|  |
| --- |
| 주니어 게임 개발자를 위한 게임 엔진 & 게임 엔진 언어 + 라이브러리  테스트 웹 개발 제안  가반 404 ERROR 20233075 김민호 |

|  |
| --- |
| **2. 요약**  - 목표: 주니어 게임 개발자들이 빠르게 게임 개발 실력을 향상시키고 게임 엔진에 대한 이해도를 높이기 위해 구상함  - 핵심 내용 : 원래 게임 엔진 사용법을 익히고 어떤 기능이 있는지 파악하기 위해 여러 군데 퍼져 있는 공식문서나 유튜브 영상들을 일일이 다 찾아보면서 배웠음. 영상 강의나 책을 통해 배운다고 해도 자신이 이 기능을 정확히 이해하고 활용할 수 있는지 가늠하기 어려움. 이러한 문제를 해결하기 위해 게임엔진 테스트 웹을 개발하려고 함.  - 중요성 (e.g. 기대되는 효과) : 주니어 게임 개발자들이 게임 엔진을 활용한 문제에 도전해보면서 자신에게 부족한 것이 어떤 것인지를 알아갈 수 있고, 문제를 풀며 전보다 더 빠르게 엔진과 엔진에서 제공하는 라이브러리에 대한 이해도를 향상시킬 수 있음. 그로 인해 더욱 다양하고 고도화된 게임을 제작할 수 있을 거라 기대함. |
| **3. 대표 그림**  **- 그림 1. 게임 엔진 테스트 웹 인터페이스**  텍스트, 소프트웨어, 멀티미디어 소프트웨어, 웹사이트이(가) 표시된 사진  자동 생성된 설명 |

|  |
| --- |
| **4. 서론**  **- 배경 설명 :** 본인이 1학기 때 python game을 통해 프로젝트를 만들어 봤지만, 어떤 기능이 있는지 조차 일단 알기 힘들고, 아는 게 적다보니 유튜브 영상에서 활용된 기능들을 일부 수정하고 재조합하는 것이 한계였음. 이때 가장 힘들었던 것은 여기저기 퍼져있는 게임 엔진 자체에서 제공하는 기능들과 사용법들을 찾는 게 일단 굉장히 힘들었고, 사람에 따라 전체적인 큰 틀이 아예 다르다보니 정보도 통일되어 있지 않아 더 좋은 예시가 있으면, 중간에 다 지우고 처음부터 하는 경우가 많았음. 하지만 이중에서 가장 힘들다고 느낀 것은 엔진이 제공하는 기능에 대한 활용과 훈련이 안되어있다보니 내가 원하는 것을 구현하는 데에 많은 시간이 들었고, 이 또한 어떤 기능을 사용해야 하는지 감이 오지 않아 찾아보고 지우고 다시 작성하는 것을 반복했던 것이 기억에 남음.  **- 사례 분석 :** 사례를 찾아본 결과 강의 사이트, 커뮤니티, codeingame이라는 웹페이지의 사례로 구분할 수 있었음.  **강의사이트 :** 일단 이 강의 사이트는 보통 책처럼 전반적인 기능을 가르쳐주고 직접 따라해보면서 배워가는 형식이었음. 하지만 특정 기능에 대해서 설명이나 활용하는 것은 강의에서 따라 만들어보는 게임이 무엇이냐에 따라 한계가 있고, 팁을 알아갈 수 있다는 장점은 있지만, 이것도 사람마다 다름. 그리고 학습시간이 굉장히 길어지는 단점과 개인이 활용하고 훈련하는 것과는 거리가 있음.  **커뮤니티 :** 사실상 대부분 게임 개발은 이쪽에서 배우는 경우가 많음. 하지만 아까 말했듯이 정보가 크게 분산되어 있고, 내가 원하는 질문과 답이 없는 경우도 꽤나 많기 때문에 시간이 오래 걸리는 것도 문제임. 그리고 역시나 이것도 활용과는 거리가 있음.  **CodeinGame :** 이 사이트에 대해 설명을 하자면 게임 엔진을 배우는 것보다는 어떤 게임의 컨셉에 맞추어 프로그래밍 언어를 “재미있게” 배운다는 취지가 강함. 하지만 이 역시 게임 엔진과는 관련이 없었지만, 구체적은 구현 방식에 대해 생각할 때 꽤나 도움이 되었던 사례였음.  **- 문제 정의 :** 결국에 여럿 사이트를 돌아다녀본 결과 게임 엔진 기능에 대한 정보는 분산되어 있어 찾는데만 시간이 오래 걸렸고, 무엇보다 기 기능을 자신이 직접 활용하고 훈련할 수 있는 환경이 존재하지 않았음.  **- 극복 방안 :** 게임 엔진을 배우고 습득하는 데에 있어 중요한 것은 프로그래밍 언어를 배우는 것처럼 우선 전반적인 기능에 대해 나와있는 기본서와 심화서, 그리고 문제풀이를 통한 활용 능력이라고 생각함. 하지만 이 중 어떠한 것도 게임 엔진에 관련해서는 잘 정리되어 있는 곳 적고, 특히 훈련할 수 있는 환경이 없었기 때문에, 게임엔진을 활용하고 훈련할 수 있도록 게임 엔진 테스트 웹을 만들면 좋을 것 같음. |

