Game Design Document (GDD)

Operation Clean Sweep

Team:

Muhammad Affan bin Aamir – Project Lead, C++ Developer, Planner

Muhammad Umar – Map & Models Research, Blueprint Assistant, Visual Designer

Engine: Unreal Engine 5 (C++ project)

Platform: PC

Development Period: July 5–August 29, 2025

Contents

1 Overview

Operation Clean Sweep is a 3D tactical shooter built in Unreal Engine 5, featuring a hybrid third-person and first-person camera system. The game pits a squad of four (1 player + 3 AI bots, with multiplayer as a stretch goal) against a fortified enemy base with 25 initial AI bots and 15 reinforcement bots. Designed as part of a Data Structures and Algorithms (DSA) project, the game delivers fast-paced action with real-time strategy and DSA-driven logic.

2 Objective

- Infiltrate and clear an enemy HQ defended by 25 AI bots, followed by 15 reinforcement bots (7 from one side, 8 from another)
- Survive with limited health and healing resources
- Showcase DSA concepts through AI, inventory, team stats, and enemy waves

3 Target Audience

- Game developers and students learning Unreal Engine, AI, or DSA
- Players who enjoy tactical third-person shooters
- Academic reviewers evaluating practical application of DSA in game logic

4 Core Gameplay Mechanics

4.1 Player Mechanics

- Third-person view by default, with toggle to first-person
- Scope aiming switches to first-person view
- Smooth transitions between TPS and FPS
- Movement: walk, sprint, crouch, prone, jump, slide
- Health system with limited healing (20–25 HP from rare pickups)
- Gunplay with animations, muzzle flash, and bullet impact effects

4.2 Weapons

- Primary: AR (AK47 or M416, 300 rounds)
- Secondary: Pistol (Revolver or Desert Eagle, 50 rounds)
- Melee: Knife

- Ammo pickups from boxes (automatic on proximity)
- Hip fire with spread; scoped aim in FPS view

4.3 Shooting

- Hit detection with AI damage
- Recoil and visual feedback
- Impact particles and animations on hit

4.4 AI Enemy Bots

- 25 initial bots with Patrol, Alert, Attack, Chase behaviors
- 15 reinforcement bots (7 from one side, 8 from another) triggered after clearing initial enemies
- Bots can run, crouch, slide, and deal significant damage
- Each bot has health and death logic
- Behavior driven by Behavior Trees

4.5 Squad Mechanics

- 4-player squad (1 player + 3 AI bots; multiplayer optional)
- Real-time stats via Tab menu (name, kills, points, assists, alive/dead)
- Spectator mode: on player death, camera switches to a teammate

4.6 Victory and Loss Conditions

- **Win**: Eliminate all 40 enemies (25 initial + 15 reinforcements) and reach HQ
- **Loss**: Entire squad dies
- Pause/resume functionality

5 Environment & Map Design

- **Spawn Area:** Open ground with forest elements (trees, grass)
- **Boundaries:** River at the back (death trigger), unclimbable mountains on three sides
- **Objective HQ:** Fortified enemy base at the front, guarded by AI bots
- **Assets:** Customized map, characters, weapons from Unreal Marketplace/Mixamo

- **Pickups:** Ammo boxes and limited healing items (on furniture, granting 20–25 HP)

5.1 Mini-Map

- Limited circle in top corner showing nearby terrain and enemies
- Press ‘M’ to expand to full-screen map; release to revert

6 DSA Concepts Integration

DSA Concept	Integration Area
Arrays	Store squad members, enemy pools, bullet tracking
Queues	Manage spawn timers, healing cooldowns, reinforcement wave scheduling
Stacks	Handle undo actions for player inputs (e.g., reverting a move), manage recent damage events
Linked Lists	Maintain dynamic lists of active AI bots or ammunition pickups for efficient insertion/removal
Trees	Implement Behavior Trees for enemy AI logic (patrol, alert, attack)
Graphs	Utilize Navigation Mesh for AI pathfinding across the map
Hash Maps	Track kills, points, assists, and damage stats per player
State Machines	Manage player states (idle, shooting, sprinting, prone) and game states (start, playing, win/loss)

7 UI Elements

- Health & Ammo HUD
- Mission box with objective text (appears at start for 5 seconds, accessible via ‘Alt’)
- Team Tab Menu (name, kills, points, assists, alive/dead)
- Objective notifications
- Scope Zoom UI for FPS mode
- Mini-map (limited circle, expandable via ‘M’)

8 Audio and Visual Effects

- Basic audio: gunfire, impact, death sounds
- Visual effects: muzzle flash, bullet trails, impact particles
- Animations: running, crouching, sliding, shooting, death

9 Assets & Tools

- **Engine:** Unreal Engine 5 (C++ project)
- **Assets:** Unreal Marketplace, Mixamo (map, characters, weapons)
- **Animation:** Animation Blueprints for recoil, running, sliding, prone
- **UI:** UMG for health bar, mission box, stats, mini-map

10 Development Phases (Sprint Overview)

Sprint	Duration	Focus
Sprint 1	Jul 5–11	Unreal Engine setup, character movement
Sprint 2	Jul 12–18	Combat mechanics, weapon firing, FPS/TPS toggle
Sprint 3	Jul 19–25	AI bots, behavior trees, enemy shooting
Sprint 4	Jul 26–Aug 1	Map setup, mini-map, death zone
Sprint 5	Aug 2–8	UI, team stats, spectator mode
Sprint 6	Aug 9–15	Reinforcements, game states, pause/resume
Sprint 7	Aug 16–22	Objective logic, win/loss, testing
Sprint 8	Aug 23–29	Final polish, demo video, documentation

11 Team

Name	Role	Contributions
Muhammad Affan bin Aamir	Project Lead, C++ Developer, Planner	Core logic, DSA integration, planning
Muhammad Umar	Map & Models Research, Blueprint Assistant, Visual Designer	Map design, asset integration, visual polish

12 External Documentation

- **Project Proposal:** ProjectProposal.pdf
- **Notion Workspace:** Operation Clean Sweep - Project Hub
- **Cloud Storage for Final Files:** Google Drive (link to be added post-completion)

13 Stretch Goals (Optional)

- Add cave pathway to a second base
- Multiplayer support for 4-player squad
- Additional weapons or levels