Documents

M1 INFO GL CM

Rôle des documents

Rôle : contractuel

Compréhension du client

- Cas d’utilisations
- Besoins fonctionnels et non fonctionnels
- Priorités, exigences de qualité

Cahier des charges - Spécifications

Plan de développement - Plan de validation

Contrat - Plan Assurance Qualité

Décrire ce qui est attendu, et pour quand.

Rôle : financier

En externe (client)

- Coût prévu
- Facturation

En interne, suivi:

- Coûts
- Temps

Devis - Contrat

Tableau de bord

Rôle : conceptuel

- Analyse
- Conception de l’architecture
- Conception détaillée

Spécifications -

Dossier d’Analyse Fonctionnelle

Rapport de Conception Détaillée

Rôle : suivi, organisation

- Planning
- Gestion des risques
- Suivi des avancées
- Maintient de l’adéquation au client
- Trace des réunions

Plan de développement - Fiche d’itération - Tableau de bord - Compte Rendu de Réunion - Analyse des risques - Planning

Rôle : qualité

- Les responsabilités
- Les risques
- Les exigences
- Leur évaluation

Plan Assurance Qualité - Plan de validation - Plan de tests d’intégration/unitaires - Rapport de tests unitaires/d’intégration - Dossier de tests - Dossier de validation - Recette

Rôle : livraison

- Livrer
- Installer
- Faire fonctionner
- Maintenir

Dossier de tests - Recette - Plan de déploiement - Manuel utilisateur

Rôle : utilisation / maintenance

Manuel

- Installation
- Utilisation
- Aide

Manuel utilisateur - Documentation des API - Commentaire de code - Contrat de maintenance - Dossier d’Analyse Fonctionnelle - Rapport de Conception Détaillée

Cahier des Charges

Cahier des Charges

Responsabilité du client, mais souvent fait par l'entreprise.

Résultats de l'analyse des besoins. Préciser les acteurs et leurs besoins:

- Fonctionnels, GUI, Priorités
- Non-fonctionnels (performance, capacité, compatibilité)
- Contraintes (normes, matériel, langage)
- [Plan assurance qualité]

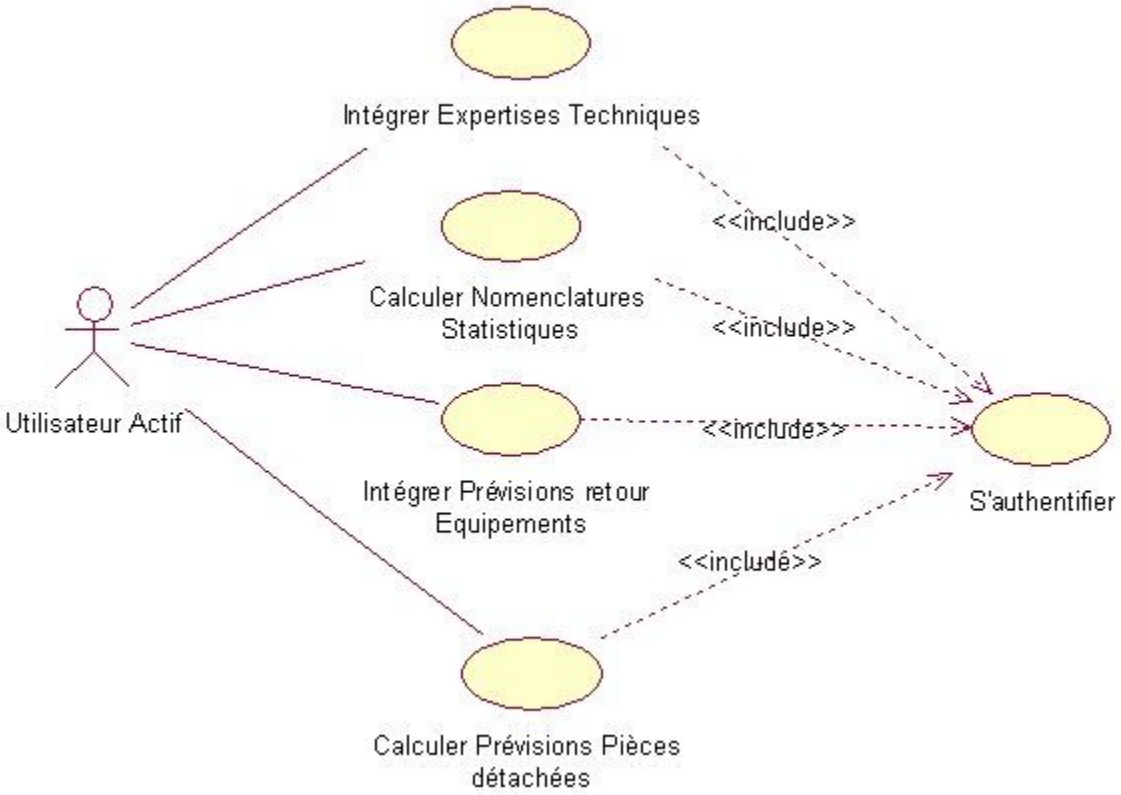
Cf. CM UML/Diagrammes CU/Processus Unifié.

Cf. CM Méthodes/Phases Analyse des besoins.

Comprendre et décrire les besoins du client.

