

# Manuel de Rédaction des documents d'Under

THE JDN

20 avril 2025



# Objectif du document

L'objectif de ce manuel est de présenter l'ensemble des documents qui seront rédigés et utilisés tout au long du projet Under dans l'ensemble de ses phases. Il s'insère ainsi dans l'atteinte de plusieurs objectifs divers couvrant chacun un aspect en étroite relation avec la gestion de la communication et la standardisation des pratiques de travail à employer tout au long du projet. Ses objectifs principaux sont alors **la standardisation de la rédaction, la définition des structures des documents, l'identification des responsabilités, l'amélioration de la Communication, la prévention, détection et gestion des erreurs et incohérences**. Le présent document permettra ainsi de définir quelles sont les parties prenantes en charge de la rédaction de chaque document ainsi que les raisons et aspects techniques pour lesquelles celles-ci sont plus aptes à mener à bien la rédaction de ces documents. En plus des parties prenantes à la rédaction, il sera également question de présenter quelles sont les ayants droits d'utilisation pour chaque document, afin de limiter le cadre d'utilisation d'un document tant à son périmètre d'utilisation dans la réalisation du projet, qu'à son périmètre d'utilisateurs. Il s'agira enfin de fournir une structure de base à respecter de vigueur comme template de rédaction de chaque document, afin qu'il réponde aussi bien sur l'aspect structurel que fonctionnel aux exigences pour lesquelles il a été conçu.



# Guide d'utilisation

L'utilisation de ce document de par ses objectifs énoncés ci-haut, est voué à diverses utilisations, chacune prenant vie du titre qu'occupe son utilisateur et en moindre mesure du niveau d'avancement ou de la phase courante du projet.

Pour :

**Le Chef de projet :** le manuel ici présent servira de socle pour chaque phase du projet, présentant à la fois le minimum requis de contenu au sein de chaque document et de chaque phase.

**Architectes système, équipe de développement, DevOps :** l'ensemble des conventions, normes et pratiques utilisées devront soit être en accord avec les standards définis au sein de ce manuel, soit ne pas être en désaccord avec l'une d'entre elles.

**Architectes système, équipe de développement, DevOps :** Donner un aperçu rapide et concret sur le projet pour les nouveaux membres et ressourcer quand nécessaire les anciens.



# Conventions typographiques, Définitions, acronymes et abréviations

Tout au long du document, un ensemble de symboles, acronymes, abréviations ou termes techniques pourront être utilisés afin d'alléger la charges du texte au sein du document. l'ensemble de ces éléments se réfèreront alors à une définition ou un sens bien précis, dont on énumèrera les correspondances (éléments (signes, défintions,..) = significations) au sein de cette partie.





## Part I

# Initiation et Planification du projet



# Chapter 1

## Cahier des Charges

**Rôle** : Décrire le projet dans sa globalité, Définir les besoins fonctionnels et non-fonctionnels, établir les critères d'acceptabilité du livrable final ainsi que les ressources y dédiés.

**Créé par** : Analystes d'affaires, Product Owner, Architectes système

**Utilisé par** : Toute l'équipe de développement, stakeholders

## **1.1 Sommaire :**

### **1.1.1 Introduction**

Objectif du document

Définitions, acronymes et abréviations

Références

### **1.1.2 Description générale**

Naissance et contexte du projet

Etude de l'existant

Présentation générale du projet

Analyse générale du projet

Perspective du produit

Fonctions du produit

Caractéristiques des utilisateurs

Contraintes générales

Hypothèses et dépendances

### **1.1.3 Exigences fonctionnelles**

Core Layer

Infrastructure Layer

Domain Layer

Application Layer

Presentation Layer

Communication inter-applications

### **1.1.4 Exigences non-fonctionnelles**

4.1 Performance

4.2 Sécurité

4.3 Fiabilité

4.4 Maintenabilité

4.5 Portabilité

4.6 Extensibilité

**1.1.5 Autres exigences**

**Standards à respecter**

**Exigences système**

**Exigences légales**

**1.1.6 Annexes**

**Modèles d'analyse**

**Glossaire**

**Références**



## Chapter 2

# Plan de Réalisation du Projet

**Rôle** : Décrire les différentes phases d'exécution du projet selon leur chronologie ainsi que les livrables attendus à la fin de chaque phase, les différentes parties prenantes à la réalisation de chaque phase ainsi que le responsable de chaque phase la documentation y dédiée et les liens logiques entre les phases.

