

# 1. Plan Assurance Qualité

## 1.1. Objectifs et caractéristiques du PAQ

### 1.1.1. Objectifs

L'objectif de ce TER est de prendre en compte les informations ainsi que les indications fournies par nos encadrants et de les mettre en œuvre. Le développement de l'application s'appuiera sur les travaux de recherches et d'analyses effectuées en amont, mais aussi suivant les règles de développement énoncées dans ce Plan d'Assurance Qualité (noté PAQ).

Le but de ce document est de regrouper l'ensemble des dispositions envisagées par le groupe afin d'assurer la qualité de l'application que nous serons amenés à développer.

### 1.1.2. Domaine d'application

Ce plan d'assurance qualité vise à assurer le bon déroulement du TER. Ainsi, dans un premier temps, une interface permettant de visualiser des calculs de différents types d'automates sera mise en place, puis, selon l'avancée, différentes constructions d'automates seront implémentées.

Une documentation utilisateur et technique sera également fournie.

### 1.1.3. Responsabilités de réalisation et de suivi

#### 1.1.3.1. Mise en œuvre

La définition, la rédaction et la mise à jour du PAQ sont soumises aux membres du groupe chargés de l'assurance qualité du TER. Ainsi, après définition et rédaction de ce dernier, celui-ci sera soumis à différents approbateurs. Après la remonté de remarques ou de différentes suggestions de corrections, les personnes chargées du PAQ devront s'occuper d'apporter les modifications nécessaires.

Une fois le PAQ terminé et validé, celui-ci sera soumis aux encadrants pour une validation définitive.

L'ensemble des documents produits pour le TER sera composé des éléments listés ci-dessous :

- Plan d'assurance qualité (PAQ)
- Spécification des besoins
- Dossier d'analyse fonctionnelle
- Dossier de conception
- Dossier de recherche
- Dossier d'implémentation
- Annexes

#### 1.1.3.2. Acteurs

Acteurs	Rôle	Contact
Samir-Eddine AZZAG	Étudiant	<i>samir_azzag@yahoo.fr</i>
Amaury DOUDEMMENT	Étudiant	<i>amaury.doudement@gmail.com</i>
Didier MANNONE	Étudiant	<i>dmdinodm1@gmail.com</i>

Hannah SCHREIBER	Étudiant	<i>hannah.schreiber.k@googlemail.com</i>
Makki VOUNDY	Encadrant	<i>makki.voundy@lif.univ-mrs.fr</i>
Sèverine FRATANI	Encadrant	<i>severine.fratani@lif.univ-mrs.fr</i>

Étant donné le fait que tout au long de la réalisation du TER, la ou les personnes affectées à une tâche peut varier, le rôle de chaque acteur ne peut être prédéfini. Cependant, on peut d’ores et déjà différencier :

- Les Encadrants : s’assurent de la bonne compréhension du sujet et de sa mise en œuvre.
- Les Développeurs : sont chargés d’implémenter les différents concepts abordés.
- Les Analystes : définissent les différents besoins du client et en extraient les informations pertinentes.
- Documentalistes : s’occupent de la rédaction des divers documents.

### 1.1.4. Documents applicables

#### 1.1.4.1. Mise en œuvre

La définition et la rédaction de chaque document devra être réalisée par un ou plusieurs membres du groupe. Une fois la rédaction terminée, les membres du groupe devront soumettre leur travail aux approbateurs. Ces derniers devront s’assurer de la bonne qualité du travail réalisé, puis émettre d’éventuelles remarques et suggestions de modifications.

Enfin, les auteurs du document devront prendre en compte les retours apportés par les approbateurs et seront chargés d’effectuer les corrections nécessaires.

#### 1.1.4.2. Convention de nommage des documents

Les différents documents seront soumis à une nomenclature particulière. Chacun devra être nommé de la manière suivante :

**[nom\_du\_document]\_[version].pdf**  
**Exemple : PAQ\_v1.0.pdf**

### 1.1.5. Procédure d’évolution du PAQ

Toute modification ultérieure de ce plan d’assurance qualité devra être justifiée par l’apparition d’une contrainte supplémentaire ou omise.

