

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 :** GdP | | Cahier de spécification : Composant de graphe de proximité | | |
| **Emetteur :** | | Benoît Richard &  Thomas Rossi | | **MOA :** Gilles Venturini |
| **Date d’émission :** | | 28/10/2015 | | |
| **Validation** | | | | |
| Nom | | Date | Valide (O/N) | Commentaires |
|  | |  |  |  |
|  | |  |  |  |
|  | |  |  |  |
| **Historique des modifications** | | | | |
| Version | Date | Description de la modification | | |
| 00 | 16/10/18 | Première version du cahier de spécifications | | |
|  |  |  | | |
|  |  |  | | |
|  |  |  | | |
|  |  |  | | |
|  |  |  | | |
|  |  |  | | |

Table des matières

Cahier de spécifications 6

Contexte de la réalisation 6

Description générale 7

Spécifications fonctionnelles 7

Spécifications non fonctionnelles 7

Glossaire 9

Bibliographie 11

Index 13

# 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.

De plus, il existe déjà un package de composants nommé « Network Master » permettant de créer et visualiser des graphes. Nous allons donc nous appuyer sur ce projet afin de développer le notre, y ajouter les algorithmes demandés par notre client, etc.

### 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.

### Fonction principale I

#### Création de graphes via la méthode KNN

Cette fonction devra permettre à l’utilisateur de créer un graphe à partir des matrices de distance créées par le composant Orange3 associé. L’algorithme implémenté se nomme « K nearest-neighbor ». Il permettra de regrouper les différentes informations en liant chaque objet à ses K plus proches voisins.

Entrée requise :

* K, un entier : le nombre de voisins voulu ;
* La matrice de distance créée à partir du fichier de données fourni.

Cette fonction devra fournir des données exploitables par le composant d’affichage de graphe présent dans le projet « Network Master ».

### Fonction principale II

#### Création de graphes via la méthode RNG

Cette fonction devra permettre à l’utilisateur de créer un graphe à partir des matrices de distance créées par le composant Orange3 associé. L’algorithme implémenté se nomme « relative neighborhood graph ». Dans cet algorithme, pour deux points autour desquels on trace deux cercles de rayon R, on vérifie qu’il n’existe pas d’autre point se trouvant dans l’intersection des deux cercles. Si c’est le cas, alors ils sont connectés l’un à l’autre.

Entrée requise :

* R, un entier : le rayon du cercle ;
* La matrice de distance créée à partir du fichier de données fourni.

Cette fonction devra fournir des données exploitables par le composant d’affichage de graphe présent dans le projet « Network Master ».

## Spécifications non fonctionnelles

### Contraintes de développement et conception

Langage de programmation imposé : Python ;

Logiciels et bibliothèques à utiliser pour le développement : Projet Github « Network Master » <https://github.com/biolab/orange3-network>

{à rajouter : bibliothèques pythons}

### Contraintes de fonctionnement et d’exploitation

#### Performances

#TODO - A demander au client : y a-t-il des attentes sur la « réactivité » du programme ? (traitement lourd = temps important à attendre)

#### Capacités

Etant donné que nos composants vont implémenter des composants plus ou moins complexes (O(n²) pour KNN, O(n3) pour RNG), il serait bien vu de prévenir l’utilisateur si le nombre de données qu’il souhaite traiter est trop important (message type « L’opération demandée peut prendre très longtemps à s’exécuter, souhaitez vous continuer ? ».

# 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.