

ÉCOLE POLYTECHNIQUE DE L’UNIVERSITE FRANÇOIS RABELAIS DE TOURS

Spécialité Informatique

64 av. Jean Portalis

37200 TOURS, FRANCE

Tél +33 (0)2 47 36 14 31

wwwpolytechuniv-tours fr

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Cahier De Spécification** | | | | |
| **Projet :** CDS03 | | Maquette détaillée d’uncahier de spécification | | |
| **Emetteur :** | | N. Ragot | | **MOA :**EPU-DI |
| **Date d’émission :** | | 28/10/2015 | | |
| **Validation** | | | | |
| Nom | | Date | Valide (O/N) | Commentaires |
| N. Ragot | | 28/10/2015 | O |  |
|  | |  |  |  |
|  | |  |  |  |
| **Historique des modifications** | | | | |
| Version | Date | Description de la modification | | |
| 00 | 11/2008 | Version initiale : synthèse de différents documents | | |
| 01 | 10/10/2009 | Refonte basée sur les docs de Génie Log (VTK) | | |
| 02 | 21/10/2009 | Correction suite aux relectures des EC-DI | | |
| 03 | 28/10/2015 | Mise à jour + séparation plan de développement | | |
|  |  |  | | |
|  |  |  | | |
|  |  |  | | |

Table des matières

Cahier de spécifications 5

1. Contexte de la réalisation 5

2. Description générale 5

3. Description des interfaces externes du logiciel 6

4. Architecture générale du système 7

5. Description des fonctionnalités 7

6. Conditions de fonctionnement 8

Glossaire 11

Bibliographie 13

Index 15

# Cahier de spécifications

Ce document énonce les spécifications systèmes concernant le projet : « Composant graphe de proximité sous Orange ».

Ce projet a été proposé par Gilles Venturini, enseignant chercheur au département informatique de l’école d’ingénieur polytechnique de l’université de Tours.

## Contexte de la réalisation

Orange est un logiciel gratuit et libre de droit proposant des outils d’analyse de données. Il permet aux utilisateurs de programmer à l’aide d’une interface de programmation visuelle ou de scripts Python. Il est développé en Python.

### Objectifs

Les objectifs de ce projet sont de comprendre comment ajouter un composant Python à Orange, comprendre les structures de données du logiciel pour représenter des tables et des matrices, ainsi que la notion de package de composants.

Il s’agit ensuite d’ajouter un composant pour construire un graphe de voisinage, à l’aide d’algorithmes simples, puis de l’agrémenter d’un composant graphique permettant d’afficher des graphes de type force-ressort. Ces deux composants doivent enfin être regroupés dans un package de composants et intégrés au logiciel Orange.

Enfin, nous souhaiterions, dans la mesure du possible, publier et intégrer notre package de composants à Orange.

### Hypothèses

Dans le cas où l’intégration de composants serait trop difficile à mettre en place ou si la récupération de données issues d’un composant n’est pas possible, alors nous programmerons des scripts pythons remplissant les mêmes fonctionnalités sans être intégrés à Orange.

Aussi, s’il existe déjà un composant d’affichage de graphes alors nous ferons en sorte d’adapter notre composant à ce dernier, sinon nous devrons l’implémenter nous même.

### Bases méthodologiques

Dans le cadre du développement puis de l’intégration d’un nouveau composant au logiciel Orange, il est pertinent de privilégier un OS Unix ou Unix-like. Nous avons donc décidé de réaliser ce projet sous le système d’exploitation Ubuntu 18.04.

Ensuite, les versions d’Orange se basant sur les mêmes numéros de version que python, nous avons décidé d’adopter la version la plus récente. Pour cette raison, nous utiliserons les librairies Python 3.6 afin de réaliser un composant sur Orange 3.

**//TODO rajouter bibliothèques python utilisées**

En ce qui concerne la méthode de gestion de projet, nous tenterons d’appliquer la méthodologie Scrum : il s’agit d’un dérivé de la méthode Agile. Cette méthode permet d’alterner entre des « sprints » (périodes de développement de 2 à 4 semaines) et des réunions avec le client, afin d’avoir un retour régulier sur le produit et d’ajuster au mieux le produit final au point de vue du client.

Cette méthode correspond à notre projet car nous aurons régulièrement besoin d’être aiguillés par notre enseignant référant afin de respecter au mieux ses attentes.

## Description générale

### Environnement du projet

Il s’agit d’un projet indépendant, regroupant divers composants qui devront parfois interagir avec les composants déjà existant dans le logiciel Orange. Dans l’idéal, nos composants viendront directement s’intégrer à Orange, en effet ce logiciel permet de facilement développer et intégrer de nouveaux packages de composants.