Cahier des charges

Ex:



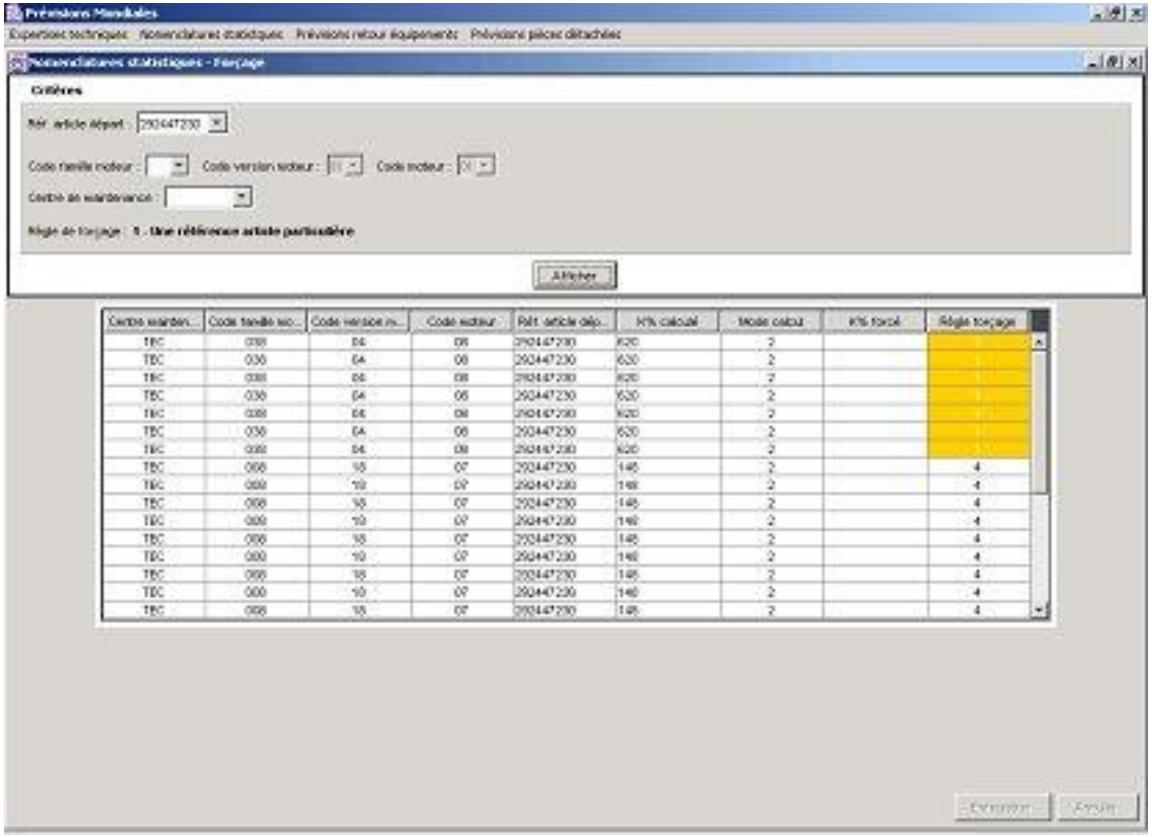
Cahier des charges

Ex:

Acteur principal	Utilisateur Identifié
Acteur secondaire	Utilisateur Inconnu
Objectifs	L'Utilisateur Identifié souhaite se connecter à l'application
Préconditions	L'Utilisateur Identifié possède un login et un mot de passe
Postconditions	1. L'Utilisateur Identifié est authentifié 2. L'Utilisateur Inconnu n'est pas authentifié
Scénario nominal	1. L'Utilisateur Identifié saisit un login et un mot de passe depuis l'écran de connexion 2. Le système vérifie en base la cohérence du login et du mot de passe 3. L'Utilisateur Identifié est connecté.
Extensions	2a. Le login ou le mot de passe n'est pas saisi. 1. Le système affiche un message d'erreur «Login (ou mot de passe) manquant ». 2. Le système revient à l'étape 1 du scénario nominal. 2a. Le login ou le mot de passe est incorrect. 1. Le système affiche un message d'erreur «Il est impossible de se connecter à l'application : login ou mot de passe erroné ». 2. Le système revient à l'étape 1 du scénario nominal
Besoins IHM	Cacher la saisie du mot de passe

Cahier des charges

Ex: Maquette de GUI



Spécifications

Spécifications

A partir du Cahier des Charges.

Contient une formalisation des fonctionnalités, ex:

- Pre/post-conditions, invariants
- Diagrammes de séquences externes
- Diagrammes Etat-transition “processus métiers”
- Diagrammes de classe “métier”

Ce que doit faire le système, ses contours. Point de vue métier. Critères de réussite, bonne pratique: faire valider par le client, en s’en servant comme interface avec lui.

* * *

Dossier d’Analyse Fonctionnelle

Dossier d'Analyse Fonctionnelle

= Dossier de Conception Préliminaire.

A partir des spécifications.

- Modéliser (Diagrammes UML)
- Packager (Diagrammes de packages)
- Classifier (Diagramme de classes principales)
- [Priorités], [Plan de développement]

Architecture du logiciel. Peut rester compréhensible par le client.

Rapport de conception détaillée

Rapport de conception détaillée

= (Dossier) d'Analyse Technique

A partir du Dossier d'Analyse Fonctionnelle.

