GAME DESIGN DOCUMENT

Working Title: PixelCore

Engine: Unreal Engine

1. Introduction

1.1 Scope of the Document

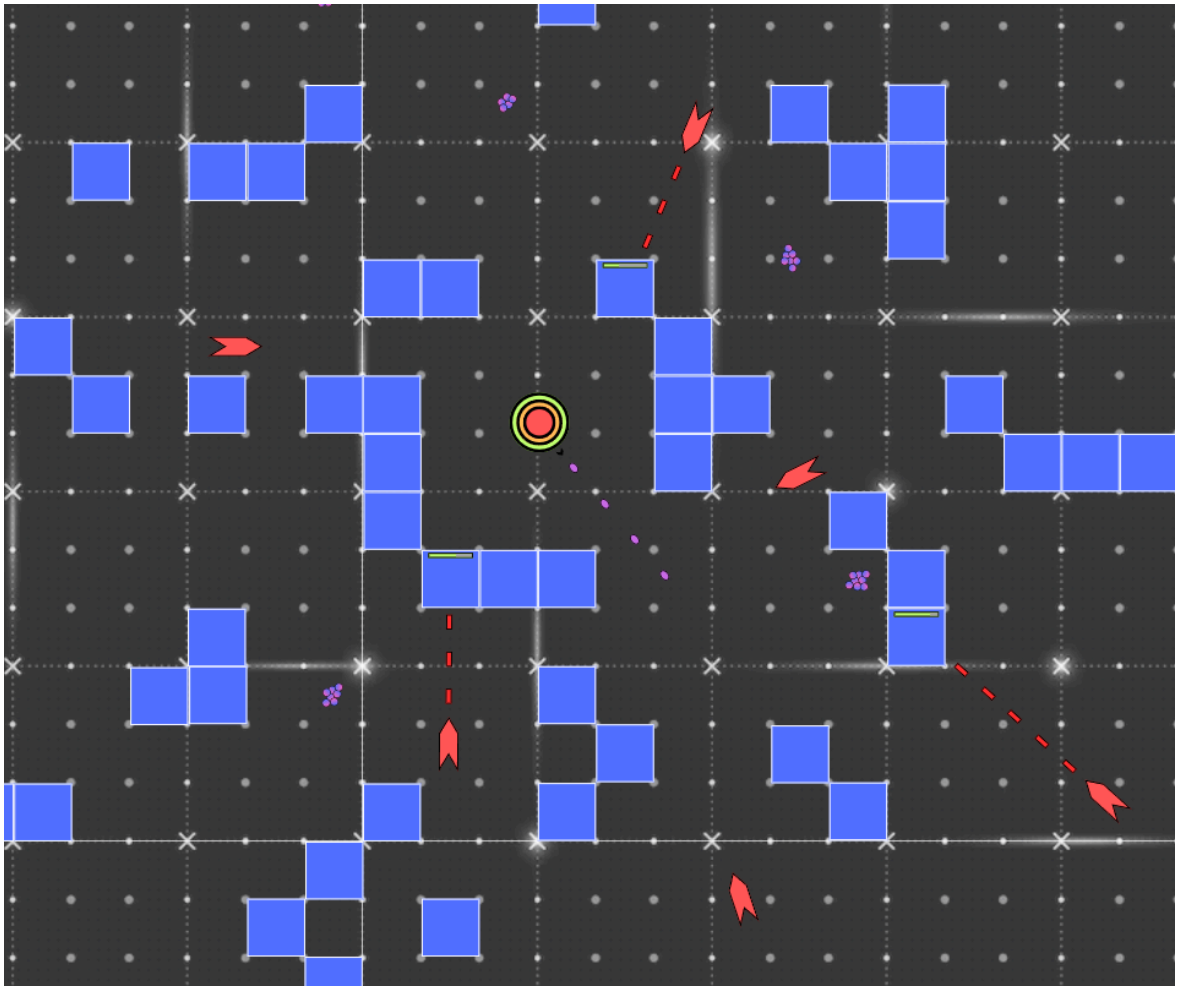
This document outlines the design, systems, and intended player experience of *PixelCore*. It is intended for:

- The developers
- Potential collaborators

It defines gameplay mechanics, progression systems, and technical direction for development in Unreal Engine.

