CS 319 - Object-Oriented Software Engineering

API for manager classes

Syracuse

Supervisor: Uğur Doğrusöz

Group 27

Muhammad Usman

Onur Kulaksızoğlu

Ayşe Kübra Taş

1- Age Manager class:

void checkAge(): This method checks and updates the age situation of the player if he meets the conditions and unlocks some features that are unique to this new age.

2- City Manager class:

void updateAnarchy(): If player's faction is in anarchy, it reduces the left anarchy turn time.

void updateCivics(string[]): This methods gets strings from GUI as an array and creates and updates the civics of the player and creates anarchy.

3- Materials and Happiness Manager class:

void gatherRawResources(): This method iterates over the counties of the player and adds raw resources from these counties according to the worker number in a county, also if a seasonal resource, it considers the harvest season.

void manufactureMaterial(): According to user's production choices, it creates manufactured materials from materials. And these production capacities are controlled by building in the city.

void consume(): Makes proletariats, aristoi and slaves consume food and luxury resources. If food is not enough in the city, it kills some of the population. Also modifies the happiness proletariats and aristoi, proportional to their consumption amount and type.

void paySoldiers(): Iterates over all forces of the player and pays the soldiers by reducing the money from player's treasure.

void disposeOverproduction(): This method, at the end of a turn, gets rid of all non storable material that the player has. Or if a resources is storable but storage capacity is not enough, the amount of resources is kept equal to the capacity.

4- Population Manager class:

void calculateHappiness(): This methods adds all the happiness except the happiness from consuming to the people types. Like the happiness from buildings.

void calculateNewPopulation(): This method handles the increase in population every month for the city and for all the counties of the player. Population increase is proportional to the people's happiness level.

5- Recruitment Manager class:

void recruitUnits(): This method handles the recruitments that are already ordered by the player. When the training duration ends, the unit is added to the garrison of syracuse.

void orderNewRecruitments(): During player's turn, handles the creation of new orders. Reduces the resources and man power that are required to train units.

void unlockNewUnits(): Unlocks new units when player ends turn, depending on some constraints.

6- Termination Manager class:

bool isVictory(): If the player meets the conditions to win the game, it returns true.

bool isDefeat(): If the player meets the conditions to lose the game, it returns true.

7- Foreign Manager class:

void createTradeOffers(): Depending on the age of the game and situation of the AI faction, it generates random trade offers to active trade offers list.

void createShipment(): Takes an array of trade offers and checks whether they are appropriate for creating a shipment. If appropriate, creates shipment and adds the shipment object to the active shipment list.

void moveShipments(): Handles the movement of the shipments. If they reach their destination, makes appropriate transactions.

8- Building Manager class:

void newBuilding(int x, int y, boolean alignment, Building buidling): If the conditions are met, it allows the player to build a building in the given coordinates. It makes necessary bonus updates and handles the resource and man power cost changes.

void updateConstruction(): Calls passTurn methods of buildings if there isn't anarchy.

void upgradeBuilding(int x, int y): upgrades the building in the given ciirdinates if conditions are met.

void deleteBuilding(int x, int y): Deletes the building in the given coordinates. Makes necessary bonus updates.

boolean isBuildable(**int x, int y, boolean alignment, Building buidling):** Checks if the building is constructible in the given coordinates. Called when the player is hovering the mouse on the city map.

9- Game Manager class:

void newBuilding(int x, int y, boolean alignment, int ID): Delegates the parameters to the building class.

void updateConstruction(): Delegates the parameters to the building class.

void upgradeBuilding(int x, int y): Delegate the parameters to the building class.

void deleteBuilding(int x, int y): Delegate the parameters to the building class.

boolean isBuildable(**int x, int y, boolean alignment, int ID):** Delegate the parameters to the building class.