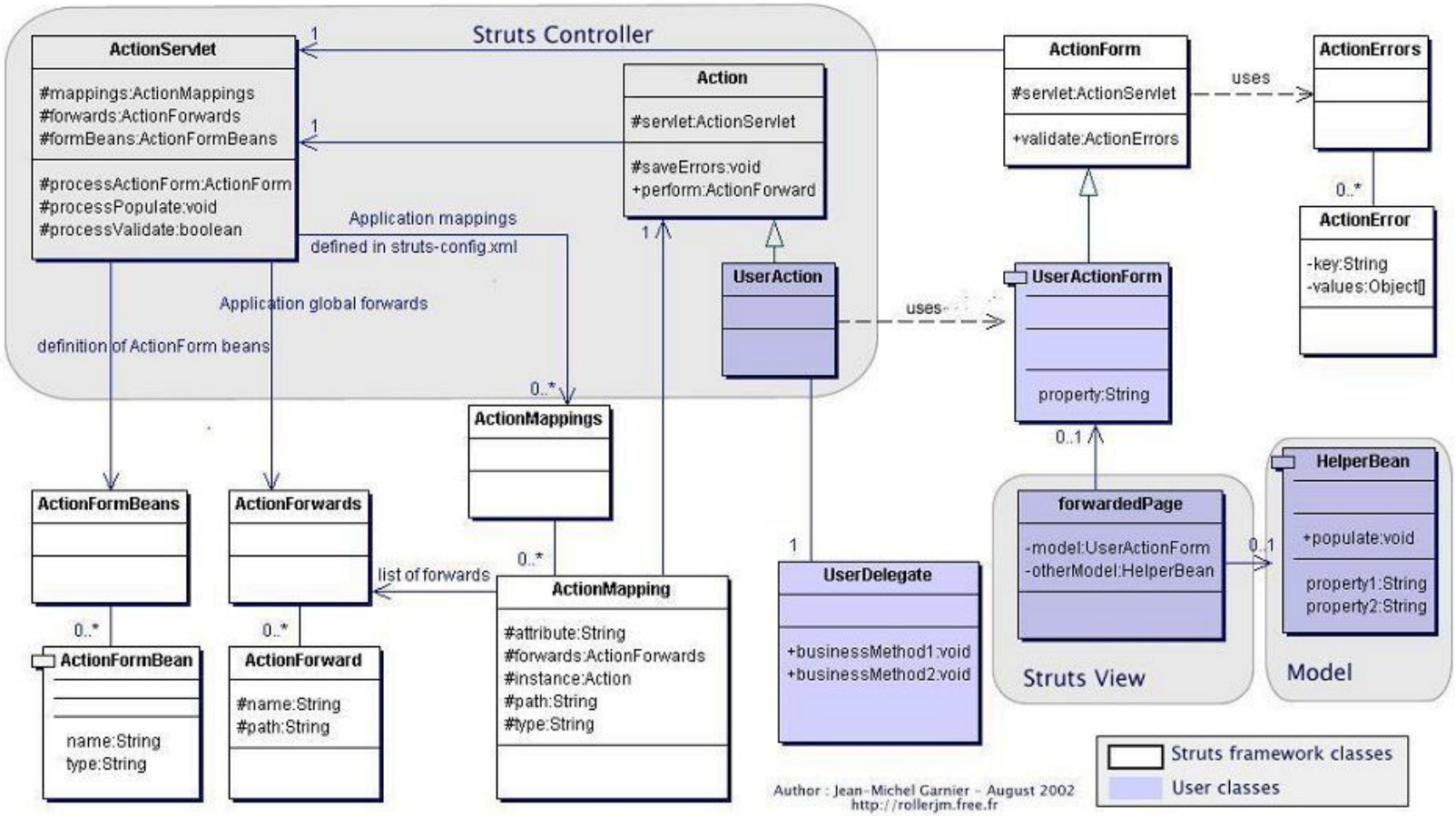
Préciser:

- Choix matériels, langages
- Diagrammes de séquences
- Diagramme de classes (avec attributs et méthodes)
- Diagrammes états-transition
- Pseudo-code

Choix techniques. Description des composantes sensibles du logiciel. Interne.

Rapport de conception détaillée

Ex:



* * *

Vocabulaire

Vocabulaire

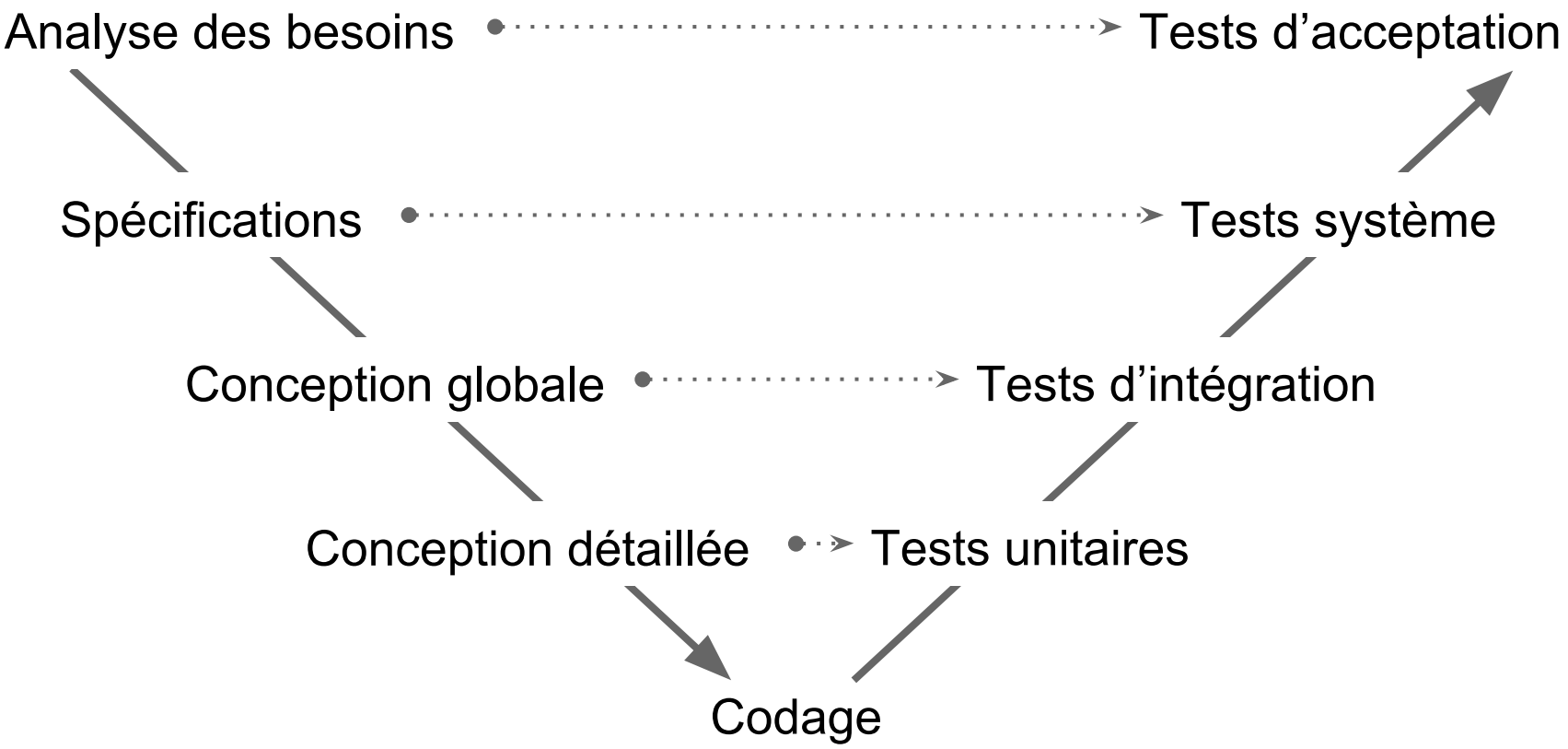
Vérification versus Validation.

