



Surfaces de révolution discrètes

Plan d'Assurance Qualité Logicielle

Date : 2015/2016
Versions : 1.x
Statut : diffusable

Auteur :
Participants :

Zied Ben Othmane
Thomas Benoist
Adrien Bisutti
Lydie Richaume

Tables des révisions

Version	Date	Modifications
1.1	23/12/2015	Génération de surface à partir de courbes préchargées et non modifiables
1.2	21/01/2016	Choisir des courbes parmi des listes prédéfinies et navigation dans l'espace 3D
1.3	29/01/2016	Accès aux paramètres simples des courbes prédéfinies et le traçage à main levée
1.4	19/02/2016	Possibilité de rentrer une équation pour la méridienne et d'offrir des différents formats d'exploration des courbes et de la surface
1.5	02/03/2016	Implémentation de toutes les fonctionnalités du cahier de charge

Table des matières

1. But, portée, responsabilité	4
1.1. Objectif du document	4
1.2. Portée du document	4
1.3. Responsabilités associées	4
1.4. Procédure d'évolution du PAQL	4
2. Documentation utilisée	4
2.1. Documents de référence	4
3. Terminologie	5
4. Organisation	5
4.1. Maîtrise d'ouvrage	5
4.2. Maîtrise d'œuvre	6
4.3. Chef de projet	6
4.4. Responsable qualité	6
5. Cycle de vie	7
5.1. Motivation du choix	8
5.2. Description des étapes du cycle de vie	8

6.Méthodes, règles, normes	9
6.1. Règles et normes à respecter.....	9
7.Les fiches de la qualité : Méthodes, Métriques, Calcul.....	9
7.1. Les fonctionnalités.....	9
7.2. L'efficacité.....	12
7.3. Ergonomie	14
7.4. La fiabilité	15
7.5. La maintenabilité	17
8.évaluation de la qualité logicielle	21
8.1. Résultat.....	21
8.2. Analyse et discussion des résultats obtenus.....	22
9.Reproduction, protection, livraison	23
9.1. Procédure de reproduction.....	23
9.2. Protection du logiciel	23
9.3. Livraison et installation	23

1. But, portée, responsabilité

1.1. Objectif du document

Le plan d'assurance qualité (PAQL) a pour but de définir les méthodes et outils utilisés par le projet, ainsi que les mesures à prendre et les étapes pour contrôler et s'assurer de la qualité du projet.

Tout document produit sera soumis au contrôle de qualité. Il devra être conforme aux règles définies dans ce document. Tout document non conforme devra être corrigé.

1.2. Portée du document

Ce document est destiné :

- Aux clients : Éric Andres et Gaëlle Largeteau-Skapin
- Au responsable pédagogique : Philippe Meseure
- au jury du Master IMMT Info pour l'évaluation du projet
- à l'équipe du projet : Zied, Adrien, Thomas et Lydie

1.3. Responsabilités associées

Le responsable qualité est chargé de la rédaction du présent PAQL ainsi que de veiller à son application et son évolution, en collaboration avec le chef de projet. C'est à lui de décider des actions à entreprendre si le PAQL n'est pas appliqué.

1.4. Procédure d'évolution du PAQL

Le PAQL est élaboré au début du projet. A chaque étape, il est susceptible d'être modifié, auquel cas la modification sera indiquée dans la table de l'historique.

2. Documentation utilisée

2.1. Documents de référence

Le tableau suivant récapitule les principales sources documentaires qui seront utilisées dans le cadre de ce projet.

Document	Source	Outils
Courbe de cout	Cours de Jean François Renaut	Microsoft Project 2010
Diagramme de Gantt	Cours de Jean François Renaut	Microsoft Project 2010
Fiche de Risque	Cours de Jean François Renaut	Microsoft EXCEL 2010
Fiche de Taches	Cours de Jean François Renaut	Microsoft EXCEL 2010

<i>Document</i>	<i>Source</i>	<i>Outils</i>
Document de vision		
Descriptif de fonctionnalités		

3.Terminologie

Les termes suivant sont les termes spécifiques utilisés dans le PAQL.

- CdC : Cahier des Charges
- DCG : Dossier de conception globale
- DCD : Document de Conception Détaillée
- DSE : Dossier de Spécifications Externes
- PAQL : Plan d'Assurance Qualité Logiciel
- PDL : Plan de Développement Logiciel
- MOA : Maîtrise d'ouvrage
- MOAd : Maîtrise d'ouvrage déléguée
- MOE : Maîtrise d'oeuvre
- MOEd : Maîtrise d'oeuvre déléguée
- CV : Cycle de vie

4.Organisation

4.1. Maîtrise d'ouvrage

4.1.1.MOA

La maîtrise d'ouvrage représenté par les clients Éric Andres et Gaëlle Largeteau-Skapin qui représentent le département XLIM-SIC.

4.1.2. MOAd

La maîtrise d'ouvrage déléguée est représentée par Éric Andres, de l'équipe XLIM-SIC. Son rôle est de donner les objectifs de travail et de valider les résultats obtenus.

4.2. Maîtrise d'œuvre

4.2.1.MOE

La maîtrise d'œuvre est le Master2 IMMT spécialité Informatique de l'Université de Poitiers et ses représentants : Pascal Lienhardt et Samuel Peltier.

