Le client doit pouvoir mettre à jour le contenu présent sur le serveur et afficher les modifications à la console. |
| Priorité | Must |
| Version | 1.0 |

|  |  |
| --- | --- |
| ID | F\_REQ\_5 |
| Description | Sur le serveur la mise à jour du contenu est automatique avec l’affichage des modifications à la console |
| Priorité | Must |
| Version | 1.0 |

|  |  |
| --- | --- |
| ID | F\_REQ\_6 |
| Description | Le client doit pouvoir écouter les morceaux |
| Priorité | Must |
| Version | 1.0 |

|  |  |
| --- | --- |
| ID | F\_REQ\_7 |
| Description | L’administrateur doit avoir accès aux journaux d’erreurs, d’avertissements et d’information horodatés contenus dans un fichier texte |
| Priorité | Should |
| Version | 1.0 |

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Nom | Type | Description |
| P\_Q5 | Principe SOLID | NF | L’application doit respecter les principe solide |
| P\_Q6 | GITHub | NF | L’application doit être disponibles sur Github |

User stories :

US.1: En tant que client je peux me connecter au serveur afin d’accéder à son contenu.

US.2: En tant qu’administrateur je peux créer des chanson, albums et playlists afin d’ajouter et d’actualiser du contenu musical.

US.3: En tant qu’administrateur je peux afficher les chansons, albums et playlists afin de vérifier le contenu.

US.4: En tant que client je peux afficher les chansons, albums et playlists afin d’avoir accès au contenu.

US.5: En tant que client je peux actualiser le contenu présent sur le serveur et d’afficher les modifications simultanément afin d’avoir accès au nouveau contenu.

US.6: En tant que qu’administrateur je veux que les modifications du contenu soit affichée et mise à jour automatiquement afin de bien comprendre comment le contenue évolue.

US.7 : En tant que client je peux lire le contenu présent sur le serveur afin d’écouter de la musique.

US.8 : EN tant qu’administrateur je peux avoir accès aux journaux d’erreur afin d’avoir un aperçu des erreurs.

**Critères d’acceptance :**

US.1-CA :

-La commande de mise à jour du contenu est fonctionnelle.

-L’accès au contenu du serveur est possible.

-Un message de confirmation de la connexion est affiché à la console du client et celle du serveur.

US.2-CA :

-Le ficher XML est mis à jour après une modification et après avoir mis à jour le contenu.

-L’affichage de la liste des albums, playlists ou chansons prend en compte la modification.

US.3-CA :

-La liste des albums, chansons et playlist s’affichent à la console.

US.4-CA :

-La liste des albums, chansons et playlist s’affichent à la console.

US.5-CA :

-Les modifications sont bien affichées à la console.

-L’affichage des chansons, albums et playlists inclus bien les modifications.

US.6-CA :

-Un message de confirmation de modification est affiché à la console.

-Le fichier XML est mis à jour avec le nouveau contenu.

-Le client peut accéder au nouveau contenu.

US.7-CA :

-Le son se fait bien entendre.

-La musique se lit jusqu’à la fin.

-Si une musique est déjà en cours de lecture, elle s’arrête avant le commencement de la deuxième.

US.8-CA :

-La liste des informations, avertissements et erreurs s’enregistres dans un fichier texte.

-Chaque élément est sur une ligne et est horodaté.

Exigences non-fonctionnelles

|  |  |
| --- | --- |
| ID | NF\_REQ\_1 |
| Description | L’application doit respecter les principe SOLID |
| Priorité | Must |
| Version | 1.0 |

|  |  |
| --- | --- |
| ID | NF\_REQ\_2 |
| Description | L’application doit être disponible sur Github |
| Priorité | Must |
| Version | 1.0 |

|  |  |
| --- | --- |
| ID | NF\_REQ\_3 |
| Description | Les données du client ne sont pas conservé après la fermeture de l’application |
| Priorité | Must |
| Version | 1.0 |

|  |  |
| --- | --- |
| ID | NF\_REQ\_4 |
| Description | Utilisation d’un IDE tel que IntelliJ ou Eclipse |
| Priorité | Should |
| Version | 1.0 |

|  |  |
| --- | --- |
| ID | NF\_REQ\_5 |
| Description | Utiliser un patron de conception bien défini |
| Priorité | Must |
| Version | 1.0 |

|  |  |
| --- | --- |
| ID | NF\_REQ\_6 |
| Description | Utiliser un script Maven pour le build |
| Priorité | Must |
| Version | 1.0 |

|  |  |
| --- | --- |
| ID | NF\_REQ\_6 |
| Description | Utiliser JUINT et le packaging en .jar pour les test uinitaires |
| Priorité | Must |
| Version | 1.0 |

|  |  |
| --- | --- |
| ID | NF\_REQ\_8 |
| Description | Utiliser javadoc pour générer la documentation |
| Priorité | Must |
| Version | 1.0 |

Constraint stories :

CS.1 :

CS.2 :

CS.3 :

CS.4 :

CS.5 :

CS.6 :