

一、五人分工

角色	主职责（必须完成）	加分任务（可选但强烈推荐）
A: 主架构 + 摄像机	摄像机、地面、天空盒、坐标系	摄像机自动环绕、烟花跟随镜头切换、按键切视角
B: 粒子逻辑	粒子系统、发射器、多类型烟花	颜色渐变、二次爆炸、拖尾、螺旋烟花、心形烟花
C: 渲染 + 光照 + 后处理	点光源、Blinn-Phong、Bloom（辉光）	多层 Bloom、高亮阈值调控、泛光动画
D: 交互 + UI	键盘控制、鼠标发射、左上角 UI 文本	模式提示、烟花类型图标、简单动画 UI
E: 音效 + 资源管理	Boom 声、发射音效、资源加载	背景音乐节奏发射、视频录制、自动剪辑功能

二、20 天总流程

阶段 1：基础框架（D1-D5）

目标：能看到地面 + 天空盒 + 摄像机能移动，鼠标/键盘可用。

A（主架构）

- 完成摄像机类（移动/旋转）
- 搭建天空盒（Cube Map）
- 搭建地面（纹理 + 法线）

B（粒子）

- 完成**最基础**粒子系统：
粒子结构体 + 位置更新 + 重力 + 生命周期

C（渲染/光照）

- 点光源框架搭建（可先随便写个白光）
- 后处理框架（FrameBuffer → Quad）

D（交互/UI）

- 完成键盘控制
- 完成鼠标射线点击 → 返回地面坐标
- 画 UI 文本（GLFW + FreeType）

E (音效)

- 引入音频库 (如 irrKlang / miniaudio)
- 播放简单 Boom 声

✓ 阶段 1 交付:

能发射最基础烟花，黑夜场景完整，不崩溃

阶段 2：核心特效 (D6-D12)

目标：烟花效果“能看”，开始变好看。

A

- 摄像机环绕动画 (按 R 切换)
- 摄像机跟随某个爆炸点 (按 F)

B

- 完善烟花类型：
 - 球形
 - 环形
 - 多层烟花
- 添加颜色渐变：
 - 初始亮 → 中段彩色 → 消失

C

- 实现真正的高亮提取 (HDR)
- 实现 Gaussian Blur (横向 + 纵向)
- 合成 Bloom (Additive blending)

D

- 左上角 UI：
 - 当前类型
 - 当前模式 (自动 / 手动)
 - 烟花数量计数

E

- 爆炸音效与大小匹配
- 为不同类型烟花配不同 Boom 声

✓ 阶段 2 交付:

烟花已经漂亮、有光、有声音、有 UI

阶段 3：展示优化（D13-D20）

目标：做出最酷炫、最能在汇报中“秒杀其他组”的效果。

A

- 支持路径动画（摄像机沿轨道移动）
- 随鼠标点击自动对焦（看向发射点）

B

- 高级特效：
 - 二次爆炸（爆炸后再爆）
 - 螺旋烟花
 - Emoji/心形烟花
 - 随机发射器（随机时间随机位置）

C

- Bloom 提升为多层强辉光
- 烟花粒子边缘加 Soft Particle（柔化消失）
- Unlimited Additive Blending（防溢出）

D

- UI 添加动画：
 - 平滑淡入淡出
 - 打字机效果文字
- “烟花秀”模式按 UI 显示状态

E

- 背景音乐节奏驱动发射（高分项）
- 自动录制 10 秒烟花秀为 .mp4
- 背景音乐淡入淡出

✓ 阶段 3 交付：
最终可用版本 + 汇报 PPT + 视频展示

三、20 天详细排期（每天任务）

第一周（D1-D7）

日	A	B	C	D	E
D1	项目初始化	粒子结构体	FBO 初始化	键盘输入	音频库集成
D2	摄像机	简单粒子更新	Quad 渲染	鼠标拾取	播放音效

日	A	B	C	D	E
D3	天空盒	烟花发射器	HDR 通道	UI 文本	多种 Boom
D4	地面 (Phong)	球形爆炸	高亮提取	UI框架	烟花声同步
D5	坐标系统	颜色控制	Gaussian Blur	UI内容	声音播放优化
D6	摄像机环绕	多类型烟花	Bloom 组合	模式切换UI	多音效资源
D7	调试	多次爆炸框架	颜色阈值	图标UI	音效触发

第二周 (D8-D14)

日	A	B	C	D	E
D8	跟随镜头	高级爆炸形状	亮度缓动	UI动画	节奏检测
D9	路径移动	螺旋烟花	多层 Bloom	淡入淡出UI	背景音乐添加
D10	自由切视角	拖尾粒子	SoftParticle	UI整体美化	分轨音效
D11	稳定	随机烟花	Bloom 优化	UI完善	混音处理
D12	Debug	Debug	Debug	Debug	Debug
D13	演示模式	演示烟花序列	后处理统一	UI最终版	声音最终版
D14	录制轨道	烟花定序外放	色调调整	演示UI	视频录制功能

第三周 (D15-D20)

用于——
优化、写文档、做展示视频、完善 PPT、演练汇报

工程依赖关系 (重点)

- **B (粒子)** 需要 **A (相机)** 的 View/Projection
- **C (渲染)** 需要 **B 的粒子数据** (位置、颜色)
- **D (UI)** 不依赖其他人, 可并行
- **E (音效)** 只依赖 B 的爆炸事件