4.2.2. MOEd

La maîtrise d'œuvre déléguée est l'équipe de développement : Zied Ben Othman , Thomas Benois , Adrien Bisutti et Lydie Richaume . Leur rôle est de proposer une solution logicielle permettant la génération de surfaces de révolution discrètes et d'illustrer les publications des clients à ce propos, ainsi que de s'assurer de la bonne conduite du projet.

Pour cela, ils doivent faire en sorte que :

- Tous les documents demandés soient fournis
- Le planning soit observé et réajusté d'un commun accord si besoin
- Les normes de qualité soient définies et respectées
- La validation des documents soit conforme au PAQL
- La validation du code produit soit conforme au PAQL

4.3. Chef de projet

Le chef de projet sera Thomas Benois pour toute la période de réalisation du projet.

Le chef de projet est responsable :

- ✓ De la planification du projet
- ✓ Du contrôle de l'avancement du projet
- ✓ De la mobilisation des moyens nécessaires de la coordination des travaux de chacun

4.4. Responsable qualité

Le responsable qualité sera Zied Ben Othman toute au long la durée du projet.

Le responsable qualité a pour mission de :

- ✓ Définir les règles de retour arrière et les procédures de modification
- ✓ Veiller à la diffusion et au respect des règles particulières au projet
- ✓ Gérer l'évolution du PAQL
- ✓ Superviser les actions de vérification et de validation

5. Cycle de vie

5.1. Motivation du choix

Chaque attente du client peut être atteinte indépendamment des autres. L'utilisation d'un cycle de vie permettant de développer chacun des modules de bout en bout séparément est donc appropriée. Le produit final sera donc livré par lots successifs. Le cycle de vie ainsi choisi est le cycle de développement en spirale.

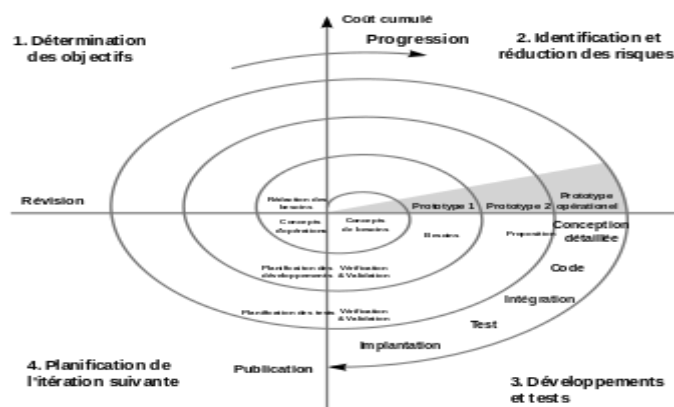


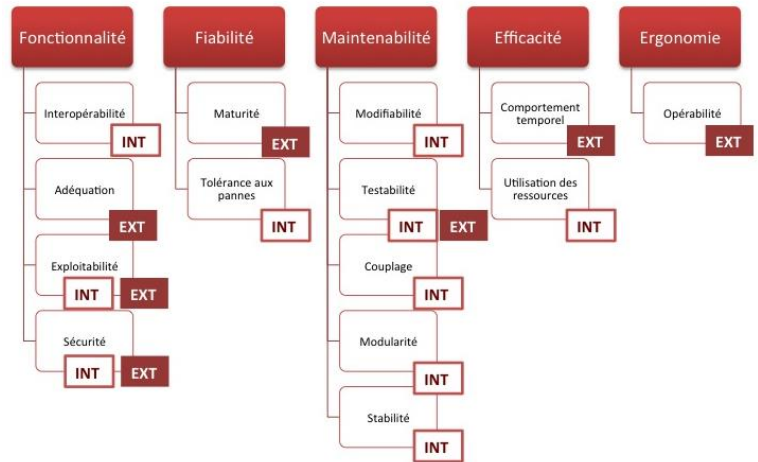
Illustration 1: Cycle de vie en spirale « Wikipedia »

5.2. Description des étapes du cycle de vie

- **Détermination des Objectifs** : Cette étape permet de prendre en compte les besoins et les exigences du client à chaque début d'une nouvelle itération, définir le planning et préparer l'acquisition des outils de programmation.
- **Identification des risques** : Cette étape permet d'identifier les risques rencontrés à chaque itération.
- **Développement et Test** : Cette étape commence par détailler la conception de chaque nouvelle version afin de pouvoir la développer ensuite de préparer des scénarii des tests unitaire et d'intégration et finira par une implémentation.
- **Planification de l'itération suivante** : Cette étape débutera par la livraison d'une version logicielle

6.Méthodes, règles, normes

La notion de qualité d'un logiciel est très large et assez vague. C'est pourquoi l'ISO en collaboration avec le SEI a défini la norme ISO/9126 afin de structurer cette notion. Cette norme caractérise la qualité de notre logiciel par un ensemble d'attributs résumé sur le schéma suivant (EXT signifie une qualité externe, c-à-d visible de l'utilisateur final et déduite du comportement à l'exécution du logiciel, INT une qualité interne, c-à-d non visible et déduite uniquement des caractéristiques du logiciel) :



7.Les fiches de la qualité

7.1. Les fonctionnalités

La fonctionnalité Désigne la capacité d'un produit logiciel à fournir les fonctions qui répondent aux besoins formulés et nécessaires quand le logiciel est utilisé dans des conditions spécifiées. Désigne ce que fait le logiciel pour remplir les besoins utilisateurs.

