***CONCEPTION***

*Ce document a pour but d’expliquer en détail la hiérarchie et la conception de notre projet*

**SOMMAIRE**

**SupportADessin……………………………………………………………………………………...2**

**TextViewer…………………………………………………………………………………………….2**

**Dessinable...............................................................................................................................2**

Système…………………………………………………………………………………………….**2**

**Vecteur2D……………………………………………………………………………………………..2**

**Montagne……………………………………………………………………………………………...2**

ChaineDeMontagnes……………………………………………………………………………...**2**

**Boite3D………………………………………………………………………………………………...3**

ChampPotentiels…………………………………………………………………………………..**3**

Potentiel…………………………………………………………………………………...…....**3**

Ciel…………………………………………………………………………………………………..**3**

CubedAir………………………………………………………………………………………...**4**

**Graphique des relations entre classes…………………………………………………………...5**

***Support à dessin***

Cette méthode est la base de notre simulation et permet, comme son nom l’indique, de définir le support au travers duquel chaque classe sera dessinée. Elle ne contient ainsi que les méthodes virtuelles pures des objets que l’on définira comme dessinables (voir classe Dessinable).

***TextViewer***

Cette classe permet la simulation en mode texte de notre système ; elle hérite de la superclasse SupportADessin et ne possède que des redéfinitions de la méthode virtuelle permettant d’afficher un objet en fonction de son type en méthodes.

***Dessinable***

Cette méthode a pour but de déterminer si une classe est dessinable ou non ; elle ne possède qu’une seule méthode virtuelle qui JSP

***Système***

Cette classe représente tout ce qui compose le système physique qui fait l’objet de notre simulation, c’est-à-dire un champ de potentiels, un ciel et une montagne (ou chaine de montagnes). Un système est dessinable, il hérite donc de la classe du même nom et de la méthode dessine\_sur permettant de dessiner chacun des objets dessinables qu’il possède.

***Vecteur2D***

Cette classe représente des vecteurs bidimensionnels à l’aide de deux réels représentant les deux composantes de chaque instance. Elle possède des opérateurs classiques permettant des opérations algébriques entre vecteurs ainsi que des manipulateurs d’attributs et une méthode affichant les deux composants.

***Montagne***

Cette classe représente une montagne dite ‘’simple’’ et regroupe les caractéristiques principales d’une montagne (coordonnées du centre, hauteur, étalement). Deux de ses méthodes sont virtuelles et pourront être redéfinies dans les sous-classes ; l’une permet la modification des attributs d’une instance à l’aide de valeurs entrées en arguments, l’autre calcule l’altitude de la montagne en un point donné. Chaque instance peut également être affichée à l’aide d’une méthode affichant ses différents attributs.

Comme une montagne est dessinable, la classe elle-même hérite de la superclasse Dessinable et est donc munie de la méthode dessine\_sur.

***ChaineDeMontagnes***

Cette classe représente une chaîne de montagnes composée de montagnes simples et de sous-chaînes ; elle possède donc des attributs réels spécifiant le nombre de chacune de ces composantes ainsi que deux tableaux les regroupant en fonction de leur type. Une chaîne

étant une montagne dessinable, elle hérite de Montagne et par extension de Dessinable. Les méthodes virtuelles de Montagne y sont redéfinies, l’une calculant l’altitude d’une montagne et retournant maintenant l’altitude maximale des montagnes et des sous-chaînes qui composent une instance en un point donné, l’autre affichant simplement les caractéristiques de la chaîne et son type. Etant dessinable, elle hérite également de la méthode dessine\_sur.

Les autres méthodes de la classe sont des manipulateurs d’attributs qui retournent le nombre de montagnes simples et de sous-chaînes composant la chaîne et permettent de remplacer une montagne ou une sous-chaîne des tableaux par une du même type donnée en argument.

***Boite3D***

Cette classe représente une boîte tridimensionnelle servant de modèle pour la conception des champs de potentiels et des ciels. Elle possède des attributs entiers représentant le nombre d’unités composant la boîte ainsi que le pas qui les sépare et un tableau tridimensionnel regroupant des variables dont le type sera déterminé lors de la déclaration des superclasses représentant les objets associés à chaque unité de la boîte (Potentiel ou Cube d’air).

Elle possède une méthode virtuelle pure permettant d’afficher les attributs d’une instance et des méthodes retournant la valeur de ces attributs.

***ChampsPotentiels***

Cette classe héritée de la superclasse Boîte3D est une collection 3D de potentiels et représente le champ de potentiels permettant d’évaluer la vitesse du vent au sein du système ; le tableau tridimensionnel dont elle hérite regroupe donc des Potentiels.

Une première méthode permet d’initialiser la valeur des potentiels sur l’ensemble de la boîte à l’aide la vitesse du vent au loin ; une autre calculer la valeur des laplaciens qui y sont associés après avoir calculé l’erreur *& itéré & résolu.* Deux méthodes permettent de calculer la vitesse du vent en un point donné et sa norme ; enfin, la classe possède une redéfinition de des méthodes héritées de Boite3D permettant d’afficher ses attributs et de retourner son tableau tridimensionnel de Potentiels.