#### 1.1.6. Procédure de dérogation du PAQ

Les différents membres du groupe devront suivre les différentes règles énoncées dans ce plan d’assurance qualité. En cas de dérogation, les autres membres du groupe devront indiquer en quoi la personne s’est soustraite à une règle afin que celle-ci puisse y remédier.

## 1.2. Terminologie

### 1.2.1. Glossaire

**Java** : Langage de programmation orienté objet

### 1.2.2. Abréviation

**PAQ** : Plan Assurance Qualité

**IDE** : Environnement de développement Intégré (*Integrated Development Environment*)

## 1.3. Objectifs et engagements qualité

### 1.3.1. Objectifs et critères

L'objectif principal du PAQ est de garantir une application de qualité qui réponde aux attentes des encadrants. Pour ce faire, l'accent sera mis sur :

- L'efficacité
- La fiabilité
- La facilité d'utilisation
- La portabilité
- La maintenabilité

### 1.3.2. Mesure de qualité

Critères	Engagements	Solutions	Métriques
<b>Efficacité</b>	Garantir une application programmée de manière efficace (n'exigeant pas l'utilisation de ressources inutiles)	S'assurer de l'optimisation du code développé	L'application devra fonctionner sur tout type d'ordinateur. Elle ne devra pas nécessiter une configuration matérielle importante
<b>Fiabilité</b>	Garantir une application fiable exempte de bugs	Mettre en place des phases de test	L'application devra avoir subi un nombre suffisant de tests pour maximiser sa fiabilité
<b>Facilité d'utilisation</b>	Garantir une application facile d'utilisation	Proposer une interface simple et ergonomique	L'interface de l'application devra être suffisamment simple pour être accessible dès sa première utilisation
<b>Portabilité</b>	Garantir une application qui sera fonctionnelle sur plusieurs systèmes d'exploitation	Compiler l'application sur Linux et Windows	Afin de pouvoir être utilisée par un plus grand nombre d'utilisateur, l'application devra être compilée sur différents systèmes d'exploitation
<b>Maintenabilité</b>	Garantir une application capable d'évoluer et d'être modifiée facilement	Développer de manière normalisé	La dénomination des différentes fonctions et variables devra être normalisée. L'application devra être structurée correctement.

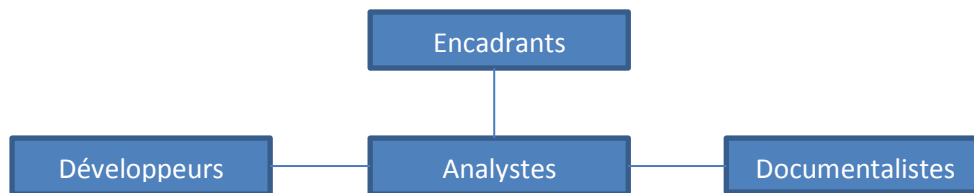
## 1.4. Conduite du projet

### 1.4.1. Organisation du projet

Chaque semaine, une réunion avec les encadrants aura lieu dans le but de présenter l'état d'avancement du projet, d'éclaircir certains points et de s'assurer de la bonne compréhension et mise en œuvre de ce dernier.

Chacune de ces réunions devra être préparée à l'avance et un plan de travail sera mis en place à l'issue de celles-ci.