1	Adéquation			
	Besoins	Fonctions	Evaluation	Moyenne
Niveau 1	Génération 3D	1	1	100
	Affichage des courbes	1	1	
Niveau2	Choix de la connexité	1	1	75
	Choix des dimensions de l'espace 3D	1	1	
	Mouvement de caméra	1	0.5	
	Mise en évidence d'une méridienne/courbe de révolution	1	1	
	Choix des courbes parmi des modèles	1	1	
	Dessin à main levée de la méridienne	0	0	
	Modification des paramètres des courbes	1	1	
	Options avancées pour les paramètres des courbes	1	0.5	
Niveau 3	Afficher/cacher les limites de l'espace 3D	0.5	0.5	37.5
	Traçage à main levé	0	0	
	Accès aux paramètres	0.5	0.5	
	Choix des dimensions d'affichage de l'espace 3D (multi coupes)	0.5	0.5	
	Affichage du repère 3D	1	1	
Niveau 4	Afficher/cacher la grille de repérage des courbes	1	1	100
	Export en PNG des courbes et de la surfaces	1	1	
	Réglage de la taille d'affichage des voxels	1	1	
Niveau 5	Affichage de l'espace 3D en vue orthographique/perspective	1	1	95.83
	Entrer une équation	1	1	
	Sauvegarde des courbes	1	1	
	Chargement des courbes	1	0.5	

	Ajout de courbe prédéfinie	1	1	
	Choix de la langue	1	1	

a. L'adéquation

L'adéquation désigne l'existence de fonctions adéquates pour les tâches requises. (Voir tableau ci-dessus)

b. L'interopérabilité

Désigne la capacité à interagir avec un ou plusieurs systèmes.

1	Interopérabilité	Evaluation				
	standards devraient être utilisé	Niveau 1	Niveau 2	Niveau 3	Niveau 4	Niveau 5
Les noms de variables	Les noms de variables commence par majuscule	0	0	0.5	0.5	0.5
	Les noms de fonctions commence par majuscule	1	1	1	1	1
	Espacement entre les opérateurs	1	1	1	1	1
	Tous les déclarations terminent par point-virgule	1	1	1	1	1
Object	accolade de début au même ligne que la déclaration	1	1	1	1	1
	espacement entre chaque valeur et propriété	1	1	1	1	1
	une ligne de code doit être inférieur à 80 caractère	1	0.5	0.5	0.5	0.5
Extension des fichiers	la forme du fichier est HTML	1	1	1	1	1
	L'extension du css est .css	1	1	1	1	1
	L'extension du JavaScript est .js	1	0.5	0.5	0.5	1
Total		9	8	8.5	8.5	9
	Moyenne	90	80	85	85	90

c. Exploitabilité

Désigne la capacité à exploiter correctement le système logiciel

0.5	Evaluation				
Exploitabilité	Niveau 1	Niveau 2	Niveau 3	Niveau 4	Niveau 5
L'existence des erreurs	0.5	0.5	0.5	1	1
Détection des bugs	1	1	1	1	1
possibilité de déployer de nouvelles versions	1	1	1	1	1
Interface claire	0.5	1	1	1	1
Efficacité des fonctions	0.5	0.5	1	1	1
code source bien annoté	0	0.5	0.5	1	1
Total	3.5	4.5	5	6	1
Moy	50	64.28	71.42	100	100

d. Sécurité

Cet attribut mesure le niveau de sécurité du différentes composant de logiciel lors de l'interaction avec l'utilisateur, vu que notre application sera produite sur la serveur du

Client, nous ne sommes pas capables alors de mesurer la sécurité du l'application. Ce qui donne l'attribution de zéro comme confession de mesure.

Tableau de mesure

1	fonctionnalités	Niveau 1		Niveau 2		Niveau 3		Niveau 4		Niveau 5	
		INT	EXT	INT	EXT	INT	Ext	INT	Ext	INT	Ext
	Interopérabilité										
Objectif	capacité à interagir avec un ou plusieurs systèmes										
Question	Est-ce qu'il utilise des normes et standards technique ?										
	Evaluation	90		75		85		100		95.83	
	Adéquation										
Objectif	Vérification de l'adéquation des taches au besoin										
Question	Est-ce que chaque fonction est adéquate au besoin client ?										
	Evaluation		100		80		37.5		85		90
	Exploitable										
Objectif	la capacité à exploiter correctement le système logiciel										
Question	A quel niveau le logiciel est exploitable ?										
	Evaluation	25	25		32.14	35.71	35.71		100		100
	Note I/E	76.66	83.33	75	74.76	60.35	36.60	100	92.5	95.83	95
	Fonctionnalité	79.99		74.88		48.47		96.25		95.41	

Métrique de mesure :

- Affectation des notes allant de zéro vers 1 à chaque mesure de moins bon vers le meilleur.
- Faire la somme de chaque critère puis le multiplier par son confession.

- Sommer tous les mesures critères puis les multiplier par la confession de l'attribut.

