

[2017][TS] Consignes

*Consignes pour le bilan documentation
stratégie de test*

Description du document

Titre	[2017][TS] Consignes
Date	12/10/2015
Auteur	Julien FREROT
Responsable	Flavien ASTRAUD
E-Mail	eip-tech@labeip.epitech.eu
Sujet	Consignes pour le bilan documentation stratégie de test
Mots clés	
Version du modèle	1.0

Tableau des révisions

Date	Auteur	Section(s)	Commentaire
28/11/2013	Julien FREROT	Toutes	Première version
28/11/2013	Julien FREROT	2	Correction après relecture
18/09/2014	Julien REROT	Toutes	Mise à jour 2016
25/09/2014	Julien FREROT	3.2.2	Mise à jour plan
12/10/2015	Julien FREROT		Mise à jour 2017

Table des matières

1	Objectif du bilan.....	1
1.1	Description	1
1.2	Glossaire	1
2	Formalités	1
3	Se préparer	1
3.1	Formalisme	1
3.2	Contenu du document	2
3.2.1	Résumé.....	2
3.2.2	Plan basique	2
4	Barème.....	4
5	Remarques.....	4

1 Objectif du bilan

1.1 Description

Ce bilan a pour but de vous faire réfléchir à la stratégie de tests, et au niveau de test que vous allez devoir fournir pour votre projet. C'est une phase importante pour la qualité de votre projet, afin de savoir comment la mesurer, et quelles sont les attentes. Outre l'aspect tests, c'est le moment pour définir le plan de release de votre projet, quels types de tests à quel moment selon les versions majeures / mineures, mais aussi sur l'outillage, et les méthodologies.

Un autre objectif de ce document est de vous sensibiliser à l'univers de la release et du test, le but est que vous cherchiez à comprendre les outils et méthodes qui existent pour gérer la qualité d'un projet.

1.2 Glossaire

- G -

Gantt : Diagramme utilisé pour représenter le travail à accomplir sur un axe temps, découpé en tâches et associé à des ressources.

Gestion de configuration : C'est l'outil utilisé pour gérer les différentes versions d'un livrable. Des exemples de gestion de configuration pour le code source sont Git, TFS, SVN, ...

- J -

Jalon : C'est une date (tâche de 0 jour de durée et 0 jour de travail), modélisée par un losange sur le Gantt, qui est notable, importante et/ou qui dépend d'autres tâches.

- R -

Release : Une release est une mise à disposition d'une version de l'application. Elle peut être privée, semi publique ou publique. En général une release fait état de la version commerciale (ou plus détaillée pour des releases fermées).

- S -

Scope : C'est ce qui définit votre projet, l'inventaire de ce qui le compose.

2 Formalités

- Le groupe doit avoir déposé au plus tard le **15 novembre 2015 à 23h42** le document dans son SVN dossier /rendu
- Le document doit être nommé **2017_TS_<Groupe>.pdf** et rendu au format document portable (PDF), <Groupe> étant votre nom de groupe.

3 Se préparer

3.1 Formalisme

Tout d'abord, le document doit avoir un format spécifique. Il est obligatoire de voir figurer des en-têtes et pieds de pages, une page de garde qui indique le titre du document, le nom du groupe et éventuellement un visuel du groupe et de l'EIP/Epitech.

Ensuite, vous devez faire figurer sur la deuxième page un résumé du document (cf Résumé)

Sur la troisième page, vous devez faire figurer un cartouche du document (propriétés, métadonnées du document), et un tableau des révisions, permettant de tracer l'historique de votre document à la façon d'un SVN (date, version, auteur, sections modifiées, commentaires).

Enfin sur la 4^{ème} page doit figurer un sommaire à jour, directement généré depuis le contenu de votre document.

Vos pages doivent être numérotées, la première page de contenu devant porter le numéro 1 (correspondant au titre 1 de votre document), les pages citées dans ce paragraphe ne faisant pas l'objet de numérotation.

3.2 Contenu du document

3.2.1 Résumé

Le résumé du document n'est pas une introduction. Il sert à synthétiser les idées principales exprimées dans le document. Le résumé s'adresse à une personne qui ne lirait pas forcément le document mais qui veut savoir quel est son contenu et s'il est destinataire du document. Le résumé peut faire de quelques lignes jusqu'à 1 page, selon la nature et le contenu du document, sans contrainte forte sur ce point. Il doit permettre de comprendre tous les points clés du document récapitulés dans cette partie.

