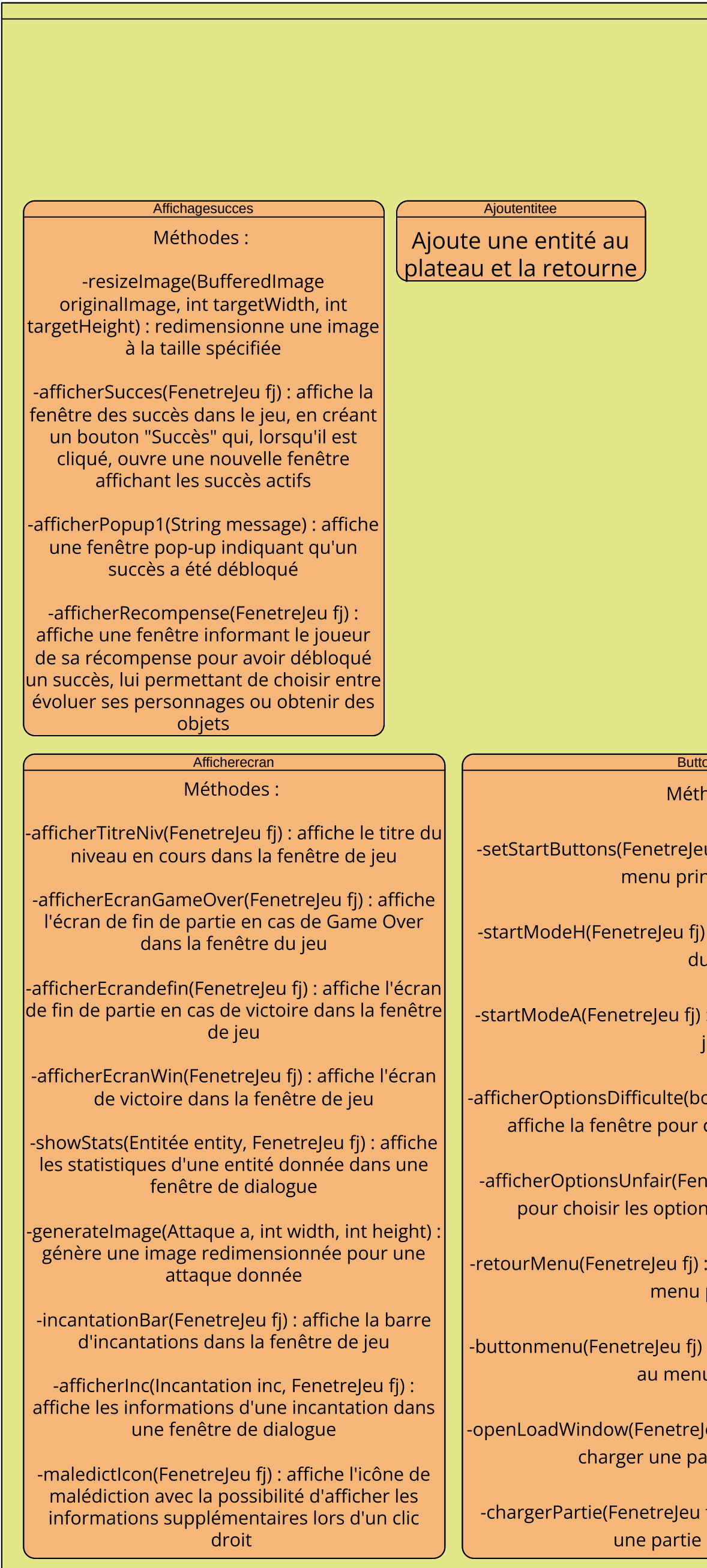
