

# striker2945 — 레트로 웹 슈팅 게임 프로젝트

레트로 아케이드 감성을 현대 웹 기술로 재현하는 2D 슈팅 게임. 1990년대 오락실의 추억을 브라우저에서 생생하게 경험해보세요.

팀장 : 황보정 - 2021658042

팀원 : 정원희 - 2021963066

팀원 : 정현수 -

깃허브 - <https://github.com/couque0319/gameProject>



# 프로젝트 개요

## 목표

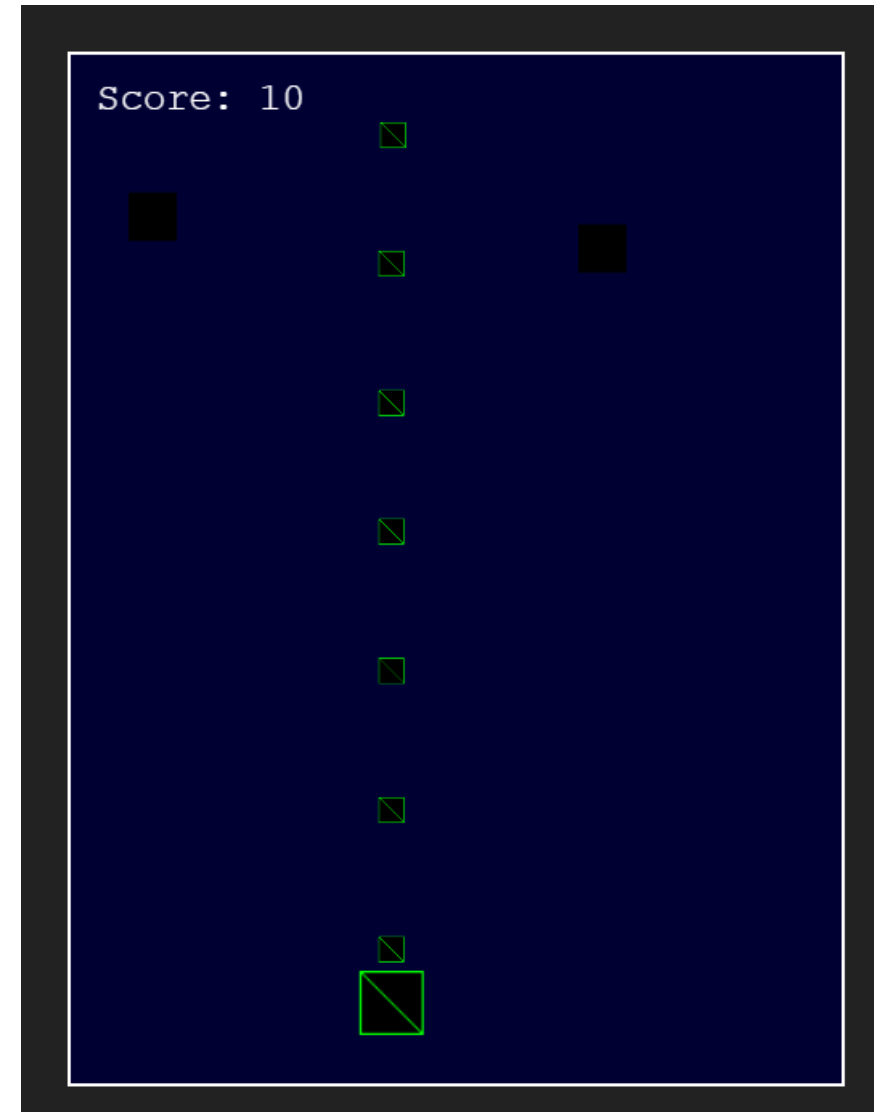
1990년대 아케이드 감성의 웹 기반 슈팅 게임을 구현하여 쉬운 접근성과 깊이 있는 플레이를 동시에 제공합니다.