|  |
| --- |
| **5. 본론**  - 시스템 개요 그림:  **텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진  자동 생성된 설명**  **- 필요한 기술 요소:**  **웹 페이지 :** 기본적으로 웹 페이지를 통해 구현해야 하기 때문에 html, css등과 같은 웹 페이지 구현에 필요한 언어와 페이지의 동적인 작동을 위한 javascript가 필요하다. 그리고 또한 사용자의 정보를 저장하기 위한 데이터베이스와 유저들에게 여럿 기능을 제공하고 받기위한 서버도 필요하다고 생각된다. 여기서 데이터 베이스는 MYSQL등으로, 서버는 cenOS를 설치하여, 아파치 서버, MariaDB, php를 통해 구현하면 될 듯 하다.  **테스트 :** 아무래도 테스트 방법은 두 가지 일 듯 하다. 우선 웹에서 게임 엔진과 연동하여 띄워주어 직접 코드를 작성한 후 돌려서 목표 행동을 달성하도록 테스트하거나, 아니면 엔진을 켜서 목표 행동을 달성했는지에 대한 정보를 웹사이트로 보내서 확인하는 방법등이 있을 듯 하다.  **구현 방법 및 개발 방향:**  일단 사용자들이 접근하기 쉽도록 웹을 통해 구현하고 각 개인에게 각자 자신이 어떤 문제를 풀었는지, 어떤 부분을 하지 않았는지 등에 대해 알려주기 위해 로그인을 구현하고, 문제풀이 인터페이스는 “백준”처럼 여러 문제를 제공할 예정이다. 인터페이스는 들어가면 예시 영상과 코드 편집창을 만들고 실행해서 목표 행동에 도달하면 웹 페이지에 정보를 보내어 통과하는 식으로 구현할 생각이다. 그 이후 답을 제공하면서 팁도 같이 알려주는 창이 나오도록 띄울 것이다. |

|  |
| --- |
| **6. 결론**  게임 엔진을 주니어 개발자들이 빠르게 익힐 수 있도록 훈련할 수 있는 웹페이지를 구현하여 더욱 정교하고 개발자 스스로도 만족하는 게임을 만들도록 할 생각이고 이를 구현하기 위해서 앞으로 기본적인 웹 언어와, 위에서 언급했던 서버와 관련된 지식을 쌓고 공부할 예정이다. |

**7. 출처 :**

[CodeinGame]( <https://www.codingame.com>)

[Bakjoon](<https://www.acmicpc.net>)

[Stoveindie](<https://indie.onstove.com>)