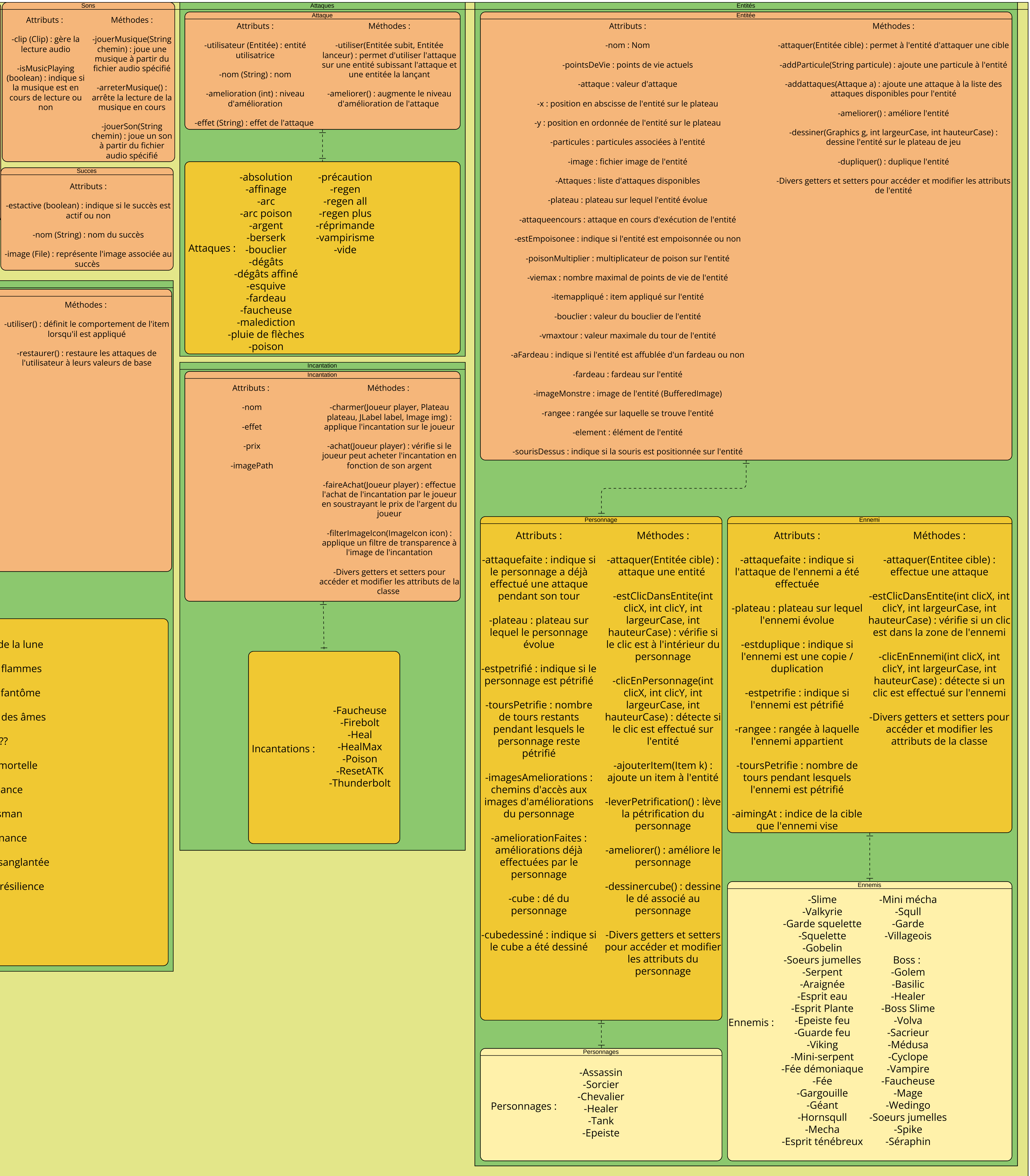
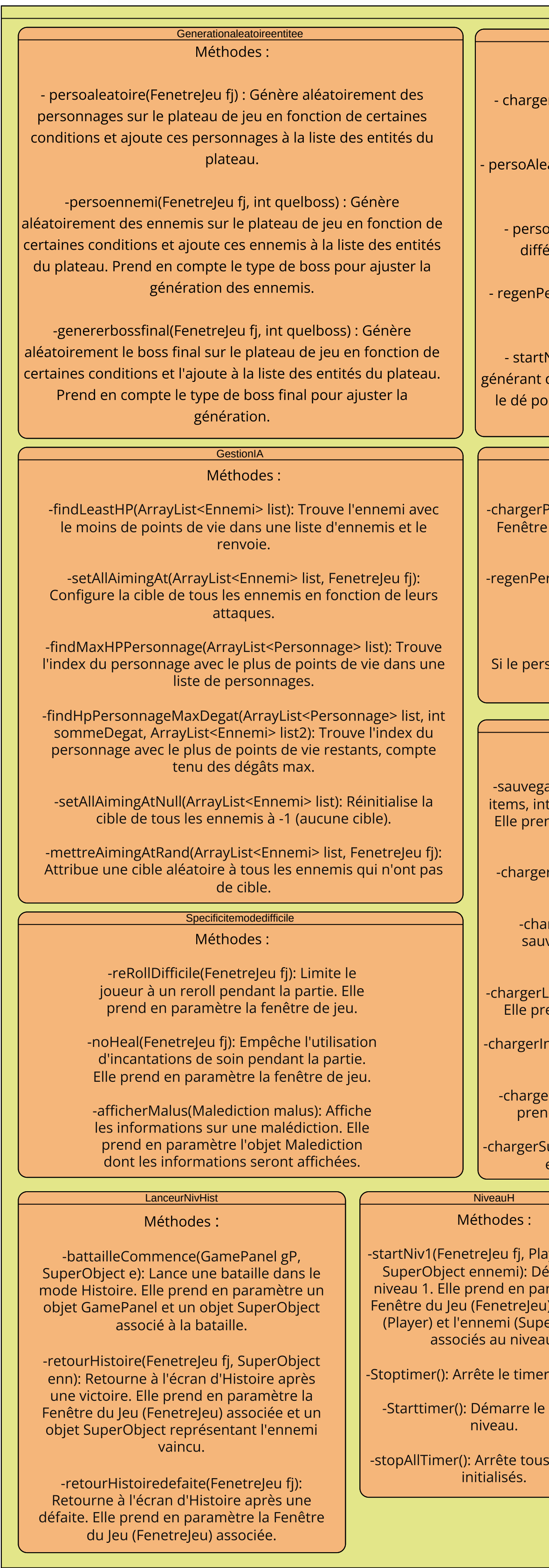
