**게임 QA 계획서**

**팀 지뉴**

**문서 이력 변경**

|  |  |  |  |
| --- | --- | --- | --- |
| 날짜 | 작성자 | 버전 | 내용 |
| 20180409 | 이연정 | 0.1 | 기초 내용 작성 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**QA 계획 승인**

|  |  |  |
| --- | --- | --- |
| 승인자 | 이름 및 서명 | 승인 날짜 |
| 팀장 | 이연정 | 20180409 |
| 교수 | 이종원 |  |

목차

## 1. [개요](#개요)

1.1 범위

1.2 목적

## 2. [테스트 베이시스](#테스트베이시스)

## 3. [QA 전략](#QA전략)

3.1 리뷰 전략

3.2 기능테스트 전략

3.3 재테스팅 전략

## 4. [일정 계획](#일정계획)

## 5. [테스팅 인프라](#테스팅인프라)

5.1 테스트 환경

5.2 테스트 도구 사용

## 6. [게임QA 조직](#게임QA조직)

## 7. [QA 산출물](#QA산출물)

7.1 QA 보고문서

7.2 주간 보고

7.3 테스트 진행 보고

7.4 테스트 실행 관련 산출물

7.5 최종보고서

## 8. [형상관리](#형상관리)

## 9. [결함관리](#_9._결함관리_1)

1. **개요**

**1.1 범위**

1) QA 대상 게임 및 프로젝트 일정

* QA 대상 게임

|  |  |  |  |
| --- | --- | --- | --- |
| 게임명 | 장르 | 주요 내용 | 개발팀 |
| 미정 | 리듬 액션 | * 유니티 엔진 사용 * 프레임을 이용한 리듬 액션 * 스토리와 음악을 즐기는 리듬게임 | 링크의 전설 |
| 미정 | 퍼즐 어드벤처 | * 유니티 엔진 사용 * 시나리오를 즐기며 퍼즐을 풀어나가는 어드벤처 | 잭팟 |
| 미정 | 액션 | * 커맨드 시스템을 이용한 액션 | 트와일라잇 |

* 프로젝트 일정

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 항목 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| 링크의 전설 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 잭팟 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 트와일라잇 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

2) 테스트 항목

|  |  |  |
| --- | --- | --- |
| 구분 | 항목 | 내용 |
| 포함 | 기획서 리뷰 | 명시 내용 오류, 오타 등 리뷰 |
| 기능 테스트 | 게임 내 시스템 구성 요소 테스트 |
| 밸런스 테스트 | 게임 내 요소간의 균형 테스트 |
| 재테스팅 | 발견된 결함 수정 후 재발 테스트 |

**1.2 목적**

1. 기획서 리뷰

* 명시 내용의 오류나 문맥상의 오류, 오타, 기타 검토 사항을 체크
* 완성도와 난독률을 낮추어 개발팀의 원활한 개발 진행에 도움을 주기 위함

1. 기능 테스트

* 기획서에 의도된 기획대로 구현되었는지 확인
* 이슈, 버그 발견 피드백
* 완성도와 재미 검증

1. 재테스팅

* 기능 테스트에서 발견된 결함이 수정된 후 재발하는지 테스트
* 게임의 완성도를 높이기 위함

1. **테스트 베이시스**

|  |  |
| --- | --- |
| 테스트 베이시스 목록 | 설명 |
| 개발 기획서 | 개발 사항이 작성된 기획서 |
| 크리틱 발표 PPT | 팀 별 주차 개발진행사항 참고용 PPT |
| 크리틱 참여 회의 기록물 | 팀 별 주차 개발진행사항 피드백 기록물 |
| 팀 회의 기록물 | 팀 별 주차 개발진행사항 기록물 |
| 빌드 파일 | 개발된 결과물 테스트용 |