1.2 Elevator Pitch

PixelCore is a top-down wave-based survival shooter where players choose to reposition walls in real time to defend against escalating enemy waves. Set in a procedurally generated geometric world, players must leave the safety of their base to scavenge upgrade modules that permanently enhance individual walls and turrets. The tension between defending and exploring creates a strategic, fast-paced survival experience.



2. Game Overview

2.1 Game Concept

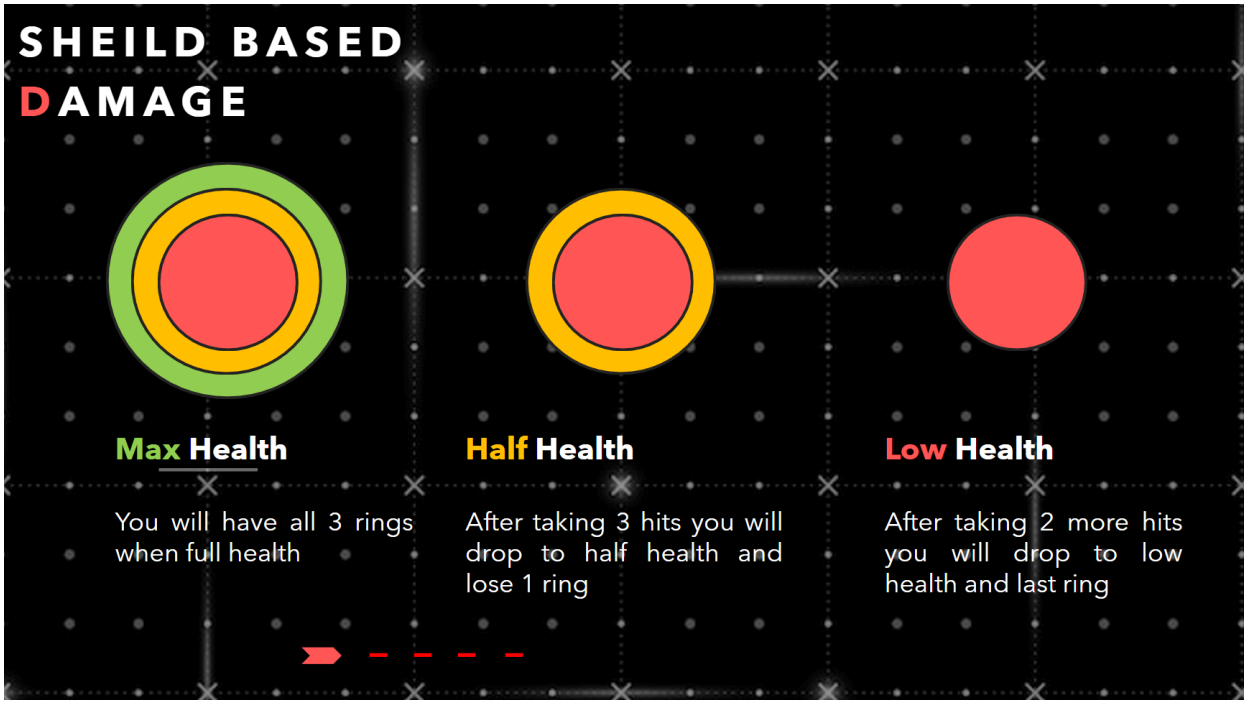
PixelCores blends:

- World survival combat
- Real-time base construction
- Risk-based scavenging progression

The player controls a circular character in a minimalist geometric world made of scattered square wall blocks. By pressing **E**, walls snap instantly to valid grid positions, allowing the player to construct defensive bases mid-combat.

Enemies (triangles) spawn in waves and attack both the player and structures. Walls have independent health and can be destroyed.