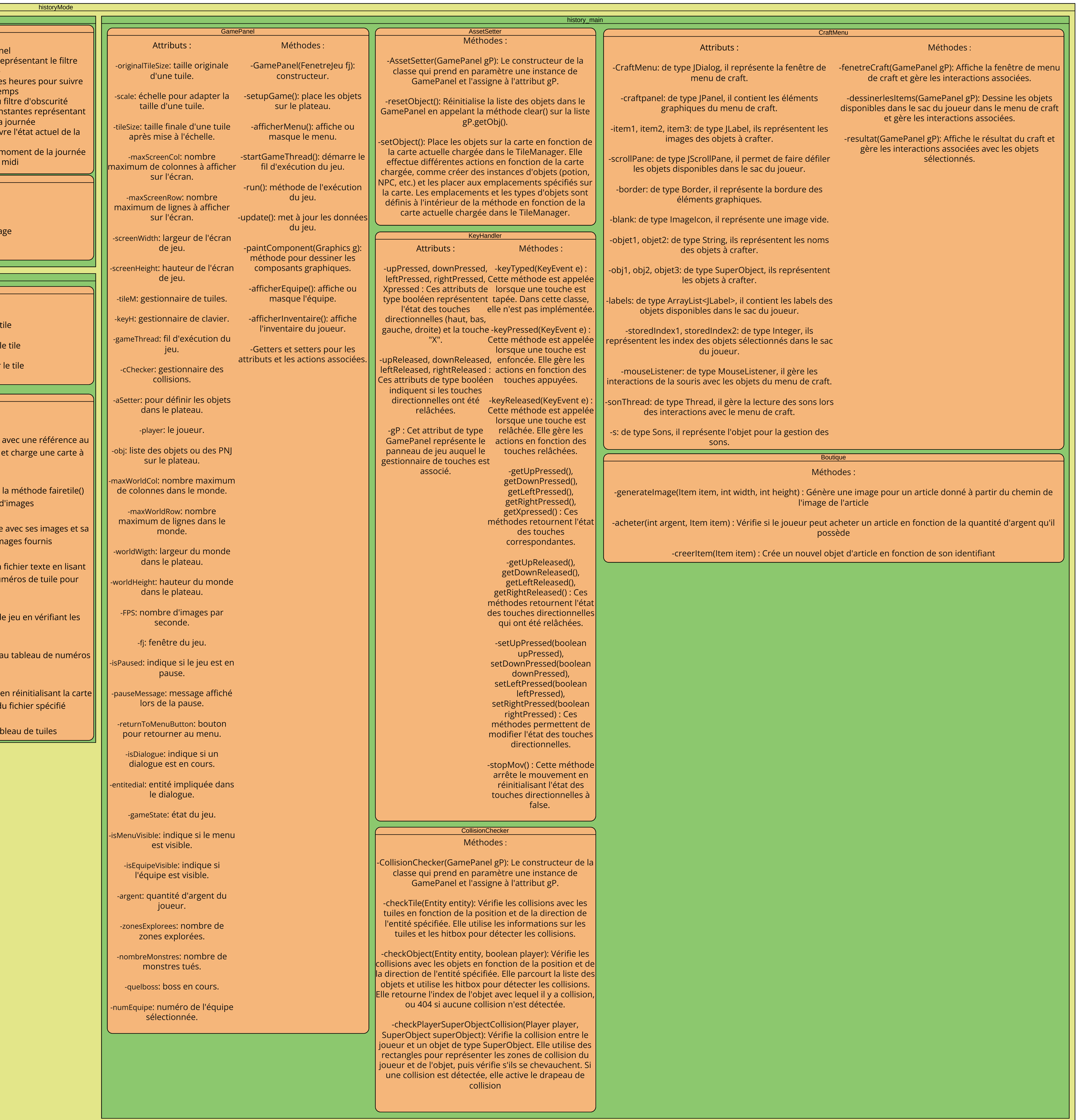