### Caractéristiques des utilisateurs

Ici, nous nous focaliserons sur un seul type d’utilisateur. En effet, nous souhaitons développer une extension pour le logiciel Orange. Nos utilisateurs sont donc ceux qui utilisent ce logiciel. Ainsi, ils ont déjà une connaissance avancée de l’informatique. Nous mettrons à leur disposition des outils pertinents s’intégrant au mieux avec les outils pré existants sous Orange.

### Fonctionnalités du système

L’utilisateur a accès aux fonctionnalités suivantes : création et affichage de graphes à l’aide de l’interface de programmation visuelle.

## Spécifications fonctionnelles

Il s’agit de l’expression des besoins fonctionnels. Cette partie a donc comme objectif de décrire l’ensemble des fonctions du système en précisant avec quels composants elles interagissent. Des diagrammes de cas d’utilisation plus détaillés, ainsi que l’arbre hiérarchique des fonctionnalités pourra être fourni ici pour donner une vision plus globale. En outre, chaque fonction sera décrite précisément(cf.ci-dessous). Là encore, il s’agit d’une pré-analyse indispensable à l’évaluation de la complexité de votre projet et à la planification de sa réalisation.

### Définition de la fonction i

Identification de la fonction i

Présenter la fonction :

* nom de la fonction ;
* rôle, présentation générale ;
* priorité associée à la réalisation de la fonction (primordiale, secondaire, facultative).

Description de la fonction i

Décrire précisément :

* les entrées et les sorties ainsi que les préconditions et postconditions déjà connues, uniquement sous forme textuelle et en langue naturelle(pas dans un pseudo langage algorithmique). Si ces E/S sont connectées à d’autres fonction ou interfaces, le préciser également ;
* les composants avec lesquels cette fonction interagie (données/composants utilisés/modifiés, etc. ;
* le traitement associé à la fonction et à ses interfaces. Il peut s’agir d’une explication ou d’un pseudo-algorithme général précisant les différentes étapes du traitement. Lors de l’analyse, ce dernier pourra être précisément représenté par un diagramme d’activité ; faire référence à un document/article précisant le fonctionnement le cas échéant ;
* si une gestion des erreurs spécifique (hors format des E/S) est prévue et comment celle-ci doit être mise en place si cela est déjà connu (notamment pour les fonctions sensibles).

## Spécifications non fonctionnelles

### Contraintes de développement et conception

Préciser les contraintes liées aux :

* + - * matériels : quelles sont les particularités du matériel qui vont contraindre le développement logiciel;
      * langages de programmation imposés ou adoptés ;
      * logiciels et bibliothèques à utiliser pour le développement;
      * environnements nécessaires : simulateurs, outils logiciels ;
      * bibliothèques de programmes imposées ;
      * protocoles de communication imposés : si nécessaire mettre en annexe une présentation de ces protocoles de communication;
      * etc.

### Contraintes de fonctionnement et d’exploitation

Il faut dans ce paragraphe décrire les dispositions qu’il est nécessaire de prendre en compte pour les différentes conditions de fonctionnement su systèmes.

#### Performances

Préciser en termes mesurables, les spécifications temps réel liées à l’utilisation du système :

* du point de vue de l’utilisateur :temps de réponse souhaité, fréquence d’utilisation, temps d’indisponibilité acceptable, etc. ;
* du point de vue de l’environnement : fréquence moyenne d’acquisition d’états ou de mesures, fréquence maximale d’E/S, etc.

#### Capacités

Décrire les limites des problèmes traitables par le système et les limites des éventuelles extensions comme par exemple :

* nombre max de terminaux ;
* nombre max de points d’acquisition ;
* nombre max de transactions simultanées de tel type, etc. ;
* capacité max de stockage ;
* taille max des données traitées ;
* etc.

#### Contrôlabilité

Il faut décrire, si elles existent, les spécifications particulières permettant de suivre l’exécution d’un traitement (fichier de log, niveaux d’affichages en mode debug, etc.)

# Glossaire

Dans cette partie on doit trouver, classés par ordre alphabétique, les définitions des termes courants utilisés, des termes techniques, abréviation, sigles et symboles employés dans l’ensemble du document.

# Bibliographie

**Aucune source spécifiée dans le document actif.**

Cette dernière partie recense les références techniques sur le projet sur :

* les documents relatifs à l’existant et à l’environnement ;
* les documents sur les méthodes et algorithmes cités ;
* les documents bibliographiques (internes et externes) ;
* les sources d’obtention des documents.

# Index

Cette partie indique les pages où sont traités et mentionnés les sujets et les termes les plus importants du document.