7.2. L'efficacité

Efficacité

1. Comportement temporelle

Désigne la capacité d'un produit logiciel à fournir des temps de réponses, de traitement et un débit appropriés quand ses fonctions sont utilisées sous certaines conditions.

1	Comportement temporelle				
scenarii de mesure	Conditions	temps de réponse	temps de traitement	débit approprié	Note
Génération 3D	Google chrome	0	0.5	1	
Déploiement des axes	Linux	0	0.5	1	
Visualisation de la méridienne	Firefox	0	0.5	1	
Version 1				Moyenne	50
Visualisation de l'espace 3D	Taille 60*60*40	0.5	0.5	1	
Mouvement de la courbe de révolution	Zoom 5X	1	1	1	
Modification des paramètre des courbe	Internet Explorer	0	0.5	1	72.22
Version 2				Moyenne	
Accès aux paramètres	Accès aux paramètres	1	1	1	
Pivotement du repère	360°	1	1	1	
Pivoter la surface de révolution	2 coupes par les sliders	0.5	0.5	0.5	
Version 3				Moyenne	83.33
Export de la CR	Extension PNG	1	1	1	
Affichage grille	Espace 3D	0.5	0.5	0.5	
Réglage de taille d'affichage	Changement des tailles des	1	1	1	

	cubes				
Version 4				Moyenne	83.33
Entrer une équation	Division par zero	0.5	1	1	
Sauvegarde	.Rar	1	1	1	
Changement de langue	anglais	1	1	1	
Version 5				Moyenne	94.44

2. Utilisation des ressources

Désigne la quantité et le type de ressources utilisés (CPU, disque, réseau, mémoire, etc.) et la durée associée quand ses fonctions sont invoquées.

Vu que notre application sera déployée sur un serveur donc nous ne sommes pas capables de mesurer les performances du serveur vu que ce dernier appartient au client. De ce fait on a affecté le coefficient zéro à ce critère.

Tableau de mesure :

0.5	Efficacité	Niveau 1		Niveau 2		Niveau 3		Niveau 4		Niveau 5	
		INT	EXT	INT	EXT	INT	Ext	INT	Ext	INT	Ext
	Comportement temporelle										
Objectif	la capacité d'un produit logiciel à fournir des temps de réponses sous des conditions										
Question	Quel est le temps de réponse du logiciel?										
Evaluation			50		72.22		83.33		83.33		94.44
	Note I/E										
	Efficacité	50		72.22		83.33		83.33		94.44	

Métrique de mesure :

- Affectation des notes allant de zéro vers 1 à chaque mesure de moins bon vers le meilleur.
- Faire la somme de chaque critère puis le multiplier par son coefficient.
- Sommer tous les mesures critères puis les multiplier par le coefficient de l'attribut.

7.3. Ergonomie

Ergonomie

1. Opérabilité

Désigne la capacité d'un produit logiciel à être compris, appris, utilisé et attrayant pour l'utilisateur

1	Question	Version 1	Version 2	Version 3	Version 4	Version 5
1	Vision globale	1	1	0.5	1	1
2	Facilité de trouver l'information	0.5	0.5	0.5	0.5	1
3	vitesse de réponse	0.5	0.5	0.5	1	1
4	utilité de l'information présente	0	0.5	0.5	1	1
5	choix de titre et rubrique et leurs signification	0.5	1	1	1	1
6	la complétude de l'information trouvé par rapport au besoin	1	0.5	1	1	1
7	Rapidité d'exécution	0	0.5	1	1	1
8	Taux d'erreurs	0.5	0.5	0.5	1	1
9	Manipulation de l'utilisation	1	1	1	0.5	0.5
10	Fiabilité de l'application	0	1	1	1	1
	Total	50%	70%	75%	90%	95%

Remarque à prendre dans la prochaine version : c'est noté

Tableau de mesure

1	Ergonomie	Niveau 1		Niveau 2		Niveau 3		Niveau 4		Niveau 5	
		INT	EXT	INT	EXT	INT	Ext	INT	Ext	INT	Ext
	Opérabilité										
Objectif	Mesure de l'interaction homme machine										
Question	Est-ce que le logiciel interagit bien avec l'utilisateur ?										
	Evaluation		50%		70%		75%		90%		95
	Note I/E		50%		70%		75%		90%		
	Fonctionnalité	50%		70%		75%		90%		95%	

Métrique de mesure :

- Affectation des notes allant de zéro vers 1 à chaque mesure de moins bon vers le meilleur.

- Faire la somme de chaque critère puis le multiplier par son confession.
- Sommer tous les mesures critères puis les multiplier par le confession de l'attribut.

7.4. La fiabilité

Fiabilité :

1. Maturité

Désigne la capacité d'un produit logiciel à éviter les pannes résultantes d'erreurs dans le logiciel.