Jeu

Point de départ du programme : Lance le jeu en initialisant un objet Fenetrejeu



UML PROJ2T RC1A "SLICE & DICE"



Contrôleur	Niveau
Méthodes :	
Perso(Fenetrejeu fj) : Charge les personnages du niveau dans la fenêtre de jeu en utilisant la méthode chargePerso de la classe Plateau de la fenêtre de jeu.	
ataire(Fenetrejeu fj) : Génère aléatoirement des personnages en utilisant différentes méthodes (ajouterperso1, ajouterperso2,...) de la classe Plateau de la fenêtre de jeu.	
Ennemi(Fenetrejeu fj) : Génère aléatoirement des ennemis, y compris des boss, en utilisant différentes méthodes (ajouterennemi1, ajouterboss1,...) de la classe Plateau de la fenêtre de jeu.	
erso() : Réinitialise les personnages, enlevant les effets néfastes et en restaurant partiellement leur santé.	
Niv1(Fenetrejeu fj) : Initialise le niveau en créant une grille de jeu, affichant la fenêtre de jeu, les personnages et des ennemis, proposant des récompenses et des améliorations, et lançant sur les personnages et les ennemis. Cette méthode utilise également un Timer pour gérer les actions du jeu.	
SauvegardePerso	Méthodes :
Perso(Fenetrejeu fj) : Charge les personnes dans le plateau de jeu. Elle prend en paramètre la du jeu (Fenetrejeu) associée. Elle vide d'abord la liste des entités du plateau, puis elle charge chaque personnage depuis une liste statique Niveau1.perso.	
erso() : Régénère les personnages après une partie. Elle parcourt la liste statique Niveau1.perso et effectue les actions suivantes pour chaque personnage : -Nettoie les particules du personnage. -Réinitialise les états d'empoisonnement et les fardeaux. -Réinitialise l'état d'attaque du personnage.	
onnage à 0 points de vie, le ramène à la moitié de sa vie maximale plus un, sinon, ramène les points de vie au maximum.	
SauvegardePartie	Méthodes :
order(Plateau plateau, ArrayList<Personnage> personnages, int levelNumber, ArrayList<items> argent, ArrayList<Success> succès) : Sauvegarde la partie en écrivant les objets dans un fichier. Elle prend en paramètres le plateau de jeu, la liste des personnages, le numéro du niveau, la liste des items, la quantité d'argent et la liste des succès.	
Plateau(String fileName) : Charge le plateau à partir d'un fichier de sauvegarde. Elle prend en paramètre le nom du fichier de sauvegarde et retourne le plateau chargé.	
gerPersonnages(String fileName) : Charge la liste des personnages à partir d'un fichier de sauvegarde. Elle prend en paramètre le nom du fichier de sauvegarde et retourne la liste des personnages chargés.	
levelNumber(String fileName) : Charge le numéro du niveau à partir d'un fichier de sauvegarde. Elle prend en paramètre le nom du fichier de sauvegarde et retourne le numéro du niveau chargé.	
inventaire(String fileName) : Charge l'inventaire à partir d'un fichier de sauvegarde. Elle prend en paramètre le nom du fichier de sauvegarde et retourne la liste des items chargés.	
Argent(String fileName) : Charge la quantité d'argent à partir d'un fichier de sauvegarde. Elle prend en paramètre le nom du fichier de sauvegarde et retourne la quantité d'argent chargée.	
succes(String fileName) : Charge la liste des succès à partir d'un fichier de sauvegarde. Elle prend en paramètre le nom du fichier de sauvegarde et retourne la liste des succès chargés.	
ayer joueur, marre le amètre la le joueur (Object) .j. du niveau. timer du les timers	

