ENTITY SYSTEM UML

i1EntityFactory inherits from i1Module, as all other modules. Impinfo, Playerinfo, Monkevinfo and CatPeasantInfo are all structs to save enemy data (animations, etc)

j1EntityFactory geueue[]: EntityInfo entities[]: Entity* playerData: player* player : PlayerInfo monkey: MonkeyInfo imp : ImpInfo paths: p2List<PathInfo*> CatPeasant spritesheet: p2SString MonkeyPlant spritesheet: p2SString Imp spritesheet : p2SString Player spritesheet : p2SString CatPeasantTex: SDL Texture* MonkevPlantTex: SDL Texture* ImpTex: SDL Texture* j1EntityFactoruy() virtual(~j1EntityFactory()) Awake(pugi::xml_node&) : bool Start(): bool PreUpdate(): bool Update(float) : bool PostUpdate(): bool CleanUp(): bool OnCollision(Collider, Collider): void LoadPathsInfo(): bool SaveRepetitivePaths(uint&): bool SaveStartEndPaths(uint&): bool AddEntities(): bool AddEntity(ENTITY_TYPES, uint): bool GetImpInfo(): ImpInfo& GetMonkeyInfo(): MonkeyInfo&

GetPlayerInfo(): PlayerInfo&

GetPathByIndex(uint) const : PathInfo* Save(pugi::xml_node&) const : **bool** Load(pugi::xml node&): bool

Entity collider : Collider* path_info: PathInfo* animation: Animation last pathfinding: const p2DynArray<iPoint> mlast_pathfiinding : p2DynArray<iPoint> deltaTime : float tvpe : int lives : int position: fPoint i pos : iPoint start pos: fPoint speed: fPoint last pos: fPoint collider_pos : iPoint collider size iPoint Entity(float, float, PathInfo*) virtual(~Entity()) GetCollider: const Collider* virtual (move(float) : void) virutal (UpdatePath() void) virtual (UpdatePathfinding() void) virtual (UpdateAnimations() void) virutal (Draw(SDL Texure*) void) virtual (OnCollision(Collider*, Collider*) void)

Plaver

default state: playerstates animationPlayer: Animation*

up : bool down : bool left: bool right: bool dt : float

player: PlayerInfo

Player(float x, float y, PathInfo* path) OnCollision(Collider*, Collider*): void

Move(float): void

PlayerStateMachine(): void DashForward(): void DashBackward(): void ShotRight(): void ShotLeft(): void MoveForward(): void MoveBackward(): void MoveForwardJumping(): void MoveBackwardJumping(): void

Jump(): float ApplySpeed(): void CheckIfDead(): void

CheckCollision(iPoint., iPoint, int, bool, bool,

bool, bool, playerstate) : void

CalculateCollision(iPoint, iPoint, uint, uint, uint, int, bool, bool, bool, bool, playerstates): void

CatPeasant

catPeasant : CatPeasantInfo catPeasantState : CatPeasantState

cooldown : float ▲
seconds_to_wait : int
pathfinding_index : uint
pathfinding_size : uint

normal path index : StartEndPath

CatPeasant(float, float, PathInfo*)
OnCollision(Collider*, Collider*): void

Move(float dt): void

UpdateAnimations(float) **void** GeneralStatesMachine(): **void**

UpdateDirection(): void

CreatePathfinding(iPoint) : **bool**

Pathfind(): bool

Update Movement (iPoint): void

UpdatePath(): void

ResetNormalPathVariables(): bool

RecalculatePath(): **void**FindDestination(iPoint&): **void**

Hit(): **void**DoHit(): **void**CoolDown(): **void**

Monkey

monkey: MonkeyInfo monkeyState: MonkeyStates pathfinding_index: uint pathfinding_size: uint create_pathfinding: bool

pathfinding : **bool**pathfinding_stop : **bool**pathfinding finished : **bool**

pathfinding : **bool** pathfind : **bool**

follow_pathfinding1 : Collider* follow_pathfinding2 : Collider* normal_path_index : uint last_normal_path_index : uint create_pathfinding_back : bool

going_back_home : bool

home : iPoint

Monkey(float, float, PathInfo*)

OnCollision(Collider*, Collider*): void

Move(float dt): void

UpdateAnimations(float) void GeneralStatesMachine(): void UpdateDirection(): void

CreatePathfinding(iPoint): bool

Pathfind(): bool

UpdateMovement(iPoint): void

UpdatePath(): void

ResetPathfindingVariables(): bool

RecalculatePath(): **void** FlipPath(PathInfo*): **void**

Hit(): void DoHit(): void

DoNormalPath(): **void**

UpdatePathfindingAffectArea(SDL_Rect&,

SDL Rect&): void

Imp

imp: ImpInfo impState: ImpState pathfinding_index: uint pathfinding_size: uint create_pathfinding: bool pathfinding_stop. bool pathfinding_finished: bool

pathfinding : **bool** pathfind : **bool**

follow_pathfinding1 : Collider* follow_pathfinding2 : Collider* normal_path_index : StartEndPath

create_normal_path : bool do normal path : bool

Imp(float, float, PathInfo*)

OnCollision(Collider*, Collider*): void

Move(float dt): void

UpdateAnimations(float) void GeneralStatesMachine(): void UpdateDirection(): void

CreatePathfinding(iPoint): bool

Pathfind(): bool

UpdateMovement(iPoint): void

UpdatePath(): void

ResetPathfindingVariables(): bool

RecalculatePath(): void FindDestination(iPoint&): void

UpdatePathfindingAffectArea(SDL Rect&,

SDL_Rect&): void