## 핵심 게임플레이

- 개성 있는 기체 선택과 단계별 스테이지 진행
- 다채로운 적 웨이브와 긴장감 넘치는 보스 전투
- 초보자부터 숙련자까지, Easy/Hard 난이도 지원
- HTML5 Canvas 기반 부드러운 2D 탄막 슈팅

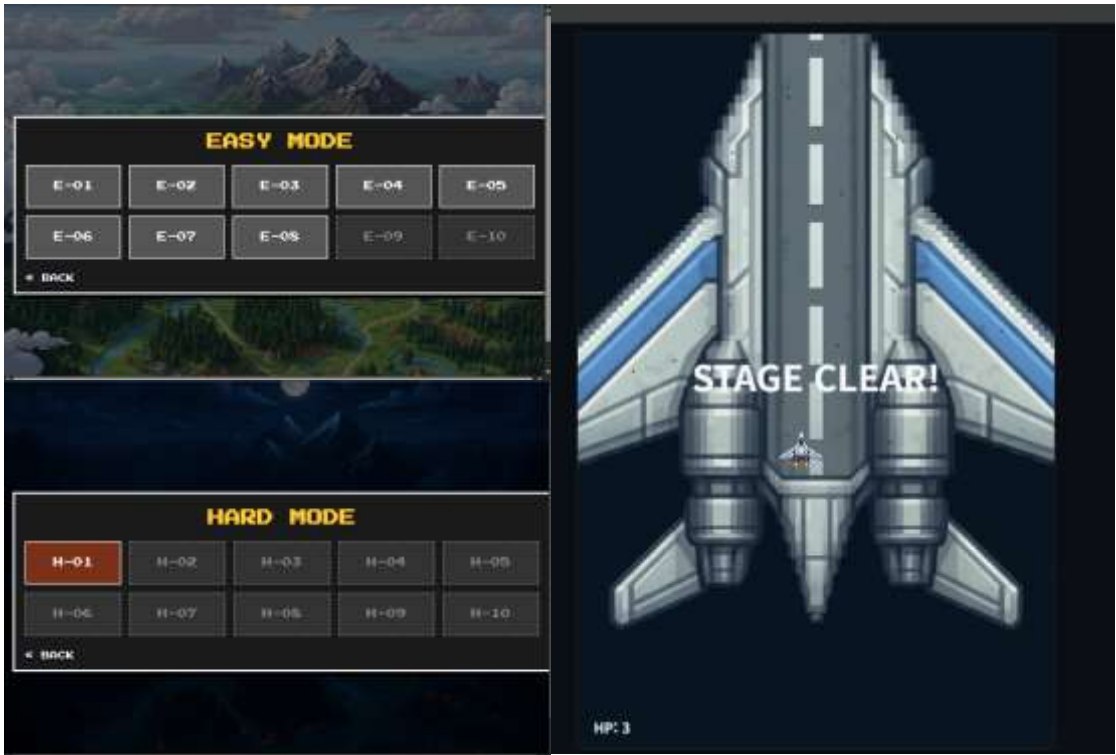
## 설계 철학

쉬운 조작, 빠른 전투 템포, 직관적인 UI로 누구나 즐길 수 있는 접근성을 추구합니다.



📝 **업데이트 목표:** 시각 효과 강화 + 사운드 완성 + 성장 요소 추가로 더욱 풍성한 게임 경험 제공

# 핵심 특징



## HTML5 Canvas 기반

경량화된 2D 렌더링으로 브라우저 호환성과 성능을 동시에 최적화했습니다.



## 데이터 기반 스테이지

stage\_defs로 웨이브·보스·패턴을 선언식으로 관리하여 확장이 용이합니다.



## 설정 저장 시스템

localStorage에 기체·조작·볼륨을 저장하여 지속적인 사용자 경험을 제공합니다.

# 주요 업데이트 요약

01

## 시각 표현

적·보스 이미지 추가 및 이펙트 강화

02

## 사운드

피격·게임오버·BGM 추가

03

## 시스템

업그레이드·상점 기능 구현

04

## 난이도

Easy/Hard 구분 지원

05

## 협업

GitHub 브랜치 관리

# 사용된 핵심 기술 스택

## Frontend 기술

- HTML5 Canvas API — 고성능 2D 렌더링
- Vanilla JavaScript — 순수 JS로 가벼운 구현
- localStorage — 클라이언트 설정 저장
- requestAnimationFrame — 부드러운 애니메이션

## 렌더링 개선

- 도형 렌더링 → 이미지(Sprite) 전환으로 시각적 퀄리티 향상
- 피격 시 블링크 효과 및 폭발 이펙트 추가
- STAGE CLEAR / GAME OVER 배너 연출 구현