**Créé par** : Chef de projet

**Utilisé par** : Toute l'équipe, Stakeholders

## **2.1 Sommaire :**

### **2.1.1 Introduction**

Objectif du projet

Portée

Hypothèses et contraintes

### **2.1.2 Organisation du projet**

Structure de l'équipe

Rôles et responsabilités

Parties prenantes

### **2.1.3 Approche de gestion**

Méthodologie

Cycles de vie

Processus de gouvernance

### **2.1.4 Planification**

WBS (Work Breakdown Structure)

Calendrier et jalons

Dépendances

Chemin critique

### **2.1.5 Ressources**

Ressources humaines

Matériel et logiciel

Budget

### **2.1.6 Gestion de la qualité**

Objectifs de qualité

Assurance qualité

Contrôle qualité



### **2.1.7 Gestion des risques**

**Identification des risques**

**Analyse qualitative**

**Stratégies d'atténuation**

**Suivi des risques**

### **2.1.8 Plan de communication**

**Stratégie de communication**

**Outils et canaux**

**Calendrier des communications**

### **2.1.9 Gestion des changements**

**Processus de contrôle des changements**

**Comité de contrôle des changements**

### **2.1.10 Annexes**

**Diagramme de Gantt**

**Matrice RACI**

**Registre des risques**



## Part II

# Conception du projet



## Chapter 3

# Cahier de conception Architecturale



## Chapter 4

# Cahier de spécifications générales

**Rôle** : Traduire les exigences et besoins en spécifications détaillées qui serviront de base pour la conception architecturale.

**Créé par** : Architectes système, Analystes fonctionnels

**Utilisé par** : Équipe de développement, Architectes techniques

## 4.1 Sommaire

### 4.1.1 Introduction

Objectif du document

Portée

Définitions et acronymes

### 4.1.2 Vue d'ensemble du système

Contexte du système

Architecture générale

Utilisateurs et rôles

### 4.1.3 Spécifications fonctionnelles

Core Layer

Kernel

Service Container

Event System

Configuration Manager



**Infrastructure Layer**

**Domain Layer**

**Application Layer**

**Presentation Layer**

#### **4.1.4 Interfaces externes**

**Interfaces utilisateur**

**Interfaces logicielles**

**Interfaces de communication**

#### **4.1.5 Attributs de qualité**

**Performance**

**Sécurité**

**Maintenabilité**

**Portabilité**

#### **4.1.6 Contraintes de conception**

**Standards à respecter**

**Limitations techniques**

#### **4.1.7 Annexes**

**Diagrammes**

**Glossaire**

**Références**



## Chapter 5

# Cahier de spécification techniques

**Rôle** : Détaille les aspects techniques de l'implémentation, servant de guide pour les développeurs.

**Créé par** : Architectes techniques, Lead Developers

**Utilisé par** : Équipe de développement

## **5.1 Sommaire:**

### **5.1.1 Introduction**

**Objectif du document**

**Portée**

**Définitions et acronymes**

### **5.1.2 Architecture technique**

**Vue d'ensemble**

**Patterns de conception utilisés**

**Frameworks et bibliothèques**

**Structure du monorepo**

### **5.1.3 Spécifications par couche**

**Core Layer**

Kernel (implémentation)

Service Container (implémentation)

Event System (implémentation)

Configuration Manager (implémentation)