***Potentiel***

Cette classe permet de représenter les potentiels d’un champ vectoriel à l’aide de deux vecteurs bidimensionnels, l’un représentant le potentiel en lui-même, l’autre son laplacien. Elle ne possède qu’une seule méthode, qui lui permet d’afficher une instance en affichant simplement les deux vecteurs.

Les potentiels vont constituer le champ de potentiels.

***Ciel***

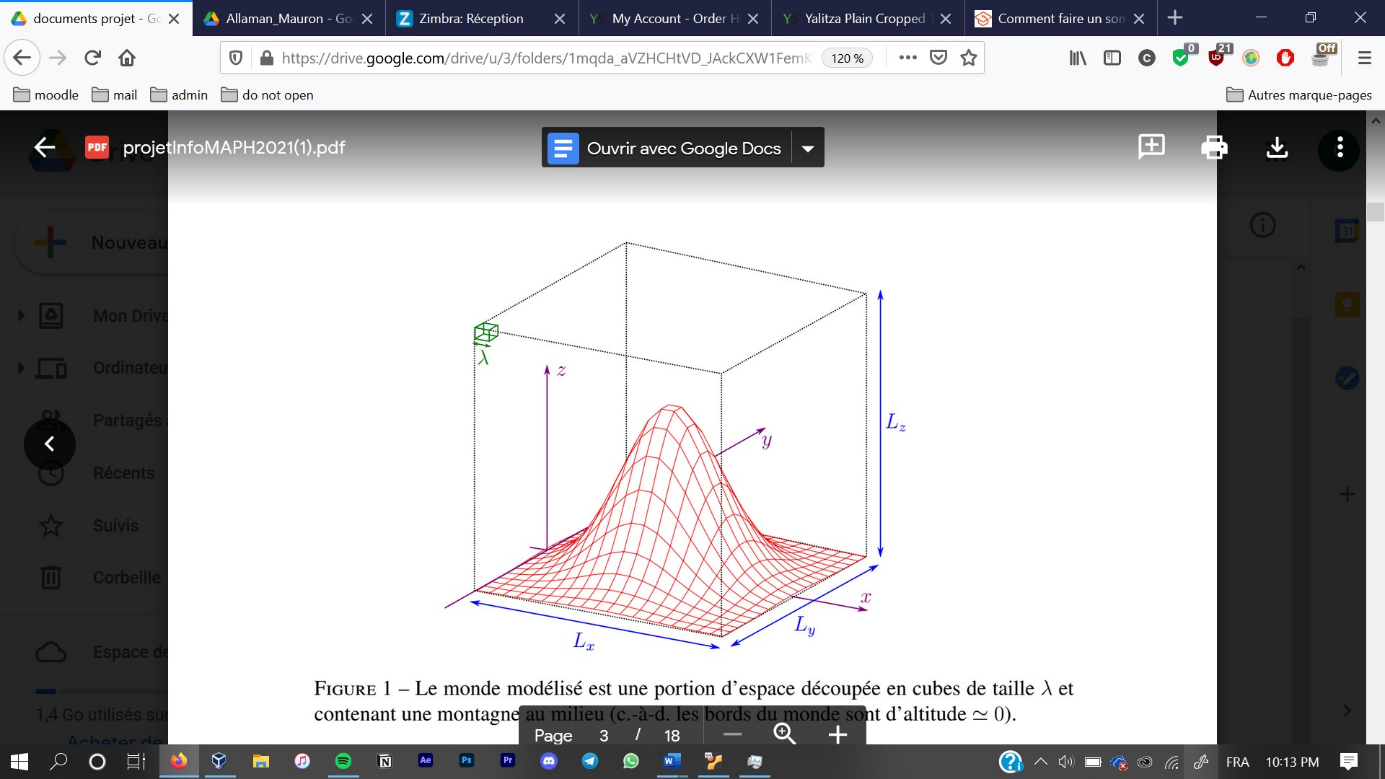
Cette classe, également héritée de Boîte 3D, est une collection 3D de cubes d’air et représente le ciel de notre système. Elle possède donc une redéfinition de la méthode virtuelle héritée de cette classe permettant d’afficher tous ses attributs ainsi qu’une méthode permettant de calculer le déplacement d’un nuage pendant un certain temps à partir d’un point donné.

Un ciel étant dessinable, la classe hérite également de la superclasse Dessinable et est donc munie d’une redéfinition de la méthode virtuelle dessine\_sur.

***CubedAir***

Cette classe représente les ‘’cubes d’air’’ qui constituent le ciel et regroupe leurs propriétés physiques, c’est-à-dire leur enthalpie, leur température, leur pression, la pression partielle de la vapeur d’eau et la pression de vapeur saturante d’eau (sous forme de réels) ainsi que leur état (nuage ou non) en booléen et leur vitesse en tableau à 3 dimensions.

Ses méthodes permettent principalement de calculer la valeur des différents attributs et d’en déduire l’état du cube, mais elle en possède également une permettant de calculer la norme de la vitesse du cube et deux permettant de l’afficher en affichant son état et l’ensemble de ses attributs.



Source : complément mathématique

Représentation d’une montagne dans une boîte3D : on y lit la longueur de chaque côté de la boîte ainsi qu’une ‘’unité’’ de longueur λ (pas)

***Graphique des relations entre classes***

***A picture containing night sky

Description automatically generated***