🟥 --> Header

🟩 --> Structs

🟨 --> Class

🟦 --> Enum

Constants:

Dialogue:

MiniGame:

LevelSystem:

Stats:
int level
int maxHealth
int health
int maxMana
int mana
int accuracy
int attack
int spAttack
int defense
int spDefense
int speed
int critical

Screen:

CharacterClass:
Warrior
Healer
Magician
Assassin

Entity:
bool isAlive()

void modifyHealth(int value)
void modifyMana(int value)

virtual Ability* chooseAbility()
virtual Entity* chooseTarget(Team* targets)

void useAbilityOnTarget(Ability* ability, Entity* target)
void useAbilityOnTeam(Ability* ability, Team* targets)

void setStats(Stats addStats)
private:
Stats stat

Interactable:
void showDialogue(int curDialogue)
vector<string> dialogues

void action(Player* player)

private:
int curDialogue

UIManager:
checkState()

GameManager:
bool isSafeScreen(Screen currScreen)
void randomEnemyEncounter()

vector<Entity*> setFightOrder(Team* good, Team* bad)
void Fight(vector<Entity*> order)
void transition()

Team:
vector<Entity*> members
void addMember()

bool isTeamAlive()

private:
vector<Item*> inventory

Ability:
int level
bool isAOE

Landmark:
enum LandmarkType
struct Landmark

Ally:
virtual Ability* chooseAbility()

virtual Entity* chooseTarget(Team* targets)

void removeItem()

Enemy:
virtual Ability* chooseAbility()

virtual Entity* chooseTarget(Team* targets)

Item:
void effect(Entity* target)

private:
int cost

Landmark:
LandmarkType type
position position

LandmarkType:
type of landmark = landmark character

Player:
void move(char input, int curScreen)
void interact(int curScreen)
void addItem()

private:
struct position