Il s'agit de choisir un niveau bien approprié à la version développée puis annoté par un coefficient de mesure comme indiqué

1	Tolérance aux pannes		
Test	Type de Test	Visualisation d'erreur	Moyenne
Niveau 1	Affichage 3D	0.5	75
	Basculement des axes X Y et Z	1	
	Application de la courbe de révolution à la méridienne	1	
	Affichage des espaces 2D	0.5	
Niveau 2	Modification de dimension de l'espace	1	62.5
	Basculement de la caméra (changement de point de vue)	0	
	Traçage de la méridienne sans erreur	1	
	Modification de paramètres de la courbe	0.5	
Niveau 3			50
	Représentation des axes dans un repère 3D	1	
	Modification de dimensions d'affichage	0	
Niveau 4	Jeu sur la grille de repérage	0.5	87.5
	Exportation de la méridienne et la courbe de révolution	1	
	Réglage de la taille des cubes	1	
	Exportation PNG	1	
Niveau 5	L'introduction des équation erronées	0	75
	Changement de la langue de l'application	1	
	Sauvegarde des courbes en vue orthographique	1	
	L'introduction d'un modèle spécifique de la courbe	1	

20 Initial – niveau 1 Il n'y a pas de méthode formelle, ni de cohérence, ni de standard, sur la base desquels les systèmes seraient construits. Le processus de développement n'est pas maîtrisé, il n'y a pas de volonté ferme de le gérer. Le succès dépend essentiellement des efforts individuels et des compétences des développeurs. Les exigences de qualité, les plannings et les budgets sont en général, difficilement respectés.

40 Répétitif – niveau 2 Il y a un consensus dans l'organisme sur la manière dont les choses doivent être gérées, mais cela n'a été ni formalisé ni écrit. Un management de projet, fondé sur la réussite des projets précédents, a été mis en place. Le processus de développement est stabilisé, sous le contrôle d'une gestion rigoureuse des coûts et des délais.

60 Défini – niveau 3 Le processus de développement est formalisé, documenté et appliqué. Les revues sont menées avec rigueur et les configurations sont convenablement gérées. Une structure Qualité & Méthodes précise et met à jour régulièrement les procédures de l'organisme.

80 Géré – niveau 4 L'organisme a institué un processus formel de collecte d'informations métriques pour suivre et gérer son processus de développement ainsi que les systèmes résultants. Des indicateurs contrôlent le bon déroulement des projets et le respect des objectifs de qualité.

100 Optimisé – niveau 5 L'organisme exploite les mesures pour optimiser en permanence son processus de développement. L'organisme maîtrise un processus de correction des aspects qui seraient jugés insuffisants, à la lecture des indicateurs. Naturellement, tous les niveaux à partir du 2ème impliquent de définir avec précision et de gérer avec rigueur les processus de l'organisme.

1	Evaluation				
Maturité	Version 1	Version 2	Version 3	Version 4	Version 5
	20	40	60	80	100

1. Tolérance aux pannes

Désigne la capacité à maintenir le niveau de performance en cas d'erreur logiciel et de non-respect des interfaces d'interactions avec le logiciel.

Table de mesure

0.1	Fiabilité	Niveau 1		Niveau 2		Niveau 3		Niveau 4		Niveau 5	
		INT	EXT	INT	EXT	INT	Ext	INT	Ext	INT	Ext
	Maturité										
Objectif	la capacité d'un produit logiciel à éviter les pannes										
Question	Comment améliorer le processus de développement ?										
	Evaluation		20		40		60		80		100
	Tolérance aux pannes										
Objectif	la capacité à maintenir le niveau de performance en cas d'erreur logiciel et de non-respect des interfaces d'interactions avec le logiciel										
Question	A quel niveau le système reste plus ou moins opérationnels en cas de panne partielle logiciel ou matériel ?										
	Evaluation	75		62.5		50		87.5		75	
	Note I/E	75	20	62.5	40	50	60	87.5	80	75	100
	Fiabilité	47.5		51.25		55		83.75		87.5	

Métrique de mesure :

- Affectation des notes allant de zéro vers 1 à chaque mesure de moins bon vers le meilleur.
- Faire la somme de chaque critère puis le multiplier par son confession.
- Sommer tous les mesures critères puis les multiplier par la confession de l'attribut.

7.5. La maintenabilité

Maintenabilité

Désigne la capacité d'un produit logiciel à être modifié.

1. Modifiabilité

Désigne la capacité d'un produit logiciel à permettre une modification spécifiée d'être implémentée.

0.5	Processus	Description de fonctionnalité	influence sur d'autres fonctionnalités	influence sur le cout	Influence sur les risques	Influence sur le durée du projet	Total	1-Moy
-----	-----------	-------------------------------	--	-----------------------	---------------------------	----------------------------------	-------	-------

		s						
Version 1	1	Génération 3D	0	0	1	0,5	1,5	
	2	Rendu 3D	0	0	1	0,5	1,5	
	3	IHM	0	0,5	1	0,5	2	
total							5	58.34
Version 2	1	Connexité	0	1	0	0.5	1.5	
	2	Les paramètres	0	0.5	0.5	0	1	
	3	Camera	0.5	0.5	1	0.5	2.5	
	4	Choix de courbes	0	0	0.5	0	0.5	
total							5.5	65.63
Version 3	1	Accélération de la génération	0	0	0	0.5	0.5	
	2	Espace 2D	0.5	0.5	0.5	0	1.5	
	3	Zoom	0	0	0.5	0	0.5	
total							2.5	79.17
Version 4	1	Main levée	0	1	0.5	1	2.5	
	2	Affichage grilles	0	0	0	0.5	0.5	
	3	Sauvegarde des courbes	0	0	0	0	0	
total							3	75
Version 5	1	Entrer Formules	0	1	0.5	0.5	2	
	2	Visualisation des coupes	0.5	0.5	0	0.5	1.5	
	3	Chargement des courbes	0	1	0.5	0.5	2	
total							6.5	54.16

2. Testabilité

Désigne la capacité d'un produit logiciel à être validé par rapport à une spécification.