❏ **확장 계획:** Phaser 3 파일(main\_game.js) 준비 완료 — 향후 고급 기능 확장 시 병행 검토 예정



# 프로젝트 구조 — 파일과 역할

1

## UI 레이어

intro.html / main.html — 스플래시, 메뉴, 설정 인터페이스

hangar.html / select\_stage.html — 기체 선택, 스테이지 선택 UI, 잠금 시스템

3

## 시스템 매니저

stage\_defs.js — 스테이지 데이터 정의

stage\_manager.js — 이벤트 매니저 및 진행 제어

## 사운드 개선

- hit.mp3, game\_over.mp3 등 상황별 사운드 구분
- 메뉴 Hover/Click 효과음으로 UI 반응성 강화

2

## 게임 코어

game.html — 메인 플레이 화면, Canvas 렌더링

game.js — 게임 로직 및 메인 루프

4

## 리소스

assets/ — 이미지, 스프라이트, 오디오 파일 관리

## 플레이어 시스템 강화

- 데미지 업그레이드 기능 추가
- 피격 시 무적 시간 및 블링크 애니메이션

# 핵심 기능 1 — 게임 루프 & 렌더링



## 초기화 단계

Canvas 설정, 리소스 로드, localStorage에서 사용자 설정 복원 처리

## 메인 루프

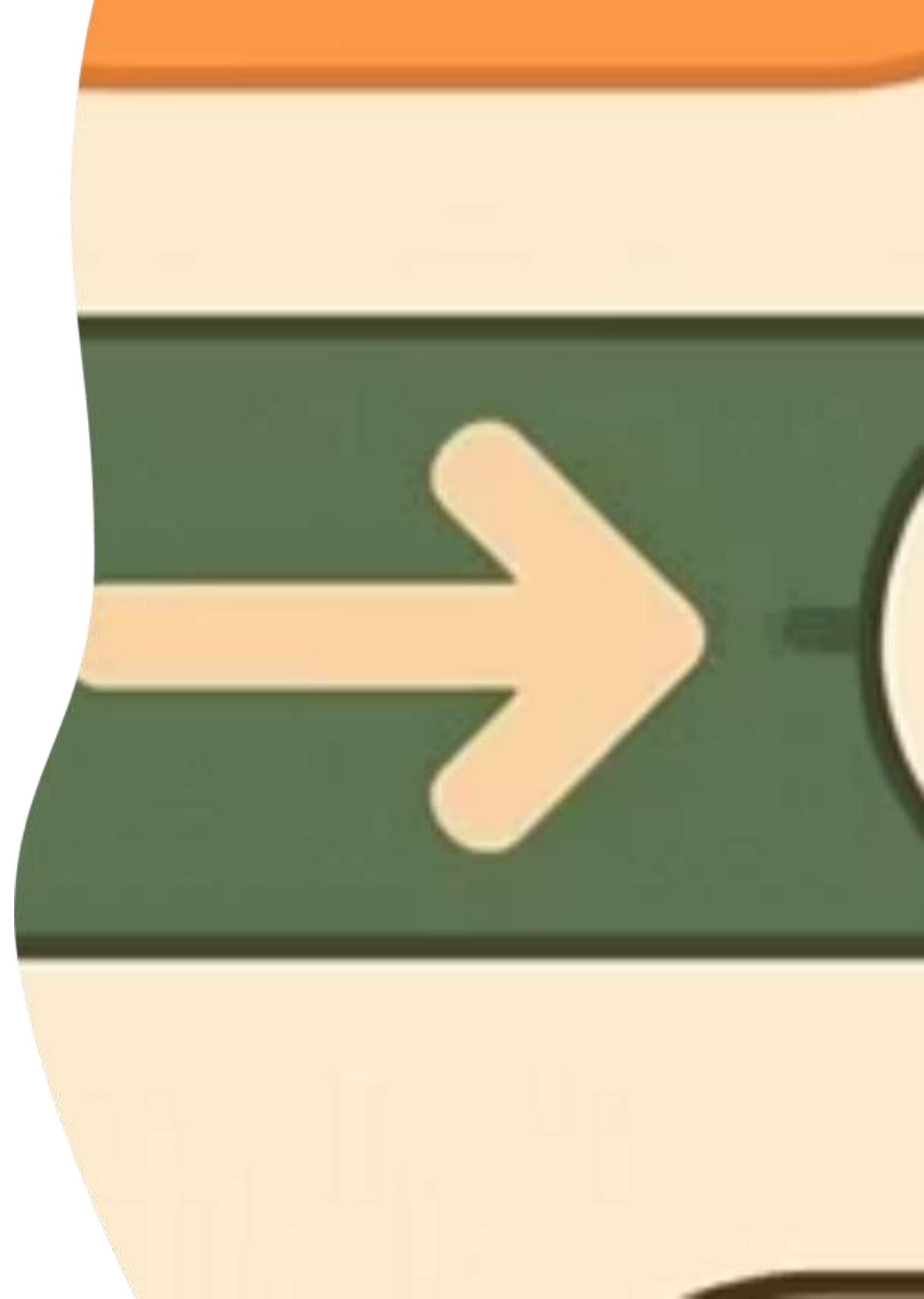
requestAnimationFrame + Fixed Timestep으로 물리 엔진 안정성 유지