Entreniveau

Méthodes :

- transitionFondu(FenetreJeu fj) : effectue une transition en fondu entre deux niveaux
- proposerItemsAuxPersonnages(Plateau p) : propose des items aux personnages du plateau
- proposerAmelioration(Plateau p) : propose une amélioration pour un des personnages présents sur le plateau
- proposerReroll(Plateau p) : propose un reroll pour les personnages sur le plateau
- resizeImageIcôn(Icône image, int width, int height) : redimensionne une icône d'image
- afficherEcranInventaire(FenetreJeu fj) : affiche l'écran d'inventaire
- afficherPersonnages(FenetreJeu fj, JPanel panel) : affiche les panneaux des personnages sur le panneau global
- afficherInventaire(FenetreJeu fj, JPanel panel) : affiche le panneau des objets de l'inventaire sur le panneau global
- ajouterItemAuPersonnage(FenetreJeu fj, Item item) : attribue un objet à un personnage après la confirmation de l'utilisateur

Gestion des icônes

Méthodes :

- getPositionEntite(Entite entite, Plateau p) : prend une entité et un plateau en paramètres et renvoie la position de l'entité sur le plateau
- afficherDialogConfirmation(Plateau p) : affiche une boîte de dialogue de confirmation et gère les actions à effectuer en fonction des choix de l'utilisateur
- trouverEntiteAuClic(int x, int y, Plateau p) : trouve une entité au clic sur le plateau
- trouverMonstreAuClic(int x, int y, Plateau p) : trouve un monstre au clic sur le plateau
- trouvePersonnageAuClic(int x, int y, Plateau p) : trouve un personnage au clic sur le plateau
- sourisSurEntite(Plateau p) : gère le survol des entités par la souris en mettant à jour le statut de survol des personnages et des ennemis
- trouvePersonnageAuClic(int x, int y, Plateau p) : trouve un personnage pour une incantation spécifique au clic sur le plateau

Cube3D

Attributs :

- angle : angle de rotation du cube
- fiff : entier initialisé à 30000
- animé : indique si l'animation est activée ou non
- randomImagePath : chemin de l'image de la face représentée du dé (attaque aléatoire)
- couleur1 : couleur principale du cube
- couleur2 : couleur secondaire du cube
- attaque : type de l'attaque
- roule : indique si le cube est en train de rouler

Méthodes :

- paintComponent(Graphics g) : dessine le composant
- connect(int i, int j, double[][] points, Graphics g) : dessine une connexion entre deux points
- matriceMultiplication(double[][] a, double[][] b) : effectue la multiplication de matrices
- drawFace(int[] face, double[][] points, Graphics g) : dessine une face du cube

designchocroitem

Attributs :

- NORMAL_COLOR : couleur normale des icônes d'items
- SELECTED_COLOR : couleur des icônes sélectionnées
- ICON_SIZE : taille des icônes
- iconLabels : icônes affichées
- selectedIcon : chemin de l'icône de l'item sélectionné
- selectedPerso : chemin de l'icône du personnage sélectionné
- selectedItems : chemins des icônes des items sélectionnés
- itemMap : carte associant les chemins des icônes des items à leurs objets correspondants
- diff : permet de différencier le contenu du panneau (1 pour les items, 2 pour les personnages)

Méthodes :

- createIconLabel(String iconName) : crée un label avec une icône et gère les événements de la souris sur celle-ci
- showDescription(String iconName) : affiche la description de l'item lors d'un clic droit sur son icône
- handleSelection(String itemName) : gère la sélection d'un item ou d'un personnage
- histoDegat, histoDegatImageAttaquant, histoDegatImageVictime, histoDegatouHeal, histoEfficacité : ArrayLists utilisées pour conserver l'historique des dégâts
- pageHistoDegatActuelle : page actuelle de l'historique des dégâts
- Gp : instance de GamePanel
- estboss : indique si le jeu est en mode boss
- inventoryPanel : affiche le contenu de l'inventaire
- persoPanel : affiche les informations sur le personnage
- histoireestlancé : indique si le mode histoire est lancé
- combatpnjgagné : indique si le joueur a gagné un combat contre un PNJ
- fontPath : chemin d'accès à la police de caractères utilisée
- font : police de caractères utilisée dans la fenêtre du jeu