Players' health changes depending on how many times the player is hit, for example if the player is full health and gets hit 3 times the player then loses one ring, if the player is hit 2 more times the player loses one more ring. If the player is then hit one more time they die and it is game over. When the player has lost rings, the player will be more nimble and faster but a lot weaker.



The player must:

- Survive escalating waves
- Strategically construct defenses
- Venture out to retrieve upgrade modules
- Apply upgrades to specific walls and turrets

- Manage base durability

Intended Player Experience

- Tension when leaving base to explore
- Satisfaction from building optimized structures
- Tactical decision-making under pressure
- Increasing chaos and intensity over time

The core contrast is:

> Safety inside your constructed base vs vulnerability while scavenging.

2.2 Audience

Target Audience:

- Age 16–30
- Fans of wave survival shooters
- Players who enjoy building + strategic upgrades
- Indie action game players

Appeals to players who enjoy:

- Arena-based combat
- Roguelike elements
- Tactical defense building
- Replayable procedural systems

2.3 Genre

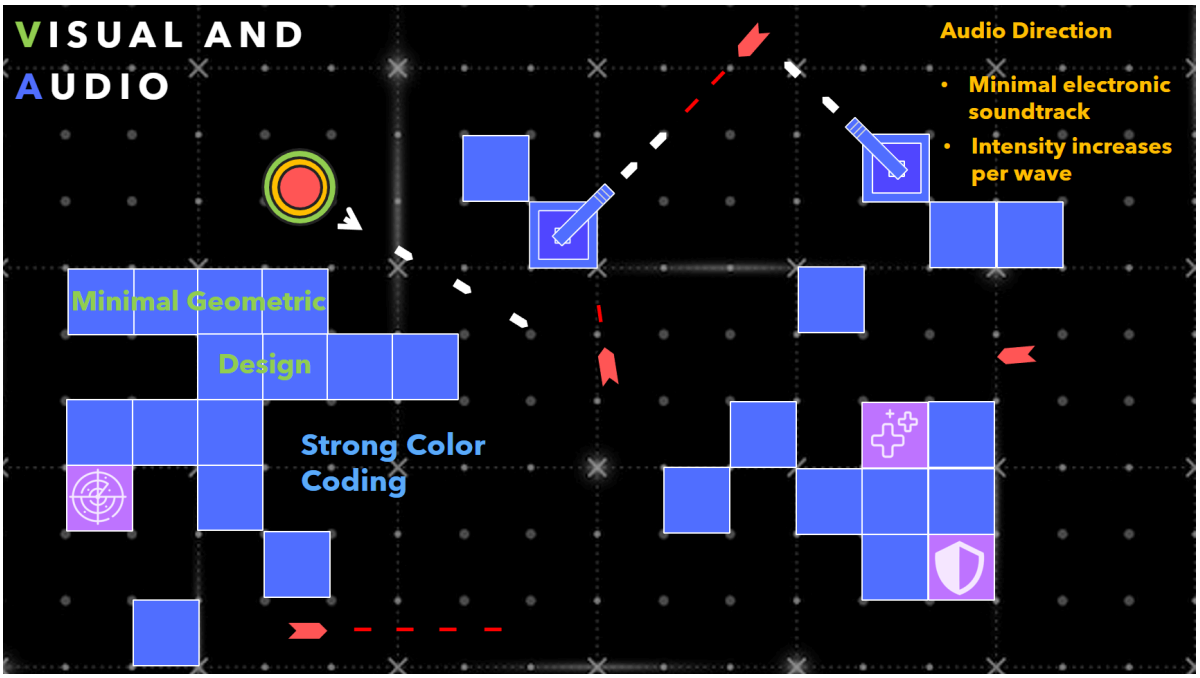
- Top-Down Shooter
- Wave Survival
- Action Strategy Hybrid
- Light Roguelike

2.4 Setting

The game takes place in an abstract geometric battlefield:

- Player = Circle
- Enemies = Triangles
- Walls = Squares
- Turrets = Modified square structures
- Upgrades = Glowing purple geometric modules

The minimal setting prioritizes gameplay clarity over narrative realism.



