**Editeur de graphe pour calcul d’itinéraires dans une ville ou région**

Les graphes sont très utiles dans de très nombreuses applications. Ici, on veut pouvoir construire et éditer un graphe orienté avec attributs pour modéliser les rues d’une ville ou région, afin de calculer des itinéraires de plus court chemin. Le programme est conçu et réalisé en UML/Java. On adoptera une méthode de conception « agile » pour spécifier et réaliser le système par étapes successives. A chaque étape, une application fonctionnelle est produite incorporant de nouvelles fonctions. On part d’une application de départ que l’on va améliorer dans différentes directions. Le système possède un mode d’utilisation (mode Utilisation) par défaut permettant le calcul d’itinéraires entre deux localisations. Une évolution majeure envisagée est d’ajouter un mode de fonctionnement (mode Edition) permettant de créer et modifier le graphe des rues (voir ci-dessous) en juxtaposition d’une carte de la zone géographique, l’ajout/suppression interactif de sommets et d’arcs avec clavier/souris qui vont définir le graphe des rues interconnectées. Deux exemples de cartes et de graphes des rues sont donnés et devront pouvoir être chargés et visualisés à titre d’exemple.

Par exemple on cherchera à intégrer les fonctionnalités suivantes :

1. Evolution et amélioration de l’interface graphique. Ceux qui le souhaitent peuvent envisager une migration vers Java FX.
2. Lecture et sauvegarde XML du graphe des rues avec la carte géographique associée. Attention la sauvegarde doit préciser l’échelle utilisée (en mètres/pixel).
3. Visualisation du graphe des rues, possibilité de choix des couleurs et graphisme, possibilité de zoomer.
4. Visualisation interactive des attributs associés aux sommets et aux arcs (position, distance, nom de rue par exemple, sens de la rue).
5. Ajout d’un mode Edition permettant de créer et modifier un graphe des rues en juxtaposition d’une carte graphique d’une ville ou région. Edition des attributs et paramètres de façon interactive.
6. Calcul de plus court chemin à l’aide d’un algorithme de type Dijkstra.

**Mode édition**

Une barre d’outils permet de passer du mode Utilisation au mode Edition, et inversement. Une fois en mode édition, on peut alors éditer et modifier des rues et des sommets, avec leurs attributs, en ajouter et en supprimer. On peut choisir de créer une nouvelle zone géographique à représenter, donnée initialement sous forme d’une image. Il faut alors saisir le graphe des rues en juxtaposition de cette image. Il faut préciser l’échelle en mètres/pixel. Par exemple, à titre indicatif, des séquences d’actions possibles pour éditer et manipuler le graphe des rues peuvent être les suivantes :

* touche clavier "shift" maintenue enfoncée et click souris dans la zone de dessin (dans le vide) : ajout d'un sommet,
* touche clavier "shift" maintenue enfoncée et click souris sur un sommet : suppression du sommet, et donc des arcs qui le relient,
* touche clavier "shift" maintenue enfoncée et click souris sur un arc : ajout d'un sommet intermédiaire, l'arc initial est scindé en deux arcs,
* touche clavier "ctrl" maintenue enfoncée avec choix à la souris d'un premier sommet puis d'un second sommet : les deux sommets sont fusionnés, les arcs précédents reliés aux deux sommets sont reliés au nouveau sommet,
* touche clavier "alt" maintenue enfoncée avec succession de clicks souris sur des sommets existants : des arcs sont successivement créés reliant deux sommets clickés consécutivement, et constituent une rue,
* triple click souris, en pointant sur un arc ou sur un sommet : fait apparaître une fenêtre de modification des attributs de l'arc ou du sommet sélectionné pour en changer la couleur, la taille ou la forme,
* touches clavier "ctrl" et "shift" maintenues enfoncées avec choix à la souris d'un premier sommet puis d'un second sommet : le chemin le plus court est calculé et se dessine en surimpression du graphe, l’itinéraire est détaillé dans une fenêtre spécifique.

On notera que toute action clavier ou souris non autorisée ou non attendue entraîne une annulation de la commande en cours. De même, un time-out de 5 secondes est armé à chaque commande entreprise : la commande en cours est annulée si aucune action utilisateur n'est entreprise pendant ces 5 secondes. Bien entendu, un changement de mode Editon/Utilisation annule toute commande en cours.

**Rendu**

Le programme est réalisé en Java avec interface graphique en respectant le modèle MVC et est associé à une démarche d’analyse/conception UML. Le rendu du mini-projet en fin de semestre comporte un rapport d’analyse/conception présentant la modélisation UML, et l’ensemble des codes sources Java (avec l’exécutable) de l’application est déposé sur Moodle. Des scripts de compilation et exécution sont fournis permettant d’éviter l’utilisation d’un IDE.

L’évaluation du mini-projet comporte une soutenance de présentation réalisée sur machine de 10/15 mn en fin du semestre (jusqu’à la semaine des finaux au maximum).

Le projet est à faire en groupe d'étudiants (de 2 à 4). Le groupe doit être défini et enregistré dès la première séance de TP auprès de l’encadrant de TP. Le point sur l’avancement du mini-projet est réalisé à chaque séance de TP, avec le responsable du TP, qui évalue l’apport de chaque participant et sa contribution. Les dates de soutenances seront fixées à l’avance. Pour un archivage de vos documents, et permettre de les traiter, les dépôts sur Moodle doivent respecter la dénomination suivante :

nom1\_nom2\_nom3\_nom4\_prj\_AP4B\_titre\_du\_sujet.zip

**Contact**

"Jean-Charles CREPUT" [jean-charles.creput@utbm.fr](mailto:jean-charles.creput@utbm.fr)

"Fatima Zahrae EL-QORAYCHY" [fatima.el-qoraychy@utbm.fr](mailto:fatima.el-qoraychy@utbm.fr)