- Validation: conformité des critères avec les attentes (externes). *Are we building the right product?*
- Vérification: conformité de la réalisation avec les critères (internes). *Are we building the product right?*

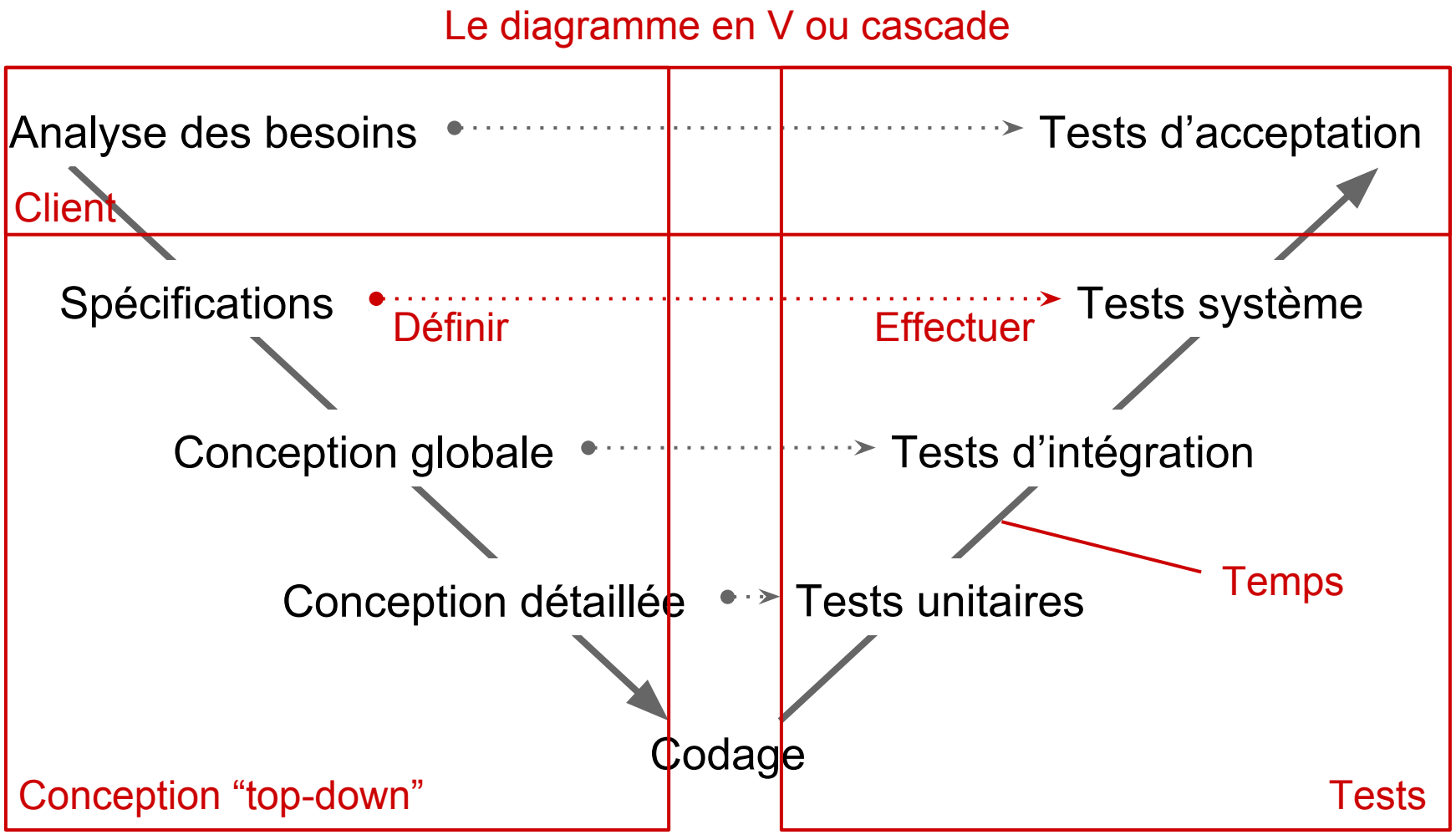
Rapport versus Dossier.

- Dossier: document à usage externe possible.
- Rapport: document interne a priori.