2.5 World Structure

The game takes place in a **procedurally generated world**.

Each run features:

- Randomized clusters of wall blocks
- Open traversal spaces between clusters
- Random upgrade spawn locations
- Enemy spawn points weighted toward world edges

This ensures replayability and strategic variation.

2.6 Player

Single-player experience.

Controls:

- WASD – Movement
- Mouse – Aim direction
- Left Click – Shoot
- Right Click – Block
- E – Move nearby wall block

The player must balance:

- Combat
- Construction
- Exploration
- Resource management

2.7 Core Loop

1. Wave begins
2. Player builds or adjusts base
3. Enemies attack
4. Walls and player take damage
5. Player eliminates enemies
6. Player ventures out to retrieve upgrades
7. Upgrades applied to specific walls/turrets
8. Next wave escalates

Repeat until death.

2.8 Look & Feel

Visual Style

- Minimalist geometric aesthetic
- Flat materials with emissive highlights
- Strong color coding:
    - Player: Multiple colors depending on health
    - Enemies: Red
    - Walls: Blue
    - Turrets: Darker Blue
    - Upgrades: Neon glow

Feel

- Responsive movement and shooting
- Instant wall snapping for smooth flow
- Rising tension per wave
- Clear visual readability during chaos

3. Gameplay

3.1 Objectives

Main Objective

- Survive as many waves as possible

Secondary Objectives

- Build efficient defensive structures
- Fully upgrade selected walls and turrets
- Achieve high wave counts

3.2 Progression

Progression occurs through three systems:

1. Wave Escalation

- Increased enemy count
- Faster projectile speeds
- Larger detection range
- Stronger enemy health

2. Currency System

Earned from:

- Enemy kills

- Surviving waves

Used for:

- Turret stat boosts
- Player stat upgrades
- Optional wall repairs

3. Scavenged Modular Upgrade System

Upgrades spawn physically in the world and must be retrieved.

The player must:

- Leave base
- Travel to high-risk zones
- Collect upgrade module
- Return to base
- Apply to specific wall or turret

Upgrades are **not global**.

3.2.1 Modular Upgrade Design

Each wall and turret has independent stats and limited upgrade slots (recommended: 1–2 slots per structure).

Wall Upgrade Examples

- **Reinforced Core** → Increased max health

- **Reflective Coating** → Chance to deflect bullets
- **Shock Plating** → Increased contact damage
- **Regenerative Matrix** → Slow passive healing