Les constats	Version1	Version2	Version3	Version4	Version5
Savoir l'état des objets testé	1	1	1	1	1

Les actions	Version1	Version2	Version3	Version4	Version5
Tester chaque composant appart	1	1	1	1	1

0.5	Testabilité	Version1	Version2	Version3	Version4	Version5
	Le contexte	1	1	1	1	1
	Tester la Conception	1	1	1	1	1
	Distinction entre objets	1	0.5	0.5	0.5	0.5
	Les Frameworks	0.5	0.5	1	1	1
	Tester les dépendances	1	0.5	0.5	1	1
	Test des cohésion	1	1	1	1	1
	Test de couplages	0,5	0.5	0.5	0.5	0.5
	Total	6	5	5.5	6	6
	Moy	85.71	71.42	78.57	85.71	85.71

3. Couplage

Le couplage est une propriété indiquant le niveau de dépendances entre composants d'un système.

C'est une propriété globale obtenant à partir du calcul de dépendances entrantes et sortantes pour chaque composant.

1	Evaluation				
Couplage	Version 1	Version 2	Version 3	Version 4	Version 5
nombre des composants	41	41	43	43	43
nombre de couplage entre composant	123(supposition 3 dépendance par classe)	123(supposition 3 dépendance par classe)	123(supposition 3 dépendance par classe)	123(supposition 3 dépendance par classe)	123(supposition 3 dépendance par classe)
nbr de couplage total = nbr_composant*nbr_composant	1681	1681	1849	1849	1849
Coeff	7.31	7.31	6.65	6.65	6.65

Plan d'Assurance Qualité Logicielle

4. Modularité

La modularité exprime la topologie de l'architecture avec le nombre de composants dépendants d'un composant en particulier. (Une classe valide est une classe qui contient ≤ 500 ligne de code)

1	Evaluation				
Modularité	V1	V2	V3	V4	V5
nombre de classes valide	37	37	37	37	37
nombre de classe	41	41	41	41	41
Coeff	90	90	90	90	90

5. Stabilité

Désigne la capacité d'un produit logiciel à éviter des effets imprévus suite à la modification du logiciel.

0.5	Stabilité	Evaluation			
Numéro de test	Scénario de test :supprimer un bout de code et noté la continuité de la fonctionnalité du logiciel				
	V1	V2	V3	V4	V5
1	1	0.5	0.5	0.5	1
2	0	0.5	0.5	0.5	0.5
3	0	0.5	0.5	0.5	0.5
Moyenne	33.33	50	50	50	66.66

Tableau de mesure :

1	fonctionnalités	Niveau 1		Niveau 2		Niveau 3		Niveau 4		Niveau 5	
		INT	EXT	INT	EXT	INT	Ext	INT	Ext	INT	Ext
	Modifiabilité										
Objectif	A quel niveau on peut modifier un process dans le logiciel ?										
Question	Moyenne de modifiabilité de chaque processus testé										
	Evaluation	29.17		32.81		39.58		37.5		27.08	
	Testabilité										
Objectif	la facilité d'un système à réaliser des tests										
Question	A quel niveau le										

	système accepte les tests										
	Evaluation	42.85	42.85	35.71	35.71	39.28	39.28	42.85	42.85	42.85	42.85
	Couplage										
Objectif	le niveau de dépendances entre composants d'un système										
Question	Quel est le taux de dépendance entre composant										
	Evaluation	7.31		7.31		6.65		6.65		6.65	
	Modularité										
Objectif	l'aptitude d'un logiciel à être composé de modules indépendants										
Question	est-ce que les composant seront réutilisable ?										
	Evaluation	90		90		90		90		90	
	stabilité										
Objectif	la capacité d'un produit logiciel à éviter des effets imprévus suite à la modification du logiciel										
Question	A quel niveau le logiciel est considéré comme stable										
	Evaluation	16.67		25		25		25		33.33	
	Note I/E	37.2	42.85	38.16	35.71	40.10	39.28	40.40	42.85	39.98	42.85
	Maintenabilité	40.025		36.93		79.38		83.25		41.41	

Métrique de mesure :

