**Plan d’assurance qualité**



Mandant : Noctambus

**Auteurs :** João AMARAL, Luca FALVO et Anthony PALAMA

**Etat :** En élaboration

**Date :** Semestre 5, 2015

Suivie des modifications

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Modification | Auteur |
| 21.10.2015 | 1 | Création du fichier | A |

Table des matières

[1. But, domaine d’application et responsabilités 4](#_Toc433697595)

[1.1 Introduction 4](#_Toc433697596)

[1.2 Définitions, Acronymes et Abréviations 4](#_Toc433697597)

[1.3 Logiciel concerné par le PAQ 4](#_Toc433697598)

[1.4 Responsabilité associée au PAQ 4](#_Toc433697599)

[1.5 Procédure d’évolution du plan qualité logiciel 4](#_Toc433697600)

[1.6 Procédure à suivre en cas de non-application du PAQ 4](#_Toc433697601)

[2. Documents applicables et de référence 4](#_Toc433697602)

[2.1 Documents applicables 4](#_Toc433697603)

[2.2 Documents de référence 4](#_Toc433697604)

[2.3 Glossaire des termes utilisés 5](#_Toc433697605)

[3. Organisation 5](#_Toc433697606)

[3.1 Structure du projet 5](#_Toc433697607)

[ Équipe de projet 5](#_Toc433697608)

[ Comité de pilotage 5](#_Toc433697609)

[4. Démarche de développement 5](#_Toc433697610)

[4.1 Le cycle de développement 5](#_Toc433697611)

[4.2 Description des phases de développement 6](#_Toc433697612)

[4.3 Identification des produits issus de chaque phase 6](#_Toc433697613)

[5. Documentation 6](#_Toc433697614)

[5.1 Les documents de gestion de projet 6](#_Toc433697615)

[5.2 Les documents techniques de réalisation 6](#_Toc433697616)

[5.3 Les manuels d’utilisation et d’exploitation 6](#_Toc433697617)

[6. Gestion de la configuration 7](#_Toc433697618)

[6.1 Structure de la configuration 7](#_Toc433697619)

[ Répertoire de production 7](#_Toc433697620)

[ Répertoire de développement 7](#_Toc433697621)

[6.2 Convention d’identification des éléments de la configuration 7](#_Toc433697622)

[6.3 Procédure d’identification et de gestion de configuration 7](#_Toc433697623)

[ Identification 7](#_Toc433697624)

[ Responsabilités 7](#_Toc433697625)

[ Procédures de gestion de la configuration 7](#_Toc433697626)

[7. Gestion des modifications 8](#_Toc433697627)

[7.1 Constat d’anomalie 8](#_Toc433697628)

[7.2 Demande d’évolution 8](#_Toc433697629)

[8. Méthodes, outils et règles 8](#_Toc433697630)

[8.1 Méthodes 8](#_Toc433697631)

[8.2 Outils 8](#_Toc433697632)

[9. Contrôle des fournisseurs 9](#_Toc433697633)

[10. Reproduction, protection, livraison 9](#_Toc433697634)

[10.1 Reproduction et protection 9](#_Toc433697635)

[10.2 Livraison-installation 9](#_Toc433697636)

[11. Suivi de l’application du plan qualité 9](#_Toc433697637)

[11.1 Revues de documentation 9](#_Toc433697638)

[11.2 Revues de début de phase 9](#_Toc433697639)

[11.3 Tests 9](#_Toc433697640)

[11.4 Acceptations 9](#_Toc433697641)

# But, domaine d’application et responsabilités

## Introduction

L’objectif de ce document est de décrire l’ensemble des dispositions spécifique du projet Noctambus.

## Définitions, Acronymes et Abréviations

PAQ = Plan d’Assurance Qualité

GREP = GRoupe d’Encadrement de Projet

UML = Unified Modeling Language

## Logiciel concerné par le PAQ

Logiciel résultant exclusivement du projet Noctambus

## Responsabilité associée au PAQ

Le PAQ est rédigé par Anthony Palama, qui devra aussi s’occuper de développer l’application. De plus, Joao Amaral et Luca Falvo sont responsables de la coordination des actions à faire pour une bonne application du PAQ ainsi que du développement.

## Procédure d’évolution du plan qualité logiciel

La procédure de modification du PAQ est la suivante :

* Trouver et rédiger la modification
* Transmettre le PAQ afin qu’il soit appliqué.

Chaque modification doit être validée par le GREP

## Procédure à suivre en cas de non-application du PAQ

En cas de non-application du PAQ, une réunion avec le GREP doit être prise le plus rapidement possible ainsi qu’avec le mandant afin de statuer sur la dérogation.

# Documents applicables et de référence

## Documents applicables

Tous les documents à respecter scrupuleusement sont les suivants :

L’Étude d’opportunité du 5 octobre 2015 référencé sous le nom Étude d’Opportunité v3.docx

Le document d’acceptation A1 du 14 octobre 2015 sous le nom Acceptation A1 v2 .docx

D’autres documents sont en cours d’élaboration, ils seront rajoutés ultérieurement.

## Documents de référence

Les documents de base sont les documents de référence, et ce sont les suivants : Les comptes rendus de réunions référencés sous le nom date-pv.docx, exemple : « 08.10.15-pv.docx »

## Glossaire des termes utilisés

Document d’acceptation: Éléments acceptés par le GREP

IBM Rational Software Architect: Outil de conception et de modélisation.

