|  |
| --- |
| {{Client}}  {{Project}}  Technical Design file  Version {{Version}} |
| **Author**  {{Author}}  {{Author\_Role}} |

To do:

* Manually replace values between <>
* Values in between {{}} will be indicated in the document’s properties

Table of Contents

1 -Versions 3

2 -Introduction 4

2.1 -Purpose of document 4

2.2 -References 4

3 -Technical Architecture 5

3.1 -General components 5

3.1.1 -Package A 5

3.1.1.1 -Component X 5

3.1.1.2 -Component Y 5

3.1.2 -Package B 5

3.1.2.1 -Component Z 5

3.2 -Web Application 5

3.2.1 -Components X 5

3.2.2 -Components Y and Z 5

3.3 - XXX Application... 5

4 -Roll-Out Architecture 6

4.1 -Database Server 6

4.2 - XXX Server 6

5 -Software Architecture 7

5.1 -General principles 7

5.1.1 -Layers 7

5.1.2 -Modules 7

5.1.3 -Source structure 7

5.2 -Web Application 8

5.3 - Xxx Application 8

6 -Specific points 9

6.1 -Log management 9

6.2 -Configuration folders 9

6.2.1 -Web application 9

6.2.1.1 -Datasources 9

6.2.1.2 - Xxx yyy folder 9

6.2.2 - Xxx Application 9

6.3 -Resources 9

6.4 -Development environment 9

6.5 -Packaging procedure / delivery 9

6.6 -XXX 9

7 -Glossary 10

# Versions

|  |  |  |  |
| --- | --- | --- | --- |
| Author | Date | Description | Version |
| Xxx | DD/MM/YYYY | Document creation | XXX |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Document purpose

The present document makes up the technical conception folder for the XXX application…

Document objective …

The elements of the present folder ensue:

* from …
* ...

## References

For further information, please refer also to the following elements:

1. **FDF - Xxx** : Functional design folder for the application
2. …

# Technical Architecture

## General components

### Package A

#### Component X

Description and role/objective

#### Component Y

### Package B

#### Component Z

## Web application

The software stack is as follows:

* **J2EE** application (JDK version 1.8) / **PHP** (version) / **Python**…
* Application server **JOnAS 5.2.4 / ...**

UML Component Diagram

### Components X

Description and role/objective

### Components Y and Z

## XXX... application

# Roll-Out Architecture

UML Roll-out diagram

Explanation/comments if needed

## Database Server

Description

Technical specifications (ex: Linux server Debian Jessie + PostgreSQL 9.6…)

Important information/specific points

## XXX Server

...

# Software architecture

## General principles

Project sources and versions are managed by **Git*,*** dependencies and packaging by **Apache Maven/Grunt/…**

...

### Layers

Application architecture is as follows:

* a **business** layer: responsible for the business logic of the component
* a **model** layer: implementation of the business objects model
* …
* …
* ...

### Modules

Ex: Maven modules in the case of a multi-module application…

### Source structure

The logic for the structure of the project directories is as follows:

* the source directories are created so as to respect the Maven philosophy (i.e. “convention over configuration”)

root  
 ├─ *pom.xml*  
 ├─ <moduleX>  
 │ ├─ *pom.xml*  
 │ └─ src  
 │ ├─ main  
 │ │ ├─ java  
 │ │ └─ resources  
 │ └─ test  
 │ ├─ java  
 │ └─ resources  
 ├─ <moduleY>  
 │ ├─ *pom.xml*  
 │ └─ src  
 │ ├─ main  
 │ │ ├─ java  
 │ │ └─ resources  
 │ └─ test  
 │ ├─ java  
 │ └─ resources  
 └─ src  
 └─ lib

* ...

## Web Application

…

If needed, UML component diagram to show the various modules and their interdependencies

## Xxx Application

…

# Specific points

## Log Management

…

## Configuration folders

### Web application

...

#### Data sources

...

#### Xxx.yyy folder

...

### Xxx Application

...

## Resources

...

## Development environment

## Packaging / delivery procedure

## XXX

…

# Glossary

|  |  |
| --- | --- |
|  |  |
|  |  |