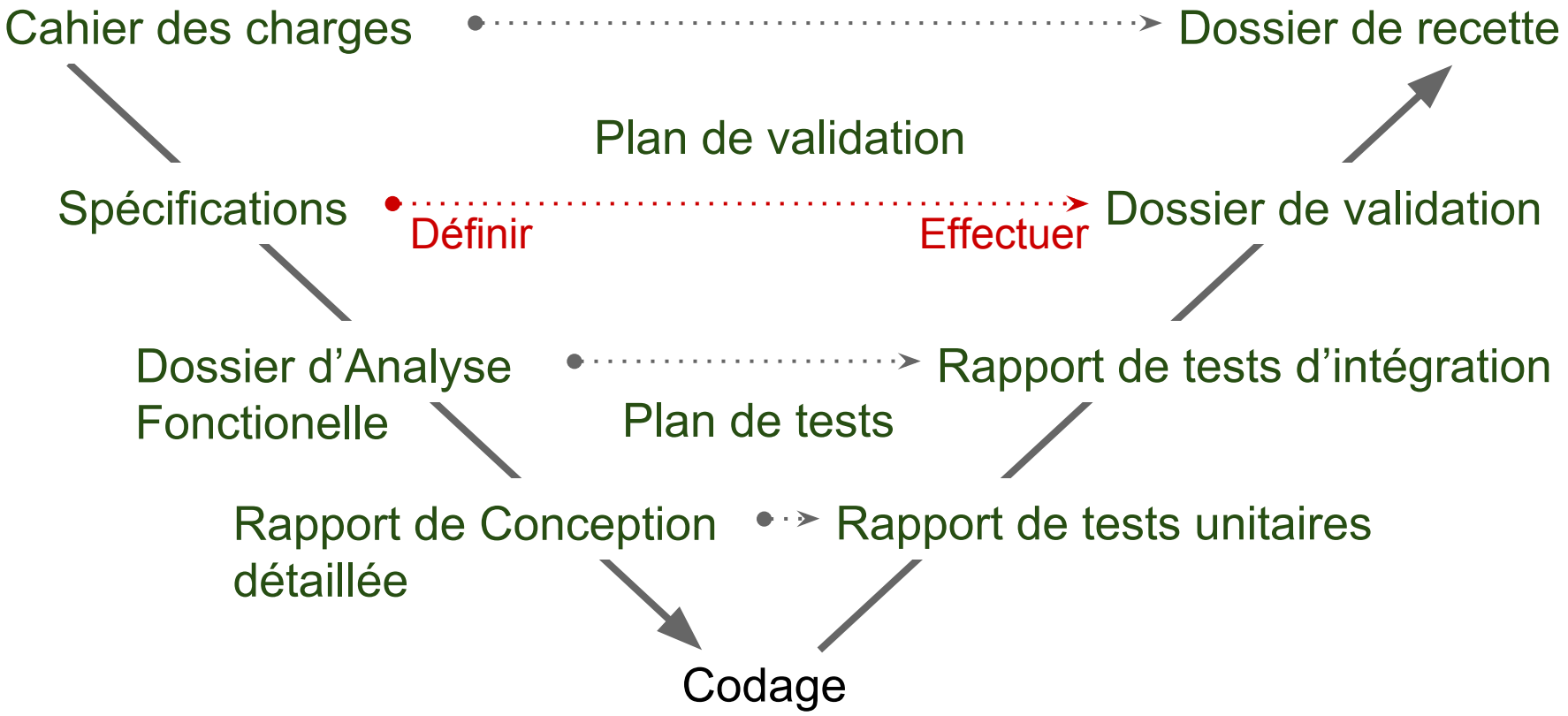
Documents méthode en V



Documents méthode en V



Documents méthode en V



* * *

Plan de validation

Plan de validation

A partir des spécifications. A concevoir avant codage.

- Plan de tests système: Tests dynamiques “boîte noire” à partir de données réelles ou simulées.
- Tests statiques: revue de code, de son architecture.
- Tests de robustesse, de compatibilité, de performance, de sécurité

Comment s’assurera-t-on que le code répond aux spécifications, et donc aux besoins (non-)fonctionnelles?

Plan de tests d’intégration

Plan de tests d'intégration

A partir du Dossier d'Analyse Fonctionnelle.

Concevoir:

- Tests de non-régression
- Tests des interactions entre classes
- Tests dynamiques des fonctionnalités de base

Comment s'assurera-t-on que le code correspond aux grandes lignes de l'architecture choisie?

Plan de tests unitaires

Plan de tests unitaires

A partir du Rapport de conception détaillée.

Concevoir:

- Tests dynamiques “boîte blanche” sur chacun des cas non-triviaux des méthodes de chacune des classes.

Comment s’assurera-t-on que chaque méthode de chaque classe est correcte?

Plan de tests

Plan de tests

A partir des plans de tests unitaires et d'intégration.

[Plan de tests unitaires.]

[Plan de tests d'intégrations.]

Quels seront les tests effectués sur le code?

Rapport de tests unitaires

Rapport de tests unitaires

A partir du plan de tests unitaires et du code.

- Méthodologie: stratégie, cas.
- Résultats: fiches / tableau d'anomalies...
- Niveau de couverture du code
- Conclusions

Estimer la correction de chaque méthode de chaque classe.

Rapport de tests d’intégration

Rapport de tests d'intégration

A partir du plan de tests d'intégration et du code.

- Méthodologie: stratégie, non-régression, scénarios.
- Fiches de faits techniques (anomalie, incidents, gravité…)
- Conclusions

Estimer la correction des interactions entre les classes, et combien le code correspond aux grandes lignes de l'architecture choisie.

Dossier de vérification

Dossier de vérification

A partir des rapports de tests unitaires et d'intégration.

[Rapport de tests unitaires.]

[Rapport de tests d'intégrations.]

Expliquer l'ampleur et la rigueur des tests effectués.

Estimer la correction du code.

Dossier de validation

Dossier de validation

A partir des Spécifications, Plan de validation, du Dossier de tests et du code. Présence du client souhaitable.

- Tableau d'adéquation du code aux spécifications
- Écart planning théorique versus planning réel.
- Méthodologie de comparaison
- Fiches de faits techniques (anomalie, incidents, gravité…), saisies d'écran.
- [Dossier de tests], [Manuel utilisateur]

Résumer le travail accompli et son niveau de validation, en relation avec les spécifications et donc les attentes.

Dossier de recette

Dossier de recette

A partir du Dossier de validation et du code.

Effectué par le client / les utilisateurs:

- Observations: installation, utilisation, revue de code
- Fiches de faits techniques (anomalie, incidents, gravité…)
- Tableau de couverture des cas d'utilisation
- Bilan du projet

Conformité du travail accompli, en comparaison avec le cahier des charges.

Analyse des risques

Analyse des risques

A partir des spécifications.

- Repérage des dépendances techniques
- Evaluations des difficultés techniques possibles
- Évaluation des menaces économiques possibles
- Lesquelles sont en amont des tâches prioritaires?
- Tableau

Repérer au plus tôt les points qui risquent d'être bloquants pour une partie centrale du projet, afin de s'y attaquer en priorité.

Planning

Planning

A partir des spécifications et de l'analyse de risque.

- Répartition des tâches, responsabilités
- Cas d'utilisations prioritaires d'abord, et risques associés
- Diagramme de Gant, Réseau PERT
- Tableau des livrables, jalons, réunions de fin de phase, revues de code...