**3.** **QA 전략**

3.1 기획서 리뷰 전략

리뷰 진행 날짜 : 개발팀으로부터 기획서를 전달받은 후부터 일주일

리뷰 진행 인원 : 개발팀 담당 QA 인원

리뷰 진행 장소 : 개별적으로 리뷰 양식에 맞추어 작성 후 해당 주 수요일 CCRC에 모여 검토, 피드백 진행

양식

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **번호** | **검토 항목** | **참고 기획서** | **참고 페이지** | **기획서 목차 항목** | **기획서 명시 내용** | **리뷰 내용** | **비고** |

번호 : 리뷰 사항에 대한 넘버링

검토 항목 : 내용 누락, 설명 부족, 오류, 기타로 분류

* + 내용 누락 : 명시 내용에서의 내용이 빠져있는 경우
  + 설명 부족 : 서술된 명시 내용으로 충분한 이해가 어려운 경우
  + 오류 : 명시 내용, 순서가 잘못 서술 되어있을 경우
  + 기타 : 오탈자 등

참고 기획서 : 작성중인 리뷰의 기획서

참고 페이지 : 검토 항목에 포함된 내용이 작성된 기획서의 페이지 넘버링

기획서 목차 항목 : 검토 항목에 포함된 내용이 작성된 기획서의 목차 항목

기획서 명시 내용 : 검토 항목에 포함된 내용이 작성된 기획서의 명시 내용

리뷰 내용 : 검토 항목을 토대로 리뷰 작성

비고 : 기타 내용 작성

3.2 기능테스트 전략

1) 수행할 테스트

|  |  |  |
| --- | --- | --- |
| 테스트 종류 | 완료조건 | 발생 산출물 |
| 기능 테스트 | TC에 작성된 모든 항목 결과에 상관없이 전체 테스트 1회 완료 | 결함 발견 시 버그 리포트 작성 |
| 밸런스 테스트 | 기획서의 수치대로 적용되어있는지 1회~3회 빌드 테스트 | 플레이 경험에 기반하여 심각한 밸런스 저해요소 발견시 리포트로 작성 |

2) TC 양식

TC 진행 날짜 : 개발팀으로부터 기획서를 전달받은 후부터 일주일

TC 진행 인원 : 개발팀 담당 QA 인원

TC 진행 장소 : 개별적으로 TC 양식에 맞추어 작성 후 해당 주 수요일 CCRC에 모여 검토, 피드백 진행

양식

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **번호** | **카테고리** | | | | **점검 내용** | **결과** | **~~TC 업데이트~~** | **~~TC 업데이트 사유~~** | **비고** |
| **대분류** | **중분류** | **소분류** | **사전조건** |

번호 : TC 넘버링

대분류 : TC별 분류 중 중분류를 포함한 전체 분류 기준

중분류 : TC별 분류 중 소분류를 포함한 분류 기준

소분류 : TC별 분류 중 TC를 포함한 분류 기준

사전 조건 : 점검 내용을 위해 선행적으로 이루어져야 하는 조건

점검 내용 : TC 체크 내용

결과 : PASS, FAIL, N/A, BLOCK으로 결과 체크

* + PASS : 점검 내용이 정상적으로 확인 될 경우
  + FAIL : 점검 내용이 확인되지 않을 경우
  + N/A : 점검 내용이 알 수 없는 이유로 확인되지 않을 경우
  + BLOCK : 구현되지 않음, 개발 변경 등

비고 : 기타 사항 작성

3) 버그 리포트 양식

버그 리포트 진행 날짜 : 개발팀으로부터 기획서를 전달받은 후부터 일주일

버그 리포트 진행 인원 : 개발팀 담당 QA 인원

버그 리포트 진행 장소 : 개별적으로 버그 리포트 양식에 맞추어 작성 후 해당 주 수요일 CCRC에 모여 검토, 피드백 진행

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **번호** | **TC 항목 번호** | **버그 분류** | **버그 발생 조건** | **버그 발생 내용** | **비고** |

