

Gravity Duck

Game Structure Documentation

Class Relationships Documentation

1. Inheritance Relationships (Is-A Relationships)

GameObject Hierarchy

GameObject (Base class)

```
Duck
Wall
Nail
Righter
Shooter
Egg
KillerObject (Abstract Base Class)
    Brick
    Ghost
    Insect
```

Polymorphism Features:

- Virtual functions overridden in all derived classes:
 - Update()
 - Draw()
 - OnCollision()
- KillerObject has pure virtual PlayDeathSound()

2. Composition Relationships (Strong Ownership)

Container	Owned Components	Lifetime Management
Game	vector<Level*>	Destroys all levels on destruction
Level	vector<GameObject*> + vector<Wall*>	Clears all objects on destruction
Shop	Textures/Sounds	Unloads resources on destruction
Shooter	vector<Bullet>	Bullets destroyed with Shooter

3. Aggregation Relationships (Weak Ownership)

Owner	Component	Notes
Game	Shop*	Created/destroyed independently
Level	Duck*	Pointer to object in <code>objects</code> vector
Level	Egg*	Pointer to object in <code>objects</code> vector

4. Association Relationships (Uses-A)

Class	Interaction Target	Description
Game	Shop	Calls public methods (<code>EnterShop()</code>)
Duck	Righter	Special collision handling
All GameObjects	Other GameObjects	Via <code>OnCollision()</code> calls

5. Polymorphism Usage

A. Virtual Functions in GameObject

```
1 // Base class declares:
2 virtual void Update(float deltaTime) = 0;
3 virtual void Draw() = 0;
4 virtual void OnCollision(GameObject* other) = 0;
```

All derived classes implement these uniquely.

B. KillerObject Hierarchy

```
1 // Abstract method:
2 virtual void PlayDeathSound() = 0; // Implemented in Brick/Ghost/Insect
```

C. Shop State Pattern

```
1 enum ShopState { MENU, SKINS, EXCHANGE, GAMBLE };
2 ShopState currentState; // Drives polymorphic behavior
```

6. Other Notable Patterns

Singleton Pattern

```
1 // Game class implementation:
2 private:
3     static Game* instance;
4 public:
5     static Game* GetInstance() {
6         if (!instance) instance = new Game();
7         return instance;
8     }
```

Static Members

Class	Static Members	Purpose
Wall	static Texture2D wallTexture	Shared texture for all walls
Egg	static Sound winSound	Shared sound resource
Duck	static Texture2D duckTextures[4]	Skin textures

Observer-like Pattern

- Game responds to level events:

```
1 void NextLevel();      // On level completion
2 void RestartLevel();   // On failure
```