Planifier et répartir le travail de façon efficace et cohérente.

Planning

Ex:

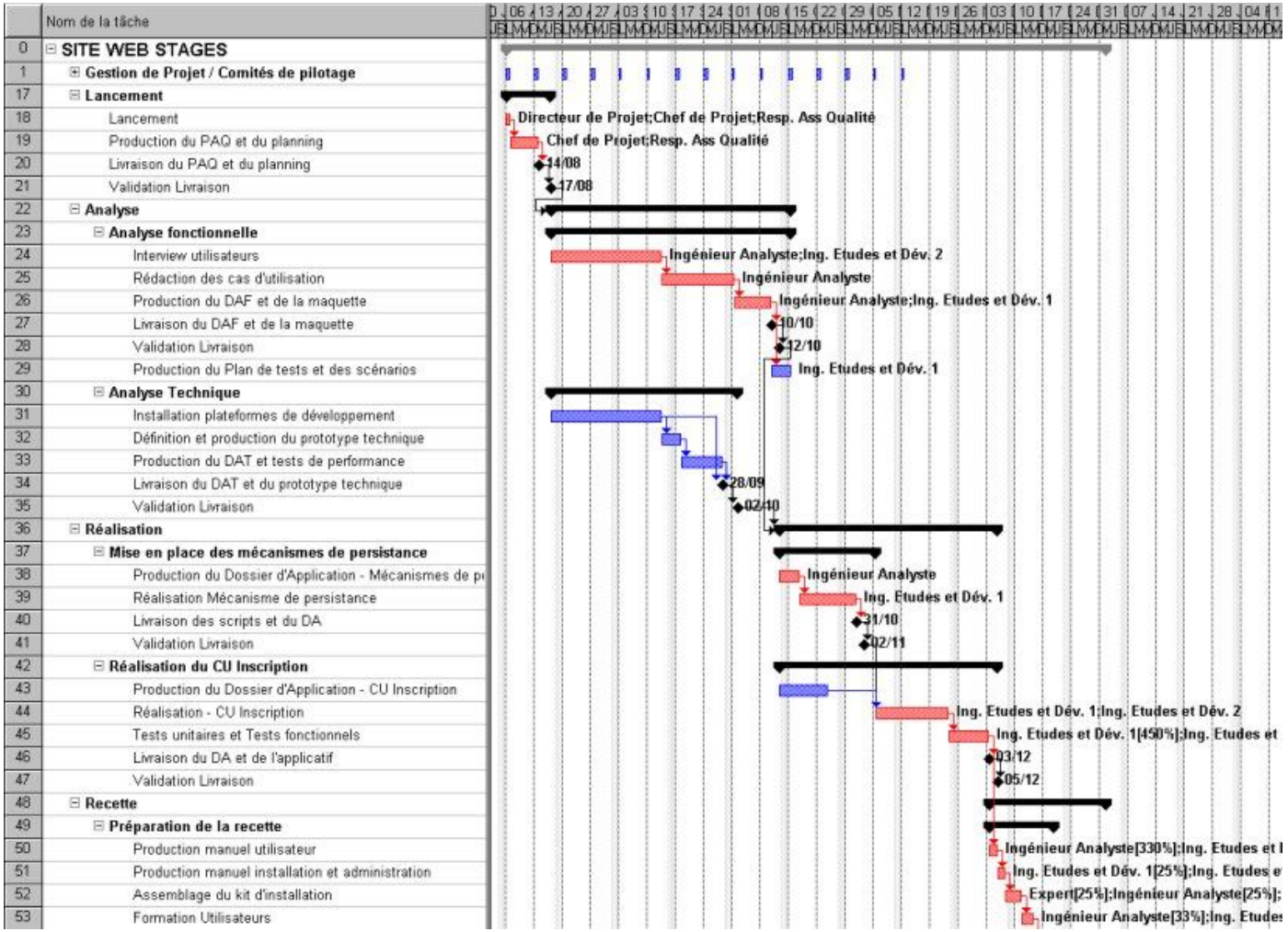

Nom de la tâche
0 SITE WEB STAGES
1 Gestion de Projet / Comités de pilotage
17 Lancement
18 Lancement
19 Production du PAQ et du planning
20 Livraison du PAQ et du planning
21 Validation Livraison
22 Analyse
23 Analyse fonctionnelle
24 Interview utilisateurs
25 Rédaction des cas d'utilisation
26 Production du DAF et de la maquette
27 Livraison du DAF et de la maquette
28 Validation Livraison
29 Production du Plan de tests et des scénarios
30 Analyse Technique
31 Installation plateformes de développement
32 Définition et production du prototype technique
33 Production du DAT et tests de performance
34 Livraison du DAT et du prototype technique
35 Validation Livraison
36 Réalisation
37 Mise en place des mécanismes de persistance
38 Production du Dossier d'Application - Mécanismes de p
39 Réalisation Mécanisme de persistance
40 Livraison des scripts et du DA
41 Validation Livraison
42 Réalisation du CU Inscription
43 Production du Dossier d'Application - CU Inscription
44 Réalisation - CU Inscription
45 Tests unitaires et Tests fonctionnels
46 Livraison du DA et de l'applicatif
47 Validation Livraison
48 Recette
49 Préparation de la recette
50 Production manuel utilisateur
51 Production manuel installation et administration
52 Assemblage du kit d'installation
53 Formation Utilisateurs
Directeur de Projet;Chef de Projet;Resp. Ass Qualité
Chef de Projet;Resp. Ass Qualité
14/08
17/08
Ingénieur Analyste;Ing. Etudes et Dév. 2
Ingénieur Analyste
Ingénieur Analyste;Ing. Etudes et Dév. 1
10/10
12/10
Ing. Etudes et Dév. 1
28/09
02/10
Ingénieur Analyste
Ing. Etudes et Dév. 1
31/10
02/11
Ing. Etudes et Dév. 1;Ing. Etudes et Dév. 2
Ing. Etudes et Dév. 1[450%];Ing. Etudes et
03/12
05/12
Ingénieur Analyste[330%];Ing. Etudes et
Ing. Etudes et Dév. 1[25%];Ing. Etudes e
Expert[25%];Ingénieur Analyste[25%];
Ingénieur Analyste[33%];Ing. Etudes

Tableau de bord

Tableau de bord

- Tableaux d'avancement vis-à-vis du planning et des plans de tests.
- Mise à jour de l'analyse de risque. Problème-Décision.
- Les livraisons, leurs configurations, les demandes d' évolution.
- Compte-rendu de réunions avec le client; questions-réponses impactant les spécifications; suivi de sa satisfaction.