**Infrastructure Layer**

**Domain Layer**

**Application Layer**

**Presentation Layer**

#### **5.1.4 Interfaces de programmation**

**API publiques**

**API internes**

**Hooks et points d'extension**

#### **5.1.5 Gestion des données**

**Modèles de données**

**Persistance**

**Caching**

#### **5.1.6 Sécurité**

**Authentification**

**Autorisation**

**Protection des données**

**Audit et logging**

#### **5.1.7 Performance**

**Optimisations**

**Benchmarks**

**Profiling**

#### **5.1.8 Annexes**

**Code d'exemple**

**Références**



## Chapter 6

# Document de Modèle de données

**Rôle** : Définit la structure des données manipulées par le framework et les applications.

**Créé par** : Architectes de données, Développeurs

**Utilisé par** : Développeurs, DBA

## **6.1 Sommaire:**

### **6.1.1 Introduction**

Objectif du document

Portée

Définitions et conventions

### **6.1.2 Vue d'ensemble du modèle de données**

Principes de conception

Diagramme global

### **6.1.3 Entités et attributs**

Core Entities

Infrastructure Entities

Domain Entities

Application Entities

### **6.1.4 Relations**

Relations entre entités

Cardinalités

Contraintes d'intégrité

### **6.1.5 Modèles spécifiques**

Modèle relationnel (ORM)

Modèle documentaire (ODM)

