j1EntityManager

- entities: private p2DynArray<Entity*>
- player_tex: private SDL_Texture*
- enemy_level01_ground_tex: private SDL_Texture*
- enemy_level01_air_tex: private SDL_Texture*

- j1EntityManager()
- ~j1EntityManager()
- Awake(pugi::xml_node&): public bool
- Start(): public bool
- PreUpdate(): public bool
- Update(float): public bool
- PostUpdate(): public bool
- CleanUp(): public bool
- Load(pugi::xml_node&): public bool
- Save(pugi::xml_node&) const: public bool
- CreateEntity(ENTITY_TYPES, int, int): public Entity*

Entity

- animation: protected Animation*
- collider: protected Collider*
- flip: protected SDL_RenderFlip
- hit_points: protected int
- position: public fPoint
- is_alive: public bool
- type: public ENTITY_TYPES

- Entity(ENTITY_TYPES, int, int)
- virtual ~Entity()
- GetCollider() const: public const Collider*
- Update(float): public virtual void
- Draw(SDL_Texture*, float): public virtual void
- GetHP(): public virtual int
- OnCollision(Collider*, Collider*): public virtual void

Inherits from j1Module

j1App   [1]

ENTITY_TYPES: enum

[0...*]

Player

- idle_anim: private Animation
- walk_anim: private Animation
- original_pos: private iPoint
- velocity_x: private float
- velocity_y: private float
- going_left: private bool
- going_right: private bool
- going_up: private bool
- going_down: private bool
- gravity: private float
- time: private float
- initial_speed: private float
- jump: private bool
- jump_start: private bool
- able_to_jump: private bool
- equation_is_possible: private short
- dash: private bool
- dashTime: private float
- able_to_dash: private bool

- rect: public SDL_Rect*
- previous_position: public fPoint
- god_mode: public bool
- time_finished: public bool
- Bullet_time_init: public unsigned int
- Bullet_delay: public unsigned int
- Reset_time_bullets: public bool
- player_facing: public bool

- Player(int, int)
- Update(float dt): public void
- OnCollision(Collider*, Collider*): public void
- ~Player()

Enemy_level01_air

- original_pos: private iPoint
- anim: private Animation
- is_moving_to_player: private bool
- is_moving_to_origin: private bool
- player_is_in_range: private bool
- objective: private iPoint
- range[]: private iPoint
- velocity: private iPoint

- Enemy_level_01_air(int, int)
- Update(float): public void
- CreateRange(): public void
- PlayerIsInRange(): public bool
- GetHP(): public int
- PathMovement(const p2DynArray<iPoint>*, iPoint, float): public void
- OnCollision(Collider*, Collider*): public void
- ~Enemy_level01_air()

Enemy_level01_ground

- original_pos: private iPoint
- idle: private Animation
- walk: private Animation
- is_moving_to_player: private bool
- is_moving_to_origin: private bool
- player_is_in_range: private bool
- objective: private iPoint
- range[]: private iPoint
- velocity: private iPoint

- Enemy_level_01_ground(int, int)
- Update(float): public void
- CreateRange(): public void
- PlayerIsInRange(): public bool
- GetHP(): public int
- PathMovement(const p2DynArray<iPoint>*, iPoint, float): public void
- OnCollision(Collider*, Collider*): public void
- ~Enemy_level01_air()