



| **Révision** | **Rédacteurs** | **Date** | **Objet** |
| --- | --- | --- | --- |
| 1 | R. BONNAMY | 17/02/2021 | Création du document |
|  |  |  |  |
|  |  |  |  |

1. **INTRODUCTION**
   1. ***Objet du document***

Ce document a pour objectif de présenter l’essentiel des questions techniques liées à la mise en place de l’application V.

Ce document présente :

* Le diagramme de classes
* Le modèle physique de données

1. **SOMMAIRE**
   * 1. **Table des matières**

[**1**](#_heading=h.chr3nvwarizu) **INTRODUCTION 2**

[1.1](#_heading=h.pylorgyedach) Objet du document 2

[**2**](#_heading=h.qzp7xf4529nv) **SOMMAIRE 3**

[*2.1.1*](#_heading=h.xr65eomxxnjk) *Table des matières 3*

[**3**](#_heading=h.whgv49hgjhxj) **ARCHITECTURE LOGICIELLE 4**

[3.1](#_heading=h.mi9688or5rlz) Produits et versions 4

[*3.1.1*](#_heading=h.mi9688or5rlz) *Logiciels installés sur le poste de travail 4*

[*3.1.2*](#_heading=h.6y0umteik0ya) *Langages, frameworks et librairies spécifiques 4*

[*3.1.3*](#_heading=h.54qi69gsx312) *Serveur de base de données 4*

[**4**](#_heading=h.uh9hw3tamid5) **FOCUS TECHNIQUE 5**

[4.1](#_heading=h.7es0awy6ivly) Diagramme de classes métier (ou MCD) 5

[4.2](#_heading=h.l1cmzjkcsv0i) Diagramme entités relations (ou MPD) 5

[4.3](#_heading=h.8hc7pf77ixud) Règles de développement coté back 5

[**5**](#_heading=h.cokes8v0hggl) **TESTS ET INTÉGRATION 6**

[5.1](#_heading=h.2h0477cfun62) Tests unitaires 6

[5.2](#_heading=h.fzm2vtlmmmuh) Tests fonctionnels 6

[5.3](#_heading=h.56a6ofx8gcvm) Indicateurs de qualité de code 6

1. **ARCHITECTURE LOGICIELLE**
   1. ***Produits et versions***
      1. **Logiciels installés sur le poste de travail**

| **Nom** | **Version** |
| --- | --- |
| Spring Tool Suite ou IntelliJ | ? |
| MySQL | ? |
| Git bash | ? |
| Java Development Kit | ? |
|  |  |
|  |  |

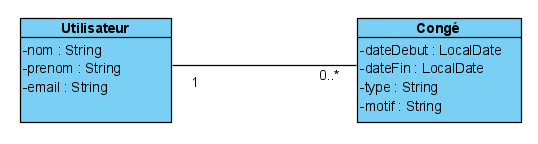
* + 1. **Langages, frameworks et librairies spécifiques**

| **Nom** | **Version** |
| --- | --- |
| Spring Boot |  |
| JPA |  |
| Angular |  |
| iText (génération de PDF) |  |
| Apache POI (génération d’excel) |  |

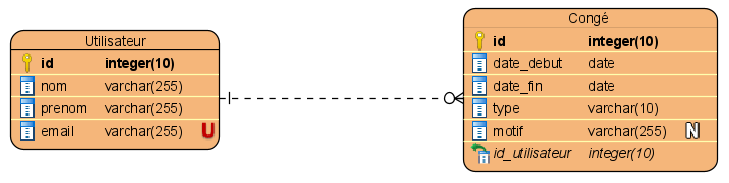
* + 1. **Serveur de base de données**

| **Nom** | **Version** |
| --- | --- |
| MySQL | 8 |
|  |  |

1. **FOCUS TECHNIQUE**
   1. ***Diagramme de classes métier (ou MCD)***



* 1. ***Diagramme entités relations (ou MPD)***



* 1. ***Règles de développement coté back***

Règles de développement :

* 100% de la Javadoc doit être renseignée
* Les règles de nommage respectent les conventions de nommage en usage en 2025.

**Découpage en couches** :

Décrire les différentes couches :

* Couche contrôleur (qui contient les routes d’accès au back)
* Couche de services qui va réaliser les contrôles métier. Ex : DemandeCongeService
* Couche repository (spring data JPA)
* Couche DTO : le contrôleur renvoie vers le front des instances de DTO.
* Couche entités avec les annotations JPA.
* Classes utilitaires : *exemple : classe DateUtils pour le traitement des dates*

Couche contrôleur

Couche service

Couche repository

1. **TESTS ET INTÉGRATION**
   1. ***Tests unitaires***

Il y aura des tests unitaires à développer pour tester le code de toutes les classes de services.

Mettre ici le tableau de couverture des tests unitaires : lorsque vous aurez eu les 3 jours de projet sur l’intégration continue.

* 1. ***Tests fonctionnels***

Des campagnes de tests seront réalisées pour tester les différents cas d’utilisation.

Mettre ici le résultat de votre campagne de tests : lorsque vous serez sur la phase de finalisation du projet.

* 1. ***Indicateurs de qualité de code***

Renseigner à 100% la javadoc.

Règles de nommage des classes, des packages : conventions standards à respecter

Mettre ici le tableau SonarQube de qualité de code : lorsque vous aurez eu les 3 jours de projet sur l’intégration continue.