

Construisons le monde de demain

Cahier des charges

L'objectif du programme est de proposer une gestion financière ludique, à travers la modélisation d'une ville qui s'agrandit au fur et à mesure que des dépenses et revenus sont enregistrés.

Le programme permet également de visionner les différentes dépenses selon leur catégorie (alimentation, santé...) et d'estimer si un budget prévu pour le mois sera respecté au vue des dépenses déjà sauvegardées.

Chaque nouvelle entrée entraîne la création d'un bâtiment. Ce dernier génère des revenus qui pourront être utilisés pour fusionner plusieurs bâtiments du même type et ainsi embellir la ville. Un bâtiment qui résulte d'une fusion génère d'avantage de revenus.

L'interface représentera la ville sous forme de quadrillage ainsi que plusieurs boutons qui permettent d'accéder aux diverses fonctionnalités et fenêtres.



Bibliographie

- Oracle Help Center. <https://docs.oracle.com/en/>
- Stack overflow. <https://stackoverflow.com/>
- Cours d'Algorithmique et Programmation dans le dossier « Formation Initiale aux Métiers d'Ingénieur (FIMI)/Informatique » disponibles sur <https://moodle.insa-lyon.fr>
- Les matrices : <https://math.nist.gov/javanumerics/jama/>
- Les moindres carrés : <https://introcs.cs.princeton.edu/java/36inheritance/LeastSquares.java.html>

Carnet de route

Date	Travail fait	Remarques / Problèmes rencontrés
Jeudi 7 mars	Réfléchir au sujet, ce que fait le programme plus en détails algorithmiquement, établir quelles classes il y a et ce que chacune contient et se répartir le travail	Quelle sera la partie scientifique du programme ? Différents types de bâtiment >> faire un héritage avec Batiment comme classe mère abstraite
Jeudi 14 mars	Séparément : Début d'écriture des classes comme planifié	Utilisation de la méthode 'interpolation' pour la partie scientifique pour faire les prévisions de dépenses. Complexité : inversion d'une matrice (ce qui est généralement fait facilement sur Matlab) On se rend compte que les méthodes prévues dans une certaine classe seront plus facilement manipulées dans une autre classe (facile à changer)
Jeudi 21 mars	Séparément : Continuer à compléter les classes Beaucoup de tests sur des programmes trouvés sur Internet qui permettent d'inverser une matrice et tracer des courbes	C'est très difficile de tracer des courbes, on n'utilise pas le logiciel le plus adapté pour cela. De plus, les dimensions sont en pixels, les axes ne sont pas dans le bon sens ...
Jeudi 28 mars	Tester les classes pour voir si elles marchent correctement, concertations avec le chef du projet et éventuelles corrections des classes.	
Jeudi 4 avril	Programmer la sauvegarde des données à la fermeture du programme et leur	Pour des notions jamais apprises en cours, difficultés lors des recherches sur Internet car de

	rétablissement au relancement du programme Début de la mise en commun des différentes classes	nombreuses méthodes différentes sont proposées mais pas toujours compatibles avec ce qu'on veut faire (ex : lire et modifier un fichier texte). Notamment pour les packages qui semblent être très utilisés et très utiles en programmation, on n'a pas pu s'en servir car les explications sur internet ne sont pas assez claires.
Jeudi 11 avril	Mise en commun Faire marcher le jeu	Partie la plus compliquée du projet. Difficulté de mettre en place une bonne communication entre les classes.
Début vacances	Finalisation du code Compte-rendu	Toujours des améliorations à faire et donc des méthodes à ajouter...

Améliorations possibles du programme et bugs connus :

- On peut enregistrer des dépenses sur le même jour mais ceci n'est pas pris en compte dans les statistiques.
- L'axe y ne s'adapte pas aux valeurs des dépenses (si la dépense est trop grande elle ne rentre pas dans le graphique).
- On aimerait bien pouvoir sélectionner les bâtiments avec la souris.
- Et afficher les dépenses enregistrées.

Diagramme UML → voir images