FenetreJeu

Attributs :

- cadre : affiche le contenu de la fenêtre de jeu
- joueur
- plateau
- Niveau : niveau de jeu
- k : BufferedImage utilisée pour le jeu
- element : indique si les éléments sont activés ou non
- nombreennemie : suit le nombre d'ennemis tués
- Bossfinal : indique si le boss final est présent
- bossactuel : suit l'index du boss actuel
- gestionAudio : gère l'audio
- estdansarcade : indique si le jeu est en mode arcade
- estdanshistoire : indique si le jeu est en mode histoire
- histoDegat, histoDegatImageAttaquant, histoDegatImageVictime, histoDegatouHeal, histoEfficacité : ArrayLists utilisées pour conserver l'historique des dégâts
- pageHistoDegatActuelle : page actuelle de l'historique des dégâts
- Gp : instance de GamePanel
- estboss : indique si le jeu est en mode boss
- inventoryPanel : affiche le contenu de l'inventaire
- persoPanel : affiche les informations sur le personnage
- histoireestlancé : indique si le mode histoire est lancé
- combatpnjgagné : indique si le joueur a gagné un combat contre un PNJ
- fontPath : chemin d'accès à la police de caractères utilisée
- font : police de caractères utilisée dans la fenêtre du jeu

Méthodes :

- startMenu() : affiche l'écran de menu principal
- activerElement() : demande au joueur s'il souhaite activer les éléments au début du jeu
- activerElement2(boolean arcade) : demande au joueur s'il souhaite activer les éléments en fonction du mode de jeu (arcade ou histoire)
- startHistoryMode() : lance le mode histoire
- clearHistoDegat() : efface l'historique des dégâts
- bossactuel : suit l'index du boss actuel
- gestionAudio : gère l'audio
- estdansarcade : indique si le jeu est en mode arcade
- estdanshistoire : indique si le jeu est en mode histoire
- histoDegat, histoDegatImageAttaquant, histoDegatImageVictime, histoDegatouHeal, histoEfficacité : ArrayLists utilisées pour conserver l'historique des dégâts
- pageHistoDegatActuelle : page actuelle de l'historique des dégâts
- Gp : instance de GamePanel
- estboss : indique si le jeu est en mode boss
- inventoryPanel : affiche le contenu de l'inventaire
- persoPanel : affiche les informations sur le personnage
- histoireestlancé : indique si le mode histoire est lancé
- combatpnjgagné : indique si le joueur a gagné un combat contre un PNJ
- fontPath : chemin d'accès à la police de caractères utilisée
- font : police de caractères utilisée dans la fenêtre du jeu

Plateau

Attributs :

- elementactive : indique si un élément est actif
- grille : plateau de jeu sous forme d'une grille de caractères
- largeurCase, hauteurCase : dimension des cases sur le plateau
- fj : fenêtre de jeu associée
- perso, ennemis : liste des personnages et ennemis présents sur le plateau
- tour : liste des tours effectués
- itemsObjects, itemsObjects2 : objets / items présents sur le plateau
- ennemiSelectionne : ennemi sélectionné
- rangeeminMonstre, rangeeminPerso : rangée la plus basse des ennemis et des personnages
- nombretour : nombre de tours
- quelboss : indique quel boss est actuellement présent
- dashPhase : phase d'un mouvement spécial
- incantations : incantations disponibles
- phase2boss : indique si le boss est dans sa deuxième phase
- kok : image
- malediction, malusEnCours : liste de malédictions et malus actuellement appliqués
- touchePresseeUneFois, difficile : indicateurs
- nbVagues : nombre de vagues

Méthodes :

- addtour, removeTour : gère les tours
- viderlist : vide les listes des personnages et des ennemis
- Getters et Setters
- chargerPerso : charge un personnage sur le plateau
- paintComponent : dessine le plateau de jeu
- ajouterSucces : ajoute des succès en fonction des actions du joueur
- estmort : gère la mort des personnages et des ennemis