## 핵심 함수

update(dt): 이동·충돌·스폰 처리draw(): 상태를 Canvas에 렌더링

## 상점 시스템

- 보스 처치 후 상점 오버레이 자동 표시
- 획득한 코인으로 다양한 업그레이드 구매 가능
- PlayerUpgrade.js와 ShopEvent.js 긴밀히 연동
- 업그레이드 항목: 공격력, 연사속도, 체력 증가 등



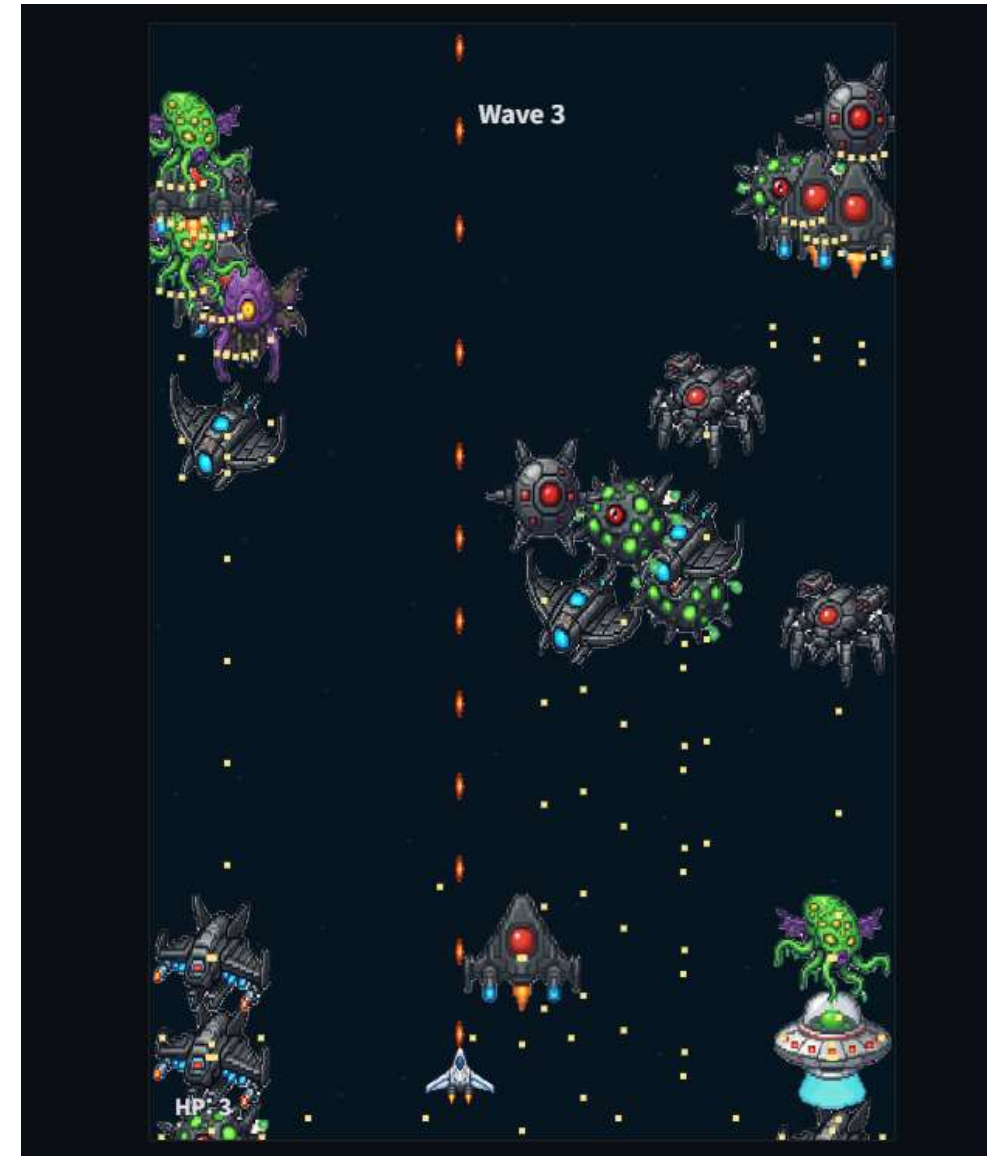


## 핵심 기능 2 — 동적 스테이지 시스템

stage\_defs.js는 시간 기반으로 웨이브를 선언합니다. 각 웨이브는 count, pattern, form을 포함하고 중간/최종 보스 타이밍을 지정합니다.

### 시스템 장점

- **확장성:** 코어 엔진을 수정하지 않고 콘텐츠만으로 새로운 스테이지 추가 가능
- **유지보수:** 선언식 데이터로 패턴 조정이 직관적
- **실시간 처리:** stage\_manager.js가 타이머를 체크해 game.js에 스폰 콜백 전송
- StageManager + StageDefs로 웨이브 자동 생성
- 보스 클리어 시 진행도 및 코인 자동 저장
- Easy / Hard 난이도 별도 밸런싱



# 핵심 기능 3 — 사용자 편의성



## 기체 선택

- Striker (A): 균형형 — 안정적인 성능
- Interceptor (B): 고속·고화력 (저내구도)

선택 정보는 localStorage의 `selectedAirplane`에 자동 저장됩니다.