Suivi du travail, traces écrites des interactions avec le client.

Tableau de bord

Ex: Systèmes de jetons / tickets ou “Trackers” disponibles avec Bugzilla, Github, Tuleap...

Ex:

	Plan de gestion et de suivi des risques						
N° Date créat.	**Niveau Prob %**	**Facteurs de risques identifiés *Risques (en italiques)***	**Indicateurs de suivi**	**Couverture des risques**	**Resp. Date**	**Actions contingentes**	**Evolution par rapport à la dernière mise à jour (↗ ➔ ↘)**

TABLEAU RECAPITULATIF DES FICHES QUESTION / REPONSE						
Référence	Date	Emetteur	Interlocuteur	Question	Date réponse	Prise en compte ? si oui, où ? si non, pourquoi ?

TABLEAU RECAPITULATIF DES COURRIERS EMIS					
U-AQT/TMA ACL/Piloter le projet/Tbb Courrier émis v01.doc					
N° ordre	Référence	Date	Emetteur	Destinataire	OBJET

TABLEAU RECAPITULATIF DES COURRIERS ET PRODUITS RECUS							
N° ordre	Référence externe	Date	Emetteur	Destinataire	OBJET	Relecteur ou testeur	Résultat relecture ou tests

Tableau de bord

Ex:

TABLEAU DE BORD DES RETOURS EN TESTS D'INTEGRATION ET EN RECETTE									
N° Anomalie	Date réception	Emetteur	DESCRIPTION ET COMMENTAIRE	Qual NQual	Bloq NBloq	Inter-venant	Temps passé	Résolu le	Renvoyé le

TABLEAU DE BORD DES EVOLUTIONS							
N° Evolution	Date réception	Emetteur	DESCRIPTION ET COMMENTAIRE	Inter-venant	Temps passé	Résolu le	Renvoyé le

TABLEAU DE BORD DES LIVRAISONS SORTANTES					
Référence de la livraison	Liv /Reliv	Date livraison	Livreur	Date acceptation client	Contenu de la livraison

Ressource	Manuel Ribeiro			janvier-03																		
Tâches	Prév. Tot.	Cons. Tot.	R.A.F.	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Evolution n°1	6,5	5,15	1,35			1,00	1,00	1,00				0,50	0,90	0,50									0,25
Anomalie n°3	2	0,6	1,4						0,50				0,10										
Anomalie n°5	5	4	1											0,50	1,00				1,00	1,00	0,50		
Anomalie n°2		1,75	-1,75																		0,50	1,00	0,25
Evolution n°6		0,5	-0,5																				0,50
Evolution n°5	6	3,5	2,5																				
Anomalie n°8		0,6	-0,6																				
Anomalie interne n°2		0,1	-0,1																				
Anomalie n°6		0,3	-0,3																				
		0	0																				
		0	0																				

Plan de développement

Plan de développement

A partir de l'analyse de risque, du planning, du tableau de bord.

- [Analyse de risque]
- [Planning]
- [Tableau de bord]

Contrôle du déroulé temporel du projet.

Réunions

Réunions : externes

Réunions client

- Établissement du Cahier des charges
- Validation des spécifications
- Recette (rendu)
- A chaque itération

Envoyer ordre du jour au moins 24h avant.

Envoyer un compte-rendu au plus tard 24h après.

Réunions : internes

Réunions du Comité de Pilotage (interne)

- Avant chaque réunion externe, pour établir ordre du jour.
- Organisation, planning
- Bilans et évaluations, validation des documents
- A chaque itération

Réunions : Ex. d'ordre du jour

- Validation d'une fiche d'itération
- Actions à mener lors d'une itération
- Avancement
- Suivi des anomalies
- Suivi des livraisons (prototypage, déploiement, manuel)
- Besoins futurs
- Indicateurs de qualité et satisfaction client
- Incidents du projet nécessitant un arbitrage
- Actions à mener pour la prochaine réunion
- Dates des prochaines réunions

Plan assurance qualité

Plan assurance qualité

A partir du cahier des charges.

- Exigences de tests, de planification, de suivi, de validation, des standards d'écriture de code, des métriques de qualité, exigences de documentation
- Responsabilités

Définition des exigences de qualité et des méthodes qui seront mises en oeuvre pour les assurer.