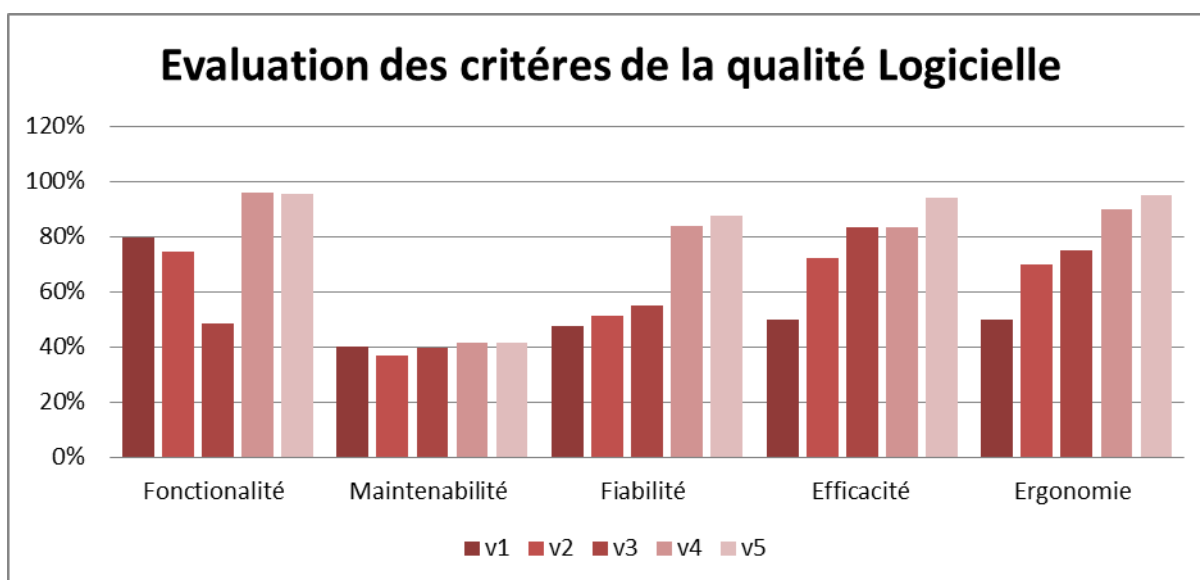
- Affectation des notes allant de zéro vers 1 à chaque mesure de moins bon vers le meilleur.
- Faire la somme de chaque critère puis le multiplier par son confession.
- Sommer tous les mesures critères puis les multiplier par le confession de l'attribut.

8.évaluation de la qualité logicielle

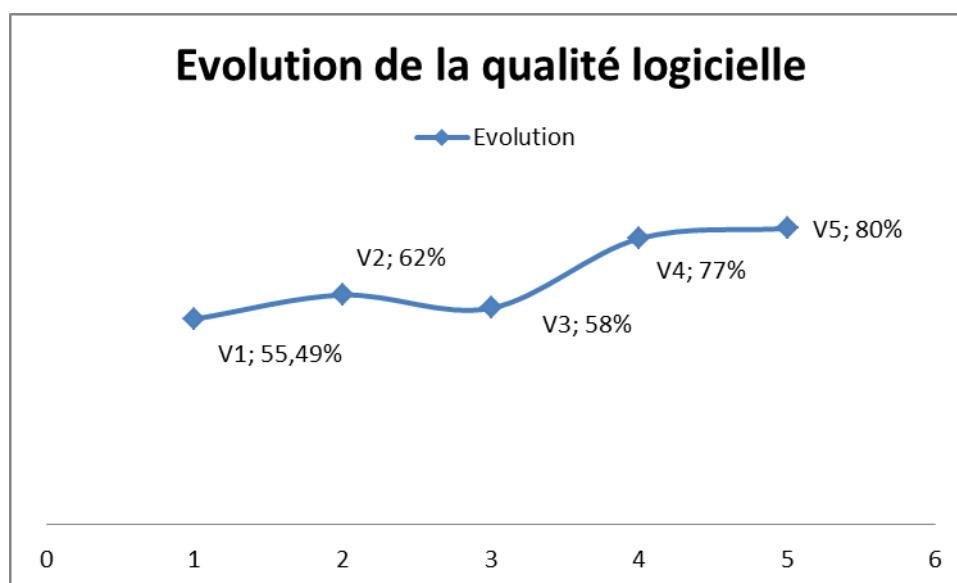
8.1. Résultat

La partie suivante présente les mesures qui seront effectuées pour quantifier le respect de chaque critère sélectionné. Les conditions de validation de chaque critère seront également précisées.

