Entity

Entity

+ Entity(GamePanel):

+ dropItem(Entity): void
+ setup(String): BufferedImage
+ checkDrop(): void
+ changeAlpha(Graphics2D, float): void
+ getXdistance(Entity): int
+ setAction(): void
+ getYdistance(Entity): int
+ update(): void
+ draw(Graphics2D): void
+ dyingAnimation(Graphics2D): void
+ damageReaction(): void

---

centerY: int
centerX: int

Player

- Player(GamePanel, KeyHandler):

+ GetInstance(GamePanel, KeyHandler): Pla
+ pickUpObject(int): void
+ setDefaultPositions(int, int, int): void
+ setDefaultValues(): void
+ damageMonster(int): void
+ interactNPC(int): void
+ setup(String): BufferedImage
+ restoreLife(): void
+ getPlayerImage(): void
+ contactMonster(int): void
+ draw(Graphics2D): void
+ setDefaultPositions(): void
+ update(): void

ZarusTheKnight

+ ZarusTheKnight(GamePanel):

+ draw(Graphics2D): void
+ update(): void
+ setup(String, int, int): BufferedImage
+ checkDrop(): void
+ getImage(): void
+ dyingAnimation(Graphics2D): void
+ setAction(): void
+ damageReaction(): void

---

mushu

+ mushu(GamePanel):

OBJ_key

+ OBJ_key(GamePanel):

OBJ_sword1

+ OBJ_sword1(GamePanel):

OBJ_door

+ OBJ_door(GamePanel):

Green

+ Green(GamePanel):

+ dyingAnimation(Graphics2D): void
+ setAction(): void
+ getImage(): void
+ update(): void
+ draw(Graphics2D): void
+ damageReaction(): void

OBJ_ruby

+ OBJ_ruby(GamePanel):

Ghost

+ Ghost(GamePanel):

+ setAction():void
+ getImage(): void
+ damageReaction(): void
+ checkDrop(): void

OBJ_diamond

+ OBJ_diamond(GamePanel):

---

npc_monkey

+ npc_monkey(GamePanel):

+ getImage(): void
+ checkDrop(): void
+ setAction(): void

OBJ_heart

+ OBJ_heart(GamePanel):

OBJ_kfc

+ OBJ_kfc(GamePanel):

Projectile

+ Projectile(GamePanel):

+ set(int, int, String, boolean, Entity): void
+ update(): void

OBJ_emerald

+ OBJ_emerald(GamePanel):

---

npc_chicken

+ npc_chicken(GamePanel):

+ checkDrop(): void
+ getImage(): void
+ setAction(): void

OBJ_cheese

+ OBJ_cheese(GamePanel):

OBJ_Fireball

+ OBJ_Fireball(GamePanel):

- getImage(): void
+ SetFireSpeed(): void

UI

+ UI(GamePanel):

+ drawWinScreen(): void
+ drawPauseScreen(): void
+ draw(Graphics2D): void
- drawGameOverScreen(): void
+ getXforCenterText(String): int
+ drawPlayerLife(): void
+ drawTitleScreen(): void

DBManager

+ DBManager():

+ score: int
+ keys: int

+ create_table(): void

---

score: int
empty: boolean
keys: int

GamePanel

+ GamePanel():

+ update(): void
+ run(): void
+ startGameThread(): void
+ resetGame(boolean): void
+ setupGame(): void
+ paintComponent(Graphics): void

EventHandler

+ EventHandler(GamePanel):

- healingPool(int): void
- damagePit(int): void
+ checkEvent(): void
+ hit(int, int, int, String): boolean
+ teleport(int, int, int): void

KeyHandler

+ KeyHandler(GamePanel):

+ keyPressed(KeyEvent): void
+ keyTyped(KeyEvent): void
+ keyReleased(KeyEvent): void
+ winState(int): void
+ gameOverState(int): void

CollisionChecker

+ CollisionChecker(GamePanel):

+ checkTile(Entity): void
+ checkObject(Entity, boolean): int
+ checkEntity(Entity, Entity[][]): int
+ checkPlayer(Entity): boolean

TileManager

+ TileManager(GamePanel):

+ loadMap(String, int): void
+ draw(Graphics2D): void
+ setup(int, String, boolean): void
+ getTileImage(): void

EnviormentManager

+ EnviormentManager(GamePanel):

+ draw(Graphics2D): void
+ setup(): void

AssetSetter

+ AssetSetter(GamePanel):

+ setObject(): void
+ setNPC(): void

SaveLoad

+ SaveLoad(GamePanel):

+ load(): void
+ save(): void

UtilityTool

+ UtilityTool():

+ scaleImage(BufferedImage, int, int): BufferedImag

Lightning

+ Lightning(GamePanel, int):

+ draw(Graphics2D): void

Main

+ Main():

+ main(String[]): void

DataStorage

+ DataStorage():

EventRect

+ EventRect():

Tile

+ Tile():