## 협업 과정

- 1 GitHub 브랜치 관리  
기능별 브랜치 단위 개발로 안정적 통합

- 2 AI 프롬프트 활용  
ChatGPT를 통한 코드 개선 및 최적화

- 3 통합 테스트  
머지 후 기능 테스트 및 시각적 검수



## 설정 메뉴

- BGM·SFX 개별 볼륨 슬라이더
- 방향키 / WASD / 마우스 조작 선택
- 모든 설정은 자동 저장되어 재접속 시에도 유지



# 협업 과정

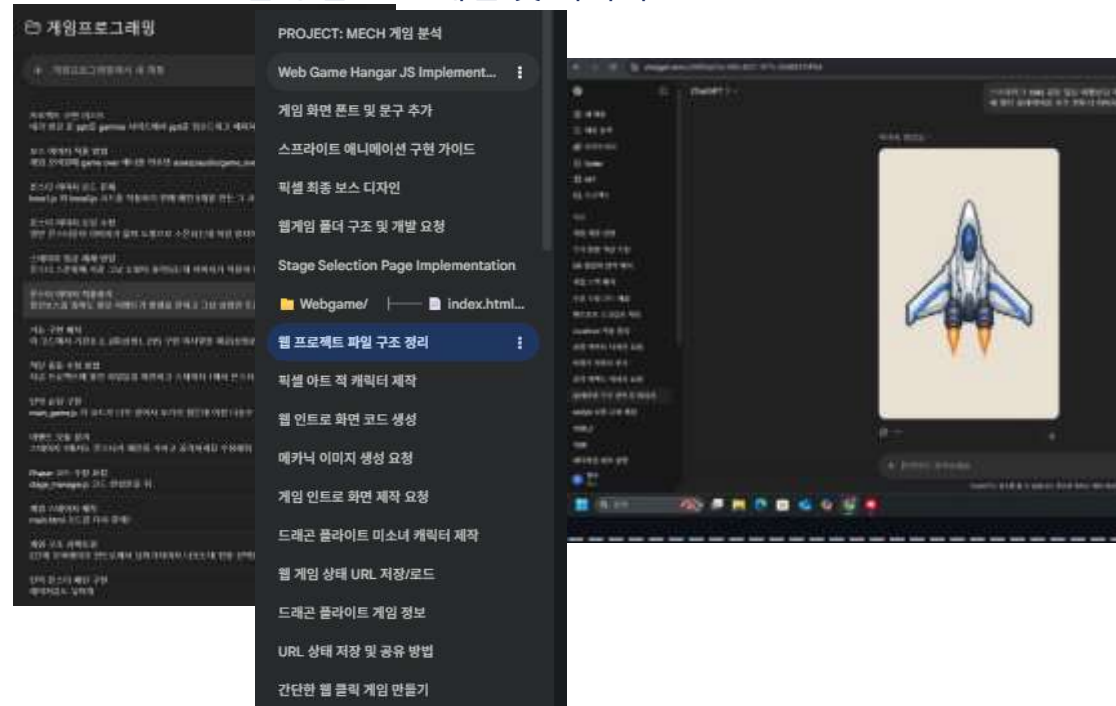
## 1 GitHub 브랜치 관리

기능별 브랜치 단위 개발로 안정적 통합



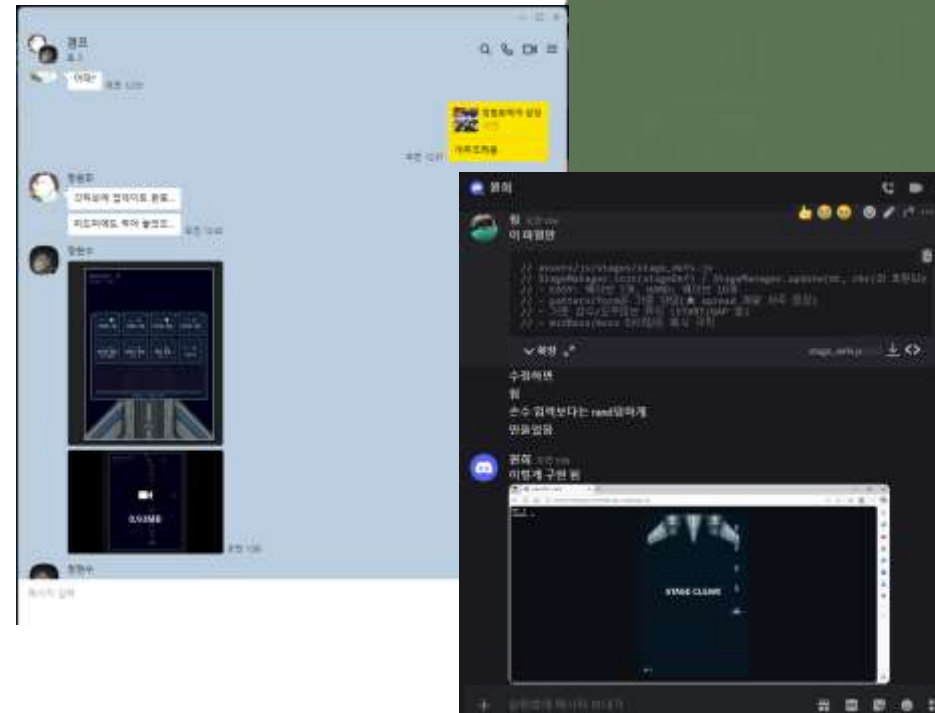
## 2 AI 프롬프트 활용

ChatGPT를 통한 코드 개선 및 최적화



## 3 협업 툴

Github를 베이스로 kakaotalk과 discord를 사용



# 향후 개선 로드맵



## 콘텐츠 확장

신규 기체·적·보스 추가, 아이템(강화·폭탄·쉴드) 도입으로 플레이 다양성 대폭 증가



## 코드 리팩토링

Phaser 3 정식 도입 검토, 모듈화 및 테스트 커버리지 강화로 유지보수성 개선



## 시각·애니메이션 강화

엔진 불꽃, 파티클 효과, 화면 셰이킹 등 피드백 강화를 통한 타격감 향상



## 출처

이미지 – gemini

게임 개발 – chat gpt, gemini

소리 파일 -<https://freesori.tistory.com/28>

협업 툴 : github, kakaotalk, discord

팀 점수 : 8점

無限

1E17