Documents méthode en V

Devis - Contrat - Facturation - Plan Assurance Qualité -

Analyse des risques - Plan de développement -

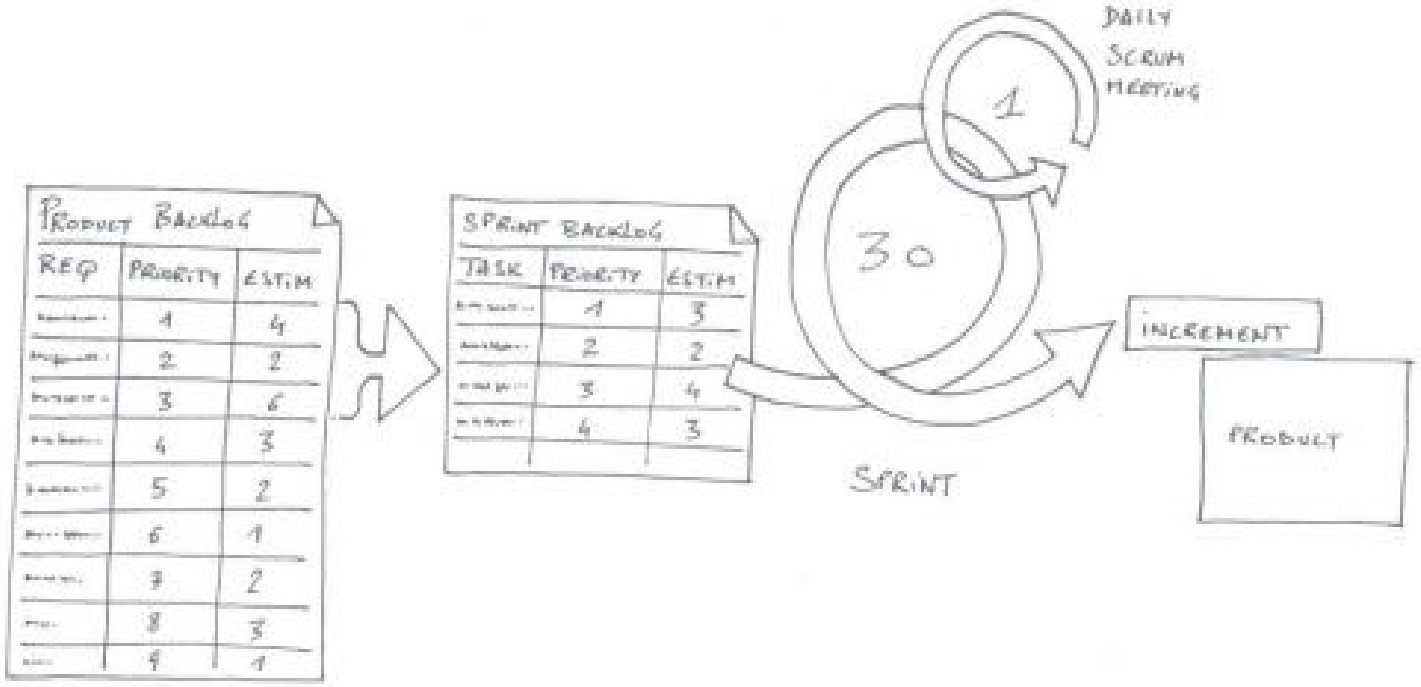
Tableau de bord et...

- Cahier des charges > Spécifications > Plan de Validation
- Dossier d'Analyse Fonctionnelle > Plan de tests d'Intégration
- Rapport de Conception Détaillée > Plan de tests Unitaires
- Rapport de tests unitaires/d'Intégration
- Dossier de tests, Dossier de validation

Plan de déploiement - Dossier de recette - Manuel utilisateur - Documentation API

* * *

Rappel : méthodes adaptives



Documents méthodes adaptives

Devis - Contrat - Facturation - Plan Assurance Qualité -

Analyse des risques - Plan de développement -

Tableau de bord et ...

PIRE CAS

- Cahier des charges > Spécifications > Plan de Validation
- Dossier d'Analyse Fonctionnelle > Plan de tests d'Intégration
- Rapport de Conception Détaillée > Plan de tests Unitaires
- Rapport de tests unitaires/d'Intégration
- Dossier de tests, Dossier de validation.

= Fiche d'itération x Nombre d'itérations .

Plan de déploiement, Dossier de recette, Manuel utilisateur, Documentation API

Fiche d’itération

Fiche d’itération

Voir ProjetsPhase2Template.

Documentation API avec Javadoc

Documentation API avec Javadoc

```
/**
 * Returns an Image object that can then be painted on the screen.
 * The url argument must specify an absolute {@link URL}. The name
 * argument is a specifier that is relative to the url argument.
 * <p>
 * This method always returns immediately, whether or not the
 * image exists.
 *
 * @param  url  an absolute URL giving the base location of the image
 * @param  name the location of the image, relative to the url argument
 * @return      the image at the specified URL
 * @see         Image
 */
public Image getImage(URL url, String name) {
       try {
           return getImage(new URL(url, name));
       } catch (MalformedURLException e) {
           return null;
       }
}
```

Documentation API avec Javadoc

```
/**
 * Returns an Image object that can then be painted on the screen.
 * The url argument mu
 * argument is a speci
 * <p>
 * This method always
 * image exists.
 *
 * @param  url  an abs
 * @param  name the lo
 * @return      the im
 * @see         Image
 */
public Image getImage
        try {
            return get
        } catch (Malfo
            return nul
        }
}
```

getImage

public Image getImage(URL url, String name)

Returns an `Image` object that can then be painted on the screen. The `url` argument must specify an absolute URL. The `name` argument is a specifier that is relative to the `url` argument.

This method always returns immediately, whether or not the image exists.

Parameters:

`url` - an absolute URL giving the base location of the image.

`name` - the location of the image, relative to the `url` argument.

Returns:

the image at the specified URL.

See Also:

`Image`

Documentation API avec Javadoc

`@author`	Nom du développeur
`@deprecated`	Marque la méthode comme dépréciée. Certains IDEs créent un avertissement à la compilation si la méthode est appelée.
`@exception, @throws.`	Documente une exception lancée par une méthode
`@param`	Définit un paramètre de méthode. Requis pour chaque paramètre.
`@return`	Documente la valeur de retour. Ce tag ne devrait pas être employé pour des constructeurs ou des méthodes définies avec un type de retour *void*
`@see`	Documente une association à une autre méthode ou classe.
`@since`	Précise à quelle version de la SDK/JDK une méthode a été ajoutée à la classe.
`@version`	Donne la version d'une classe ou d'une méthode.

Documentation API avec Javadoc

Documenter puis...

```
cd ~/projectX/src/
javadoc -d ~/projectX/doc/ packagename1 packagename2
```

ou

Eclipse > Run*, puis passer sur la méthode.*

F2 pour pouvoir naviguer, ou Window > Views > Javadoc.

Ctrl-Click pour ouvrir sa déclaration.

Window > Preferences > Java > Compiler > Javadoc

Permet d'activer warnings ou erreurs en l'absence de documentation d'une API.

Wiki

Wiki

Collaborative documentation

Ex. GitHub