Modèles hybrides

### **6.1.6 Stratégies de persistance**

Mapping objet-relationnel

Mapping objet-document

Caching



### **6.1.7 Évolution du modèle**

**Stratégies de migration**

**Versionnement**

### **6.1.8 Annexes**

**Schémas SQL**

**Schémas NoSQL**

**Glossaire**



## Chapter 7

# Cahier de conception Système

**Rôle** : Représente visuellement les classes du système et leurs relations.

**Créé par** : Architectes techniques, Développeurs

**Utilisé par** : Équipe de développement

## **7.1 Sommaire :**

### **7.1.1 Introduction**

**Objectif du document**

**Conventions de notation UML**

### **7.1.2 Vue d'ensemble**

**Organisation des packages**

**Diagramme global**

### **7.1.3 Diagrammes par couche**

**Core Layer**

Kernel

Service Container

Event System

Configuration Manager

**Infrastructure Layer**

**Domain Layer**

**Application Layer**

**Presentation Layer**

#### **7.1.4 Patterns de conception**

**Patterns créationnels**

**Patterns structurels**

**Patterns comportementaux**

#### **7.1.5 Interfaces et abstractions clés**

**Interfaces du Core**

**Interfaces d'Infrastructure**

**Interfaces de Domain**

**Interfaces d'Application**

#### **7.1.6 Diagrammes de séquence**

**Initialisation du framework**

**Traitement d'une requête**

**Résolution de services**

#### **7.1.7 Annexes**

**Légende UML**

**Références**



## Part III

# Développement





## Chapter 8

# Standards et normes de développement



## Chapter 9

# Manuel Technique

**Rôle** : Guide technique pour les développeurs qui utilisent ou contribuent au framework.

**Créé par** : Développeurs, Technical Writers

**Utilisé par** : Développeurs

## **9.1 Sommaire:**

### **9.1.1 Introduction**

**À propos de ce manuel**

**Public cible**

**Conventions utilisées**

### **9.1.2 Installation et configuration**

**Prérequis**

**Installation**

**Configuration initiale**

**Vérification de l'installation**

### **9.1.3 Architecture du framework**

**Vue d'ensemble**

**Core Layer**

**Infrastructure Layer**

**Domain Layer**

**Application Layer**

**Presentation Layer**

### **9.1.4 Développement avec Under**

**Création d'une nouvelle application**

**Structure des applications**

**Développement de bibliothèques**

**Communication inter-applications**

## **9.1.5 API de référence**

**Core API**

**Infrastructure API**

**Domain API**

**Application API**

**presentation API**

## **9.1.6 Exemples de code**

**Applications simples**

**Applications complexes**

**Intégrations**

## **9.1.7 Bonnes pratiques**

**Conventions de codage**

**Patterns recommandés**

**Optimisation des performances**

**Sécurité**

## **9.1.8 Dépannage**

**Problèmes courants**

**Debugging**

**Logging**

## **9.1.9 Annexes**

**Glossaire**

**Références**

**Ressources supplémentaires**



## Chapter 10

# Documentation APIs





## Part IV

### Test



## Chapter 11

# Plan et Méthodologie de test

**Rôle :** Définit l'approche, les ressources et le calendrier des activités de test.

**Créé par :** QA Lead, Testeurs

**Utilisé par :** Équipe de test, Développeurs

## **11.1 Sommaire:**

### **11.1.1 Introduction**

Objectif du document

Portée des tests

Références

### **11.1.2 Stratégie de test**

Types de tests

Niveaux de tests

Techniques et outils de test

### **11.1.3 Éléments à tester**

Core Layer

Infrastructure Layer

Domain Layer

Application Layer

Presentation Layer

Intégrations

### **11.1.4 Environnements de test**

Configuration matérielle

Configuration logicielle

Données de test

### **11.1.5 Critères de test**

Critères d'entrée

Critères de sortie

Critères de suspension et de reprise

### **11.1.6 Livrables de test**

Plan de test

Cas de test

Données de test

Rapports de test

## Chapter 12

# Rapport de test

**Rôle** : Documente les résultats des activités de test.

**Créé par** : QA Lead, Testeurs

**Utilisé par** : Équipe de développement, Stakeholders

## 12.1 Sommaire:

### 12.1.1 Résumé exécutif

Objectifs des tests

Portée des tests

Résumé des résultats

### 12.1.2 Environnement de test

Configuration matérielle

Configuration logicielle

Données de test

### 12.1.3 Résultats des tests par type

Tests unitaires

Tests d'intégration

Tests système

Tests de performance

Tests de sécurité

### 12.1.4 Résultats des tests par composant

Core Layer

Infrastructure Layer

Domain Layer

Application Layer

Presentation Layer

### 12.1.5 Défauts

Résumé des défauts

Défauts critiques

Défauts majeurs

Défauts mineurs

### 12.1.6 Couverture de test

Couverture de code

Couverture fonctionnelle

## Part V

# Déploiement





## Chapter 13

# Guide de déploiement

**Rôle** : Documente le processus de déploiement du framework et des applications.

**Créé par** : DevOps, Administrateurs système

**Utilisé par** : Équipe de déploiement, Administrateurs système

## **13.1 Sommaire:**

### **13.1.1 Introduction**

Objectif du document

Environnements cibles

### **13.1.2 Architecture de déploiement**

Topologie des environnements

Composants d'infrastructure

Diagramme de déploiement

### **13.1.3 Prérequis**

Matériel

Logiciel

Configuration réseau

### **13.1.4 Procédures de déploiement**

Installation initiale

Mises à jour

Rollback

### **13.1.5 Configuration**

Variables d'environnement

Fichiers de configuration

Secrets et sécurité

### **13.1.6 Monitoring et logging**

Outils de monitoring

Alertes

Logs

### **13.1.7 Procédures de récupération**

Sauvegarde

Restauration

Reprise après sinistre

### **13.1.8 Annexes**

## Chapter 14

# Manuel de sécurité



## Chapter 15

# Guide d'intégration



## Part VI

# Maintenance





## Chapter 16

# Plan de Maintenance



## Chapter 17

# Rapports des Maintenances



## Part VII

# Suivi du projet



## Chapter 18

# Cahier de suivi

**Rôle** : Suit et communique l'avancement du projet aux parties prenantes.

**Créé par** : Chef de projet, Scrum Master

**Utilisé par** :Stakeholders, Équipe de projet

## **18.1 Sommaire :**

### **18.1.1 Résumé exécutif**

État général du projet

Points clés

### **18.1.2 Avancement par rapport au plan**

Calendrier prévu vs réel

Budget prévu vs réel

Scope prévu vs réel

### **18.1.3 Jalons**

Jalons atteints

Prochains jalons

Jalons à risque

### **18.1.4 Avancement par composant**

Core Layer

Infrastructure Layer

Domain Layer

Application Layer

Presentation Layer

### **18.1.5 Problèmes et risques**

Problèmes rencontrés

Solutions mises en œuvre

Risques identifiés

Stratégies d'atténuation

### **18.1.6 Prochaines étapes**

Activités planifiées

Décisions requises

### **18.1.7 Annexes**

Métriques détaillées

Rapports de qualité