SWIFT Langage de programmation

Xcode Environnement de développement pour SWIFT

JAVA : Langage de programmation

Eclipse et Android studio : Environnement de développement pour JAVA

UML : Notation standard pour la modélisation

Ms Project Planification des tâches

# Organisation

## Structure du projet

### Équipe de projet

Les personnes suivantes s’occupent de la conception et de la réalisation du projet Noctambus.

Monsieur Joao AMARAL : Développeur

Monsieur Luca FALVO : Développeur

Monsieur Anthony PALAMA : Scrum Master

### Comité de pilotage

Les personnes suivantes s’occupent de la prise de décision ainsi que du suivi et de la coordination globale.

Monsieur David BILLARD : Responsable du GREP

Monsieur Rolf HAURI : Professeur du GREP

Monsieur Julien GLOOR : Assistant du GREP

Monsieur Sékou CISSE : Product owner

# Démarche de développement

## Le cycle de développement

Les étapes du cycle de développement sont les suivantes

1. Étude d’opportunité
2. Acceptations
   1. A1
   2. A2
3. Diagrammes
4. Livrables (chaque 5 semaines)
   1. Exécutable
   2. Manuel d’utilisateur
   3. Sprint backlogs

## Description des phases de développement

Phase 1 : Étude d’opportunité

Etude des risques

Etude de l’existant

Architecture future

Planification

Phase 2 : Sprint (5 semaines)

Phase 3 : Conclusion

## Identification des produits issus de chaque phase

Phase 1 : Etude d’opportunité

Document d’acceptation A1, A2, A3

Document d’architecture

Document d’étude des risques

Document de planification

Phase 2 : Jeux de test

Livrable

Manuel d’utilisation

Phase 3 : Dossier de récapitulatif de projet

Dossier finaux du projet

# Documentation

## Les documents de gestion de projet

PAQ

Planification

Procès-verbaux de réunion

## Les documents techniques de réalisation

Le dossier d’étude d’opportunité

Le dossier de conception

Le dossier de l’architecture

Le dossier de test

Le manuel du programmeur

## Les manuels d’utilisation et d’exploitation

Le manuel d’installation

Le manuel d’utilisation

# Gestion de la configuration

## Structure de la configuration

### Répertoire de production

Contient la dernière version de l’application

Les fichiers sont partagés via GitHub c’est un logiciel qui permet de partager des fichiers ainsi que faire du versioning.

..Noctambus

Ce répertoire est lui-même décomposé en :

Dossier contient la dernière version des documents.

Powerpoint contient la dernière version des powerpoint.

### Répertoire de développement

../Noctambus

Application contient la dernière version de l’application

Architecture contient la dernière version de l’architecture du système

## Convention d’identification des éléments de la configuration

Tous les éléments du projet Noctambus suivent les règles d’identifications suivantes :

Pour la documentation l’identification est donnée par un nom de type : <Mnémonique> <version>.docx. Par exemple : PAQ v2.docx

## Procédure d’identification et de gestion de configuration

### Identification

La gestion des versions est, comme dit précédemment, conservée par GitHub. Chaque élément validé permet de créer un nouveau « fork », une fourchette, c’est-à-dire un jalon validé. Ce qui permet de figer la version du document.

### Responsabilités

Le responsable qualité doit s’occuper de gérer les « forks » ainsi que vérifier les versions finales.

Chaque élément inséré dans ce dossier crée automatiquement une sauvegarde du fichier final dans un serveur privé.

### Procédures de gestion de la configuration

La mise à jour des nomenclatures contenant la liste des éléments de configuration identifiée et leur relation est à effectuer à la fin de chaque phase, donc chaque « fork ».

# Gestion des modifications

## Constat d’anomalie

Si une anomalie est détectée, elle doit suivre les étapes suivantes afin de corriger et de mettre à jour les documents suivants :

1. Détection de l’anomalie.
2. Analyse
3. Annonce au groupe de pilotage
4. Planification de la correction
5. Correction
6. Test
7. Mise en production
8. Validation

## Demande d’évolution

Si une évolution du logiciel est souhaitée, elle doit suivre la procédure suivante afin de corriger et de mettre à jour des documents suivants :

1. Demande de l’évolution
2. Analyser : Comprendre les composants impactés estimation des impacts
3. Validation de la demande et des impacts par le comité de pilotage
4. Planification de l’évolution
5. Modification évolutive
6. Tests
7. Mise-en-production
8. Validation

# Méthodes, outils et règles

## Méthodes

L’équipe de projet suit la méthode définie dans le manuel dans le dossier « Acceptation A1 v2.docx »

## Outils

Communication Microsoft Outlook, Mozilla ThunderBird, Mail

Gestion de projet et planification Microsoft Projet

Gestion des dossiers et versioning GitHub

Outil de conception IBM Rational Software Architect

Outil de réalisation Eclipse

Outil de réalisation Xcode

# Contrôle des fournisseurs

Dans ce projet, nous n’avons pas de communication avec les fournisseurs

# Reproduction, protection, livraison

## Reproduction et protection

Reproduction autorisée uniquement par le mandant

## Livraison-installation

La livraison-installation des différents livrables sera effectuée sous la forme suivante :

* Les logiciels sont livrés par clé USB ainsi qu’un exécutable en format .apk
* Les documents d’installation et d’accompagnement sont livrés par clé USB

# Suivi de l’application du plan qualité

## Revues de documentation

Lecture corrigée par les membres du groupe ainsi que par des membres externes au projet

## Revues de début de phase

Contrôler les objectifs à atteindre par l’équipe de développement avec le carnet de sprint (Sprint backlog)

## Tests

Tests unitaires

Tests d’intégration

## Acceptations

Chaque phase donne lieu à une acceptation de la part du propriétaire du produit (product owner) ainsi que par le comité de pilotage.