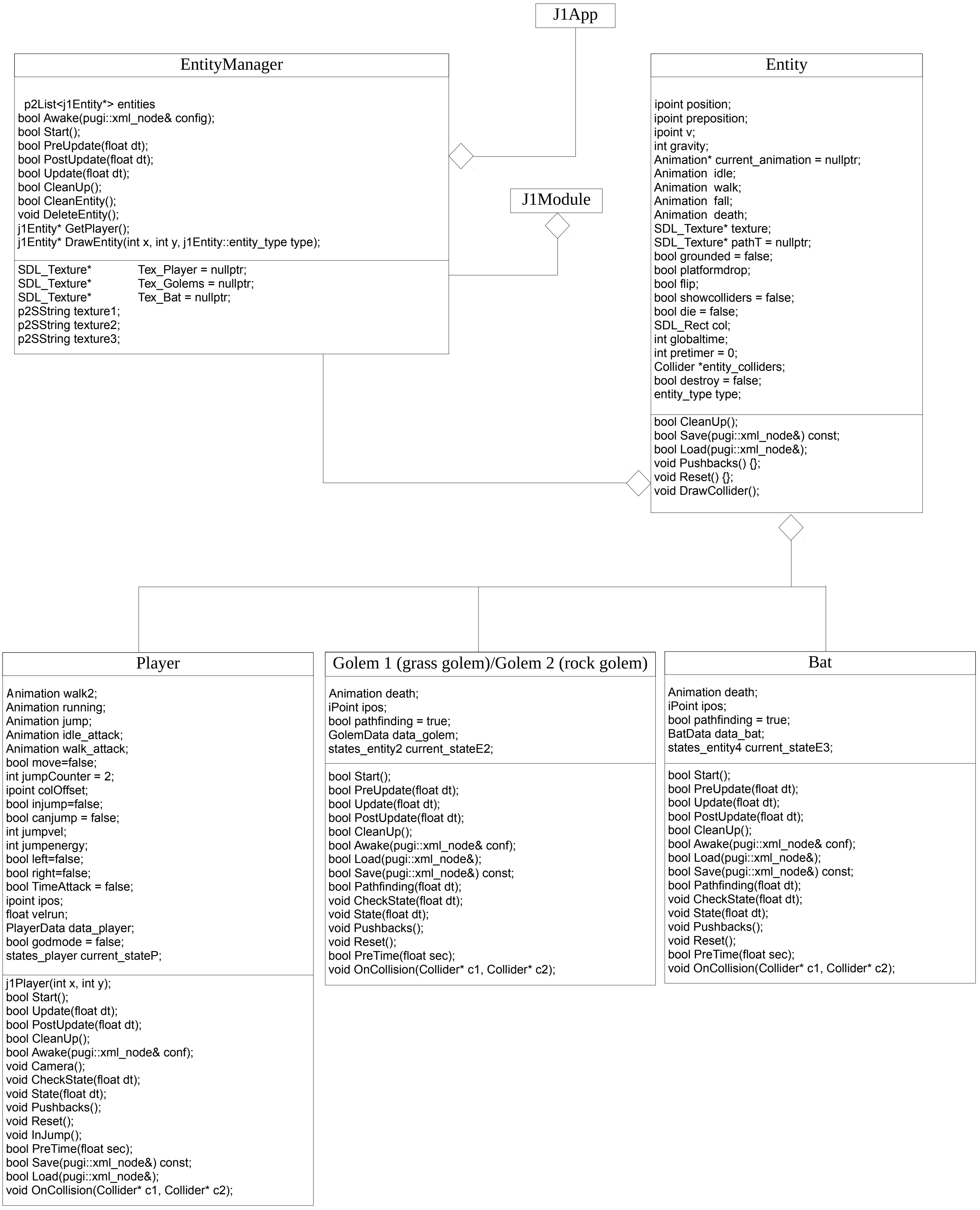

J1App

J1Module

EntityManager

```
  p2List<j1Entity*> entities
bool Awake(pugi::xml_node& config);
bool Start();
bool PreUpdate(float dt);
bool PostUpdate(float dt);
bool Update(float dt);
bool CleanUp();
bool CleanEntity();
void DeleteEntity();
j1Entity* GetPlayer();
j1Entity* DrawEntity(int x, int y, j1Entity::entity_type type);
```

```
SDL_Texture*          Tex_Player = nullptr;
SDL_Texture*          Tex_Golems = nullptr;
SDL_Texture*          Tex_Bat = nullptr;
p2SString texture1;
p2SString texture2;
p2SString texture3;
```

Entity

```
ipoint position;
ipoint preposition;
ipoint v;
int gravity;
Animation* current_animation = nullptr;
Animation  idle;
Animation  walk;
Animation  fall;
Animation  death;
SDL_Texture* texture;
SDL_Texture* pathT = nullptr;
bool grounded = false;
bool platformdrop;
bool flip;
bool showcolliders = false;
bool die = false;
SDL_Rect col;
int globaltime;
int pretimer = 0;
Collider *entity_colliders;
bool destroy = false;
entity_type type;
```

```
bool CleanUp();
bool Save(pugi::xml_node&) const;
bool Load(pugi::xml_node&);
void Pushbacks() {};
void Reset() {};
void DrawCollider();
```

Player

```
Animation walk2;
Animation running;
Animation jump;
Animation idle_attack;
Animation walk_attack;
bool move=false;
int jumpCounter = 2;
ipoint colOffset;
bool injump=false;
bool canjump = false;
int jumpvel;
int jumpenergy;
bool left=false;
bool right=false;
bool TimeAttack = false;
ipoint ipos;
float velrun;
PlayerData data_player;
bool godmode = false;
states_player current_stateP;
```

```
j1Player(int x, int y);
bool Start();
bool Update(float dt);
bool PostUpdate(float dt);
bool CleanUp();
bool Awake(pugi::xml_node& conf);
void Camera();
void CheckState(float dt);
void State(float dt);
void Pushbacks();
void Reset();
void InJump();
bool PreTime(float sec);
bool Save(pugi::xml_node&) const;
bool Load(pugi::xml_node&);
void OnCollision(Collider* c1, Collider* c2);
```

Golem 1 (grass golem)/Golem 2 (rock golem)

```
Animation death;
iPoint ipos;
bool pathfinding = true;
GolemData data_golem;
states_entity2 current_stateE2;
```

```
bool Start();
bool PreUpdate(float dt);
bool Update(float dt);
bool PostUpdate(float dt);
bool CleanUp();
bool Awake(pugi::xml_node& conf);
bool Load(pugi::xml_node&);
bool Save(pugi::xml_node&) const;
bool Pathfinding(float dt);
void CheckState(float dt);
void State(float dt);
void Pushbacks();
void Reset();
bool PreTime(float sec);
void OnCollision(Collider* c1, Collider* c2);
```

Bat

```
Animation death;
iPoint ipos;
bool pathfinding = true;
BatData data_bat;
states_entity4 current_stateE3;
```

```
bool Start();
bool PreUpdate(float dt);
bool Update(float dt);
bool PostUpdate(float dt);
bool CleanUp();
bool Awake(pugi::xml_node& conf);
bool Load(pugi::xml_node&);
bool Save(pugi::xml_node&) const;
bool Pathfinding(float dt);
void CheckState(float dt);
void State(float dt);
void Pushbacks();
void Reset();
bool PreTime(float sec);
void OnCollision(Collider* c1, Collider* c2);
```