3.2.2 Plan basique

Dans la deuxième partie du plan (§5), vous ferez votre choix dans les types de tests présentés rapidement en partie 3, et suivrez le plan du 5.1 que vous répèterez pour chaque type de tests que vous avez choisi d'implémenter. Les types de test possibles sont les suivants (liste non exhaustive que vous pouvez compléter) : **tests unitaires, tests système, tests de charge, stress test, tests de performance, test d'intégration, test d'assemblage, smoke test, tests fonctionnels, tests d'acceptation, ...**

1. Introduction.
 - 1.1. Objectif de l'EIP.
 - 1.2. Principe de base du système.
 - 1.3. Documents de référence.
 - 1.4. Terminologie.
2. Version/release
Vous décrierez ici votre plan de release avec les versions, en accord avec votre planning (identification de la version, contenu et importance), l'objectif est de poser les bases du plan, la stratégie de numérotation des versions (identifiant unique d'une release), et de pouvoir bâtir par la suite la stratégie de tests sur ce plan.
3. Types de tests et périmètre.
Vous ferez ici une présentation haut niveau des types de tests et de leur couverture qui sera détaillée plus tard dans le document. Cela correspond vraiment à votre stratégie globale. L'objectif est d'expliquer ici des grandes aspirations de la stratégie de tests tels que TDD, BDD, ... Il existe une multitude de tests (unitaires, performance, stress, système, intégration, régression, fonctionnels, validation, vérification, ...), auxquels vous associerez une couverture du projet (complet, fonctionnalité, ...)
4. Livrables.
 - 4.1. Cas de tests
Vous décrierez ici le livrable des cas de tests, dans son format, sa fréquence de mise à jour, son niveau de détail, quelles informations et pourquoi elles sont présentes.

4.2. Rapport d'exécution de tests

Vous décrirez ici le(s) livrable(s) de rapport d'exécution des tests, dans son format, les indicateurs clés qui sont présents et comment ils sont suivis. Le but de ce rapport est de remonter de façon uniforme et unique au fil du projet les résultats d'un test.

4.3. Rapport sur le suivi global de la qualité

Vous décrirez ici le(s) rapport(s) de suivi de la qualité du projet, qui reprend les indicateurs des rapports d'exécution et en ajoute certains avec une mesure qui s'inscrit dans le temps. Le but de ce rapport est de suivre l'évolution de la qualité pour donner un outil au chef de projet afin de savoir quand est-ce que le projet est hors de contrôle en terme de qualité.

5. Types de tests

Dans ce chapitre vous décrirez en détail les types de tests que vous allez utiliser et leurs objectifs en utilisant la même cinématique de plan. Vous pouvez ajouter tous types de tests, et reproduire les sous-chapitres tels que décrits dans les 5.1. Le but est de mieux cerner pour chaque type de test, quel est son objectif, dans quelles conditions il se réalise (comme si vous vouliez sous-traiter l'exécution des tests), et quelle charge est associée.

5.1. Tests XXX

5.1.1. Objectif

Dans ce chapitre vous décrirez quels sont les objectifs de base des tests XXX et pourquoi ils sont utilisés de manière générale

5.1.2. Environnement et conditions de réalisation

Dans ce chapitre vous décrirez quelles sont les conditions de réalisation (release, moment du projet, quel environnement, et quels autres critères définissent le scénario d'exécution de ces tests).

5.1.3. Configurations particulières

Dans ce chapitre vous décrirez les configurations spécifiques qu'il faut pour réaliser ces tests

5.1.4. Planning et charge

5.1.5. Critère de démarrage des tests

5.1.6. Critères de passage/échec

[...]

6. Estimation globale de la charge de travail

Vous ajusterez ici par rapport à votre Gantt quel va être la charge de travail à quel moment pour les différents types de tests

7. Outils

Vous listerez ici avec une simple description les outils que vous allez utiliser pour les tests (Git, Jira, TFS, HP Quality Center, Cucumber, ...). Le repository de test est l'endroit où sont stockés les livrables liés au test, la gestion des rejets/erreurs est la procédure/outil de ticket utilisé pour gérer les erreurs/rejets d'un test.

7.1. Repository de tests

7.2. Gestion de rejets/erreurs

7.3. Autres outils

7.3.1. Outils d'automatisation des tests

7.3.2. Outils pour les tests de performance

7.3.3. <Autres outils>

4 Barème

Question	/20
Format <i>Aspect général des documents, Orthographe, grammaire, style, entrée en matière : contextualisation, structure des documents</i>	4
Stratégie de tests <i>Définition de la stratégie et des seuils d'acceptation</i>	12
Exécution de tests <i>Définition du modèle de cas de tests et du rapport</i>	4

Vous serez sanctionnés si vous ne respectez pas les consignes et si vous rendez en retard votre document.

Toute modification du document après la date de rendu sera considérée comme un rendu en retard, et sera également sanctionné.

5 Remarques

- Pour nous contacter :
 - Tickets sur l'intranet EIP
 - Sujet [2017][<Groupe>][TS] <Sujet>
 - Mot clé : TS

Toutes les demandes par e-mail ou ne respectant pas ce formalisme de tickets **seront clôturées** sans réponse.