Evaluation des critères de la qualité Logicielle

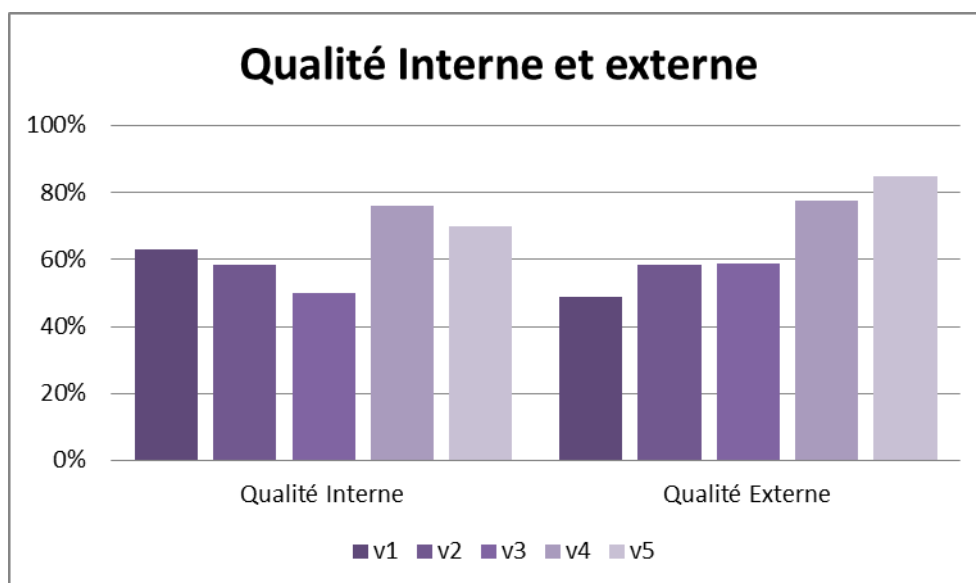


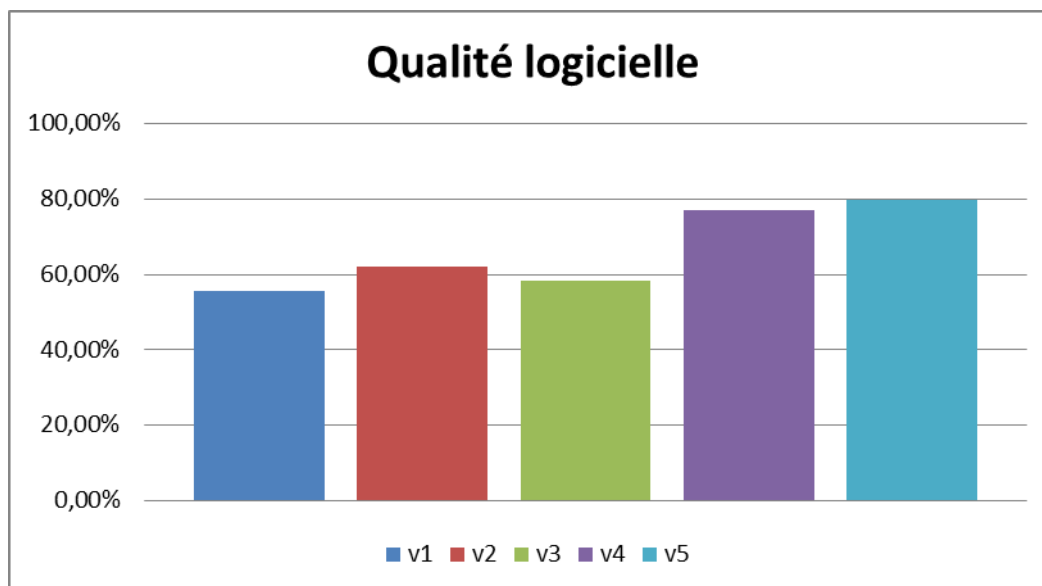
Evolution de la qualité logicielle



8.2. Analyse et discussion des résultats obtenus

Critère	coef	Version 1		Version 2		Version 3		Version 4		Version 5	
		Interne	Externe	Interne	Externe	Interne	Externe	Interne	Externe	Interne	Externe
Note I/E		76,66%	83,33%	75%	74,76%	60,35%	36,60%	100%	92,50%	95,83%	95%
Fonctionnalité	1	79,99%		74,88%		48,48%		96,25%		95,41%	
Note I/E		37,20%	42,85%	38,16%	35,71%	40,10%	39,28%	40,40%	42,85%	39,98%	42,85%
Maintenabilité	1	40,03%		36,94%		39,69%		41,63%		41,41%	
Note I/E		75%	20%	62,5%	40%	50%	60%	87,50%	80%	75%	100%
Fiabilité	0,1	47,50%		51,25%		55,00%		83,75%		87,50%	
Note I/E			50%		72,22%		83,33%		83,33%		94,44%
Efficacité	0,5	50%		72,22%		83,33%		83,33%		94,44%	
Note I/E			50%		70%		75,00%		90,00%		95%
Ergonomie	1	50%		70%		75%		90%		95%	
coef qualité E/I		63%	49%	58,55%	58,38%	50,15%	59%	75,97%	77,74%	70%	85%
coef qualité		55,49%		62%		58%		77%		80%	
				v1		v2		v3		v4	v5
			Fonctionnalité	80%		74,88%		48,48%		96,25%	95,41%
			Maintenabilité	40%		36,94%		39,69%		41,63%	41,41%
			Fiabilité	48%		51,25%		55,00%		83,75%	87,50%
			Efficacité	50%		72,22%		83,33%		83,33%	94,44%
			Ergonomie	50%		70%		75%		90%	95%
				v1	v2	v3	v4	v5			
		Qualité Interne	63%	58,55%	50,15%	75,97%	70%				
		Qualité Externe	49%	58,38%	59%	77,74%	85%				





9. Reproduction, protection, livraison

9.1. Procédure de reproduction

Aucune procédure de reproduction particulière n'est prévue.

9.2. Protection du logiciel

Ce logiciel est régi par la licence CeCILL soumise au droit français et respectant les principes de diffusion des logiciels libres.

9.3. Livraison et installation

Le logiciel final sera remis au client lors de la réunion de livraison le 02/03/2016. Il contiendra le code source de l'application.

Les documents livrables seront fournis dans un CD et seront déposés au secrétariat.

Ce logiciel sera remis au client auquel il s'occupe de son installation sur un serveur web.

Lors de la livraison du produit au client, ce dernier devrait signer la fiche de réception du produit.