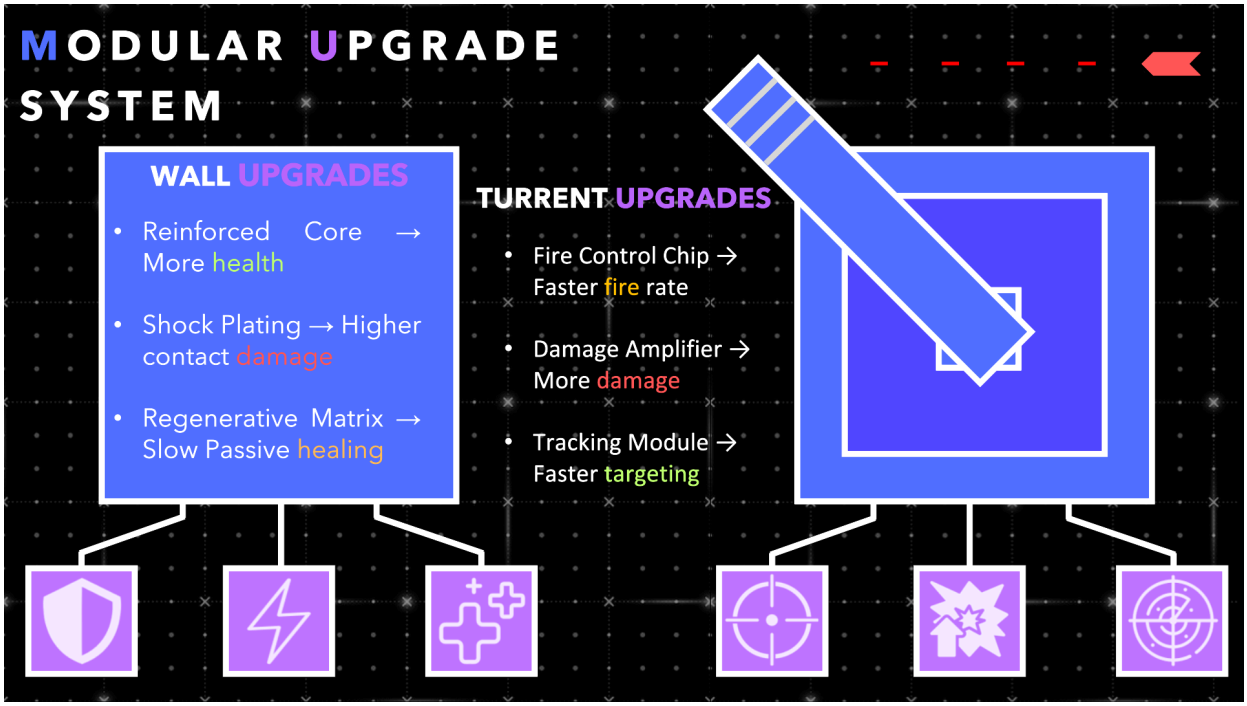
Turret Upgrade Examples

- **Fire Control Chip** → Increased fire rate
- **Damage Amplifier** → Increased damage
- **Tracking Module** → Faster targeting
- **Piercing Rounds** → Projectiles penetrate enemies

Upgrade Risk Factor

- If a wall is destroyed, attached upgrades are permanently lost.

This increases tension and strategic placement decisions.



3.3 Play Flow

Early Game:

- Small defensive structures
- Few upgrades
- Light enemy pressure

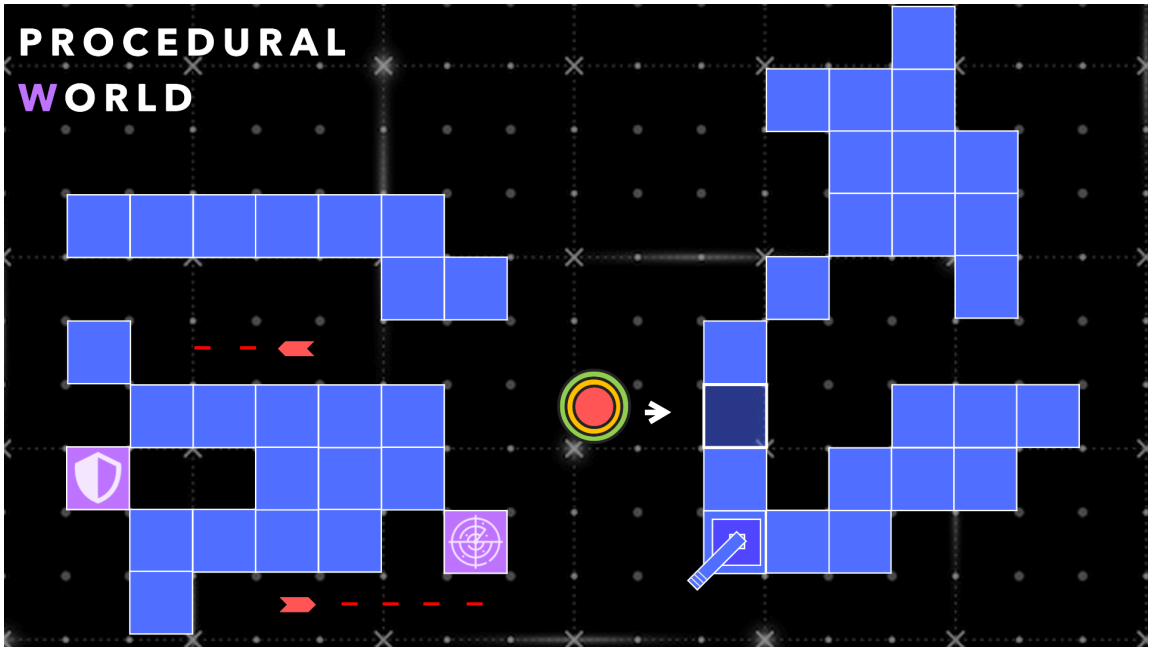
Mid Game:

- Turrets unlocked
- Strategic chokepoints formed
- Riskier scavenging required

Late Game:

- Constant wall damage

- High enemy density
- Frequent structure rebuilding
- High-pressure exploration decisions



3.4 Difficulty

Difficulty scales via:

- Enemy density
- Projectile speed
- Contact damage scaling
- Mixed enemy behaviors

Waves end when most enemies are cleared.
The next wave may begin early once only a few remain.

4. Mechanics

4.1 Rules

- Walls snap instantly to grid positions
- Walls take damage from:
    - Enemy projectiles
    - Enemy collision (damages both wall and enemy)
- Player can block incoming projectiles
- Turrets fire automatically
- Upgrades are permanently attached to specific structures
- Destroyed walls lose attached upgrades
- No time limit per wave

4.2 Game Universe

Procedural systems manage:

- Wall cluster generation
- Upgrade spawn placement
- Enemy spawn logic
- Wave scaling

4.3 Physics

- Top-down 2D gameplay in Unreal 3D space
- No gravity
- Straight-line projectile movement
- Basic collision system

4.4 Economy

Two-layer economy:

Currency

Used for player and turret improvements.

Upgrade Modules

Found physically in the world.
Applied manually to specific structures.

This creates strategic depth without overcomplication.

4.5 Character Movement

- Full 360° directional movement
- Instant directional response
- Mouse-based aiming

- Block direction tied to aim

4.6 Player Interaction

Player can interact with:

- Wall blocks (move/reposition)
- Upgrade modules
- Health pickups
- Turret placement points

4.6.1 Game Menus

- Main menu
- Pause menu
- Upgrade interface
- Game over screen

4.6.2 Saving

Prototype version:

- No mid-run save
- Run ends on player death

4.7 Assets

Player

- Circle mesh
- Projectile
- Block animation effect

Enemies

- Triangle mesh
- Enemy projectile
- Death particle effect

Environment

- Square wall blocks
- Turret structure
- Upgrade module meshes
- Health pickups

UI

- Health bar
- Wall health indicator
- Wave counter

- Currency counter
- Upgrade notification system

5. Graphics & Audio

5.1 Visual System

- 3D environment with fixed top-down camera
- Clean materials
- Emissive upgrade visuals
- Minimal lighting to preserve clarity

5.1.1 Player Camera

- Fixed top-down perspective
- Slight screen shake on impact (optional)

5.2 Interface

HUD displays:

- Player health
- Selected wall health

- Wave number
- Currency
- Upgrade indicators

Minimal UI to avoid clutter.

5.3 Audio System

- Clean electronic shooting sounds
- Impact sounds for walls and enemies
- Distortion increase during intense waves

5.3.1 Game Music

- Minimal electronic soundtrack
- Intensity scales per wave

5.3.2 Audio Feel

Audio conveys:

- Pressure
- Escalation
- Relief between waves

6. Story & Narrative

Minimal narrative focus.

Optional conceptual framing:

The player represents a digital defense system attempting to contain invading geometric corruption.

Story remains secondary to mechanics.

7. Characters

7.1 Main Character

Appearance:
White glowing circle

Abilities:

- Shoot
- Block
- Move walls
- Apply upgrades
- Deploy turrets

7.3 Enemies

Triangle entities that:

- Detect player within range
- Shoot at player and walls
- Damage structures through collision
- Scale in strength per wave

Future variants possible:

- Fast scouts
- Heavy chargers
- Sniper units

8. Game World

8.1 Look & Feel of the World

- Abstract
- Minimal
- High contrast
- Clean but tense atmosphere

8.2 Levels

Single procedural world per run.

Future expansion possibilities:

- Multiple biome variants
- Environmental hazards
- Limited wall-count modes