### 1.4.2. Organisation de l'équipe



### 1.4.3. Planification

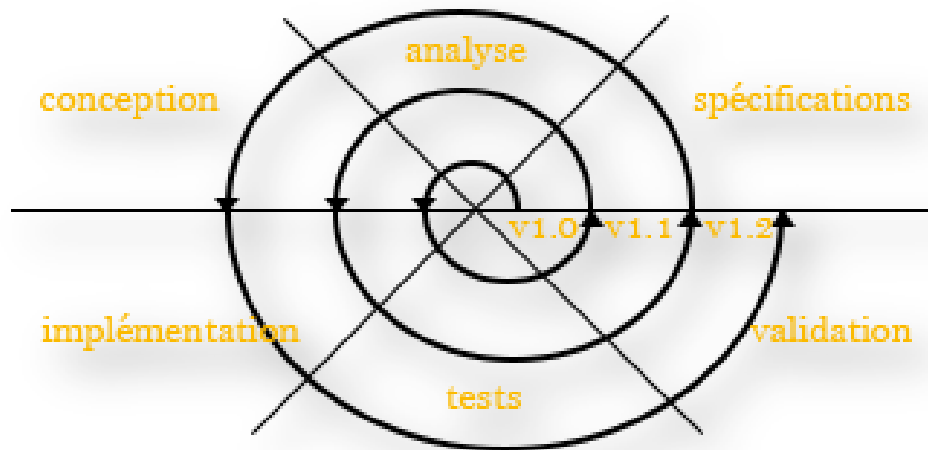
En plus de l'élaboration d'un plan prévisionnel de réalisation du TER en début de projet, chaque semaine, une distribution des tâches sera effectuée entre les différents membres du groupe. Chaque membre du groupe devra alors effectuer les tâches qui lui seront attribuées en respectant les dates d'échéances prédéfinies. Une fois les tâches terminées, le travail sera fourni aux autres membres du groupe chargés d'approuver les différents documents.

Un suivi dynamique sera effectué et, selon l'avancement, de nouvelles tâches pourront être attribuées.

## 1.5. Démarche de développement

### 1.5.1. Cycle de développement

Pour mener à bien le projet, la méthode de travail du cycle en spirale sera mise en place comme illustré ci-dessous :



### 1.5.2. Description des étapes

La méthode de développement du cycle de vie en spirale choisie pour réaliser le projet se décompose en six étapes distinctes :

- Spécifications : cette partie constitue l'étude des besoins du client et définit les objectifs et limites du projet.
- Analyse : cette partie constitue l'étude du projet et des différents risques. On y retrouve également la partie recherche.
- Conception : cette phase constitue la préparation des différents documents nécessaires à l'implémentation.
- Implémentation : implémentation des différentes fonctionnalités.
- Tests : Les différents modules développés seront testés.
- Validation : Fin du cycle, approbation du travail et livraison finale.

### 1.6. Gestion de la documentation

La rédaction des documents sera effectuée à l'aide du logiciel Microsoft Office Word 2013. Ces derniers seront également disponibles au format PDF (*Portable Document Format*).

Tous les documents seront mis à disposition et partagés avec les membres du groupe ainsi que les encadrants sur un serveur distant « dropbox ». Une copie de sauvegarde sera mise à disposition des membres sur « GitHub ».

### 1.7. Gestion du développement

#### 1.7.1. Analyse

#### 1.7.2. Conception

### 1.8. Charte de nommage

La totalité du code de l'application incluant les noms des méthodes, des classes et des attributs ainsi que les commentaires seront rédigé en anglais et dans le respect des normes et conventions de nommage relatives au langage de programmation utilisé « Java ».

Éléments	Norme
Indentation	une tabulation par niveau d'indentation (tabulation = 4 caractères)
Accolades	après la parenthèse et avant le retour à la ligne
Nom de classe	
Nom de méthodes	
Nom des attributs	
Nom des variables	
Longueur maximum	

\* La totalité du code (classes, méthodes, variables, informations supplémentaires) sera commenté tout au long de sa création pour faciliter la prise en main par un autre membre du groupe ainsi que sa compréhension et l'utilisation éventuelle de DoxyGen.

## 1.9. Gestion des modifications

### 1.9.1. Suivi des modifications

### 1.9.2. Modification des documents

### 1.9.3. Modification conceptuelles

A l'issue du développement, des modifications pourront être apportées aux fonctionnalités déjà produites. Ces dernières viseront à améliorer l'application, mais aussi à corriger celles-ci dans l'éventualité d'apparition de bugs. Compte tenu des différentes conventions énoncées dans la partie précédente, les modifications de l'application devraient être effectuées rapidement.

De plus, de nombreux tests devront être mis en place afin de s'assurer de la qualité du code produit. Ces tests pourront s'appuyer sur des fonctionnalités codées ou alors sur les performances du code produit.