

# Aspace

---

Aspace is a simple 2D top-down ship demo built with [Raylib](#) and C++.

## Introducing Collision System



Right now the game features a Separating Axis Theorem (SAT) based collision detection system that provides:

- 70% - 80% Accurate convex polygon collision detection
- AABB checks if AABB passes then only SAT is computed
- Minimum translation vectors for collision resolution
- Support for rotating hitboxes
- High performance through spatial partitioning
- Each entity can have its own collision shape defined as a convex polygon, allowing for precise interactions between game objects.
- For now the collision vertices are hardcoded to their respective ship class.

## Features

- Pixel-art ship with modular sprite-parts (thrusters, weapons, ...)
- Player-controller wrapper that can swap to bigger ships later

- Camera that follows any **CameraTarget** entity
- Component-based world grid for lightweight collision / culling
- Pure CMake build – no Makefile hacks – ships with raylib sources
- **NEW** SAT-based collision detection system for accurate hitboxes

## Prerequisites

	Windows	Linux / macOS
<b>Compiler</b>	MinGW-w64 10+	GCC 10+ / Clang 12+ (Wasn't tested on either OS so it's not guranteed that the build will be created successfully or not)
<b>CMake</b>	≥ 3.10...3.90	<code>sudo apt install cmake</code> or <code>brew install cmake</code>
<b>Ninja</b>	(optional) <code>choco install ninja</code> or <code>scoop install ninja</code>	<code>sudo apt install ninja-build</code> or <code>brew install ninja</code>

**No global raylib install is required** – the build pulls the exact tag we need. (Which is 250 MB)

## Getting the code

```
git clone --recursive https://github.com/zenzxnse/Aspace.git
cd space-something
```

Using `--recursive` is only needed if you keep raylib as a git-submodule.  
With FetchContent (default), a plain `git clone` is enough.

## Building

### Quick Build (Debug and Release)

```
# Run the build script - builds both Debug and Release versions
./build.bat
```

### Build (Debug)

```
cmake -G Ninja -B build -DCMAKE_BUILD_TYPE=Debug
cmake --build build
```

Or Hit F5 (Make sure you have CMake Tools installed as well as GCC++)

---

For MacOs or Linux you may want to use the terminal instead of Visual Studio since the compiler path is hardcoded into tasks for windows.

---

Build (Release)

```
cmake -G Ninja -B build -DCMAKE_BUILD_TYPE=Release
cmake --build build
```

Run

```
./build/bin/Aspace
```

## Project Layout

├─ CMakeLists.txt	# Build configuration (FetchContent for Raylib)
├─ include/	# Public headers (Entity, World, Animator, etc.)
├─ src/	# Game source files (main.cpp, BasicShip)
├─ rsc/	# Resources (textures, spritesheets)
└─ build/	# Out-of-source build directory

## License

Aspace is released under the MIT License. See [LICENSE](#) for details.

## Author

**Zenzxnse**