번호 : 버그 넘버링

TC 항목 번호 : TC에 작성된 항목 중 결함이 발생된 넘버링

버그 분류 : 그래픽, 클라이언트, 시스템, 사운드로 분류

버그 발생 조건 : 결함을 재현할 수 있는 재현 절차 작성

버그 발생 내용 : 발생 결함의 내용

비고 : 기타 사항 작성

3.3 재테스팅 전략

1) 수행할 테스트

|  |  |
| --- | --- |
| 테스트 종류 | 완료조건 |
| 재테스팅 | 기능 테스트에서 발견된 결함이 PASS가 될 때까지 반복 테스트 |

2) 재테스팅 양식

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **~~번호~~** | **~~카테고리~~** | | | | **~~점검 내용~~** | **~~결과~~** | **TC 업데이트** | **TC 업데이트 사유** | **~~비고~~** |
| **~~대분류~~** | **~~중분류~~** | **~~소분류~~** | **~~사전조건~~** |

TC 업데이트

* + 갱신 : 재테스팅 시 PASS 결과가 확인 될 경우
  + 수정 : 재테스팅 시 FAIL, N/A, BLOCK 결과가 확인 될 경우

TC 업데이트 사유 : TC 업데이트의 이유 작성

**4.** **일정 계획**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 내용 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| QA대상연구 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| QA계획작성 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 문서리뷰 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| TC작성 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| TC리뷰&수정 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 테스트실행 및 결함보고 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 개발팀 수정 및 테스트 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 테스트 최종보고 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 테스트 마감 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

**5.** **테스팅 인프라**

**5.1 테스트 환경**

프로세서 : Intel(R) Core(TM) i5-7500 CPU @ 3.40GHz 3.41 GHz

RAM : 16.0GB

시스템 : 64-bit operating system, x64-based processor

에디션 :　Windows 10 Pro

버전 : 1709

OS 빌드 : 16299.309

**5.2 테스트 도구 사용**

|  |  |
| --- | --- |
| 사용 도구 | 사용처 |
| Excel 2016 | 테스트 케이스 작성, 리뷰 리포트, 버그 리포트 |
| Word 2016 | 기획서 작성, 밸런스 테스트 리포트 작성 |
| github | 문서 형상관리 |

**6.** **게임QA 조직**

링크의 전설

|  |  |  |  |
| --- | --- | --- | --- |
| 이름 | 역할 | 담당 게임 | 비고 |
| 이연정 | 팀장, 테스터, 문서작성 | 트와일라잇,링크 |  |
| 김유진 | 테스터, 문서작성 | 잭팟, 트와일라잇 |  |
| 김민영 | 테스터, 문서작성 | 트와일라잇 |  |
| 장한별 | 테스터, 문서작성 | 링크, 잭팟 |  |

**7.** **QA 산출물**

**7.1 QA 보고문서**

QA 계획서, 테스트 케이스, 리뷰 리포트, 버그 리포트

**7.2 주간 보고**

개발팀의 크리틱 발표(2주)때마다 정기적인 보고 + 개발 팀별 정기 회의시 추가 기록

**7.3 테스트 진행 보고**

현 제작상황과 보고 주기 기준 약 1주 후의 제작상황까지를 보고하고 기록으로 남김

**7.4 테스트 실행 관련 산출물**

1) 테스트 케이스

기획서 별 테스트 케이스를 작성하여 테스트를 진행

2) 테스트 프로시저 및 시나리오

추후 작성

**7.5 최종보고서**

개발팀 별 테스트 케이스, 기획서 리뷰 리포트, 버그 리포트, 테스트 케이스 리포트 그래프

**8.** **형상관리**

https://github.com/naaaamm/team-jinyu

해당 주소를 저장소로 사용하여 문서의 형상관리를 진행

개발팀에서 전달받은 문서도 함께 보관

# 9. 결함관리

# 결함관리 일시 : 테스트를 진행하여 일주일 간 버그 리포트 작성

결함관리 진행 인원 : 개발팀 담당 QA 인원

# 결함관리 진행 장소 : 담당 QA 인원이 버그 리포트 작성 후 테스트 진행 주 수요일 CCRC에서 진행