기말과제 보고서

15015 김동규

**1. 프로그램명**

Mercy Fall

**2. 프로그램 소개**

메르시가 하늘에서 떨어진다. 떨어지면서 좌우로 움직여 장애물을 피하고 아이템을 먹어 점수를 얻는 게임이다. 최대한 오랫동안 버티면서 점수를 얻는 게임이다. 아래 그림은 실제 게임의 플레이 화면이다. 화면 구성에서 더 자세히 설명하겠다.



Figure 게임 플레이 화면

**3. 플레이 하는 방법**

화면의 중앙을 기준으로 왼쪽을 터치하면 메르시가 왼쪽으로 이동하고, 오른쪽을 터치하면 오른쪽으로 이동한다.

화면을 열심히 터치해서 메르시를 좌우로 움직여 장애물을 피하고 아이템을 먹으면 된다.

**4. 화면 구성 및 프로그램 동작 설명**

아래는 메인 화면이다. 메인화면은 화면 상단의 Title Bar, 중앙의 메르시 비트맵 이미지와 Image Button, 그 밑에 최고 시간과 점수를 보여주는 Text View, 마지막으로 두 개의 Image Button으로 구성된다.

Start 버튼을 누르면 앞서 보여준 실제 게임 플레이 화면이 나오며 게임을 할 수 있다.

Exit 버튼을 누르면 Main Activity, 즉 애플리케이션을 종료한다.



Figure main 화면

화면 중앙의 메르시 이미지 옆에 있는 두 개의 버튼을 누르면 게임 플레이에서 보일 메르시의 스킨을 변경할 수 있다. 버튼을 통해 이전 스킨과 다음 스킨으로 이동하면서 볼 수 있다.

Figure ,4 메르시 스킨 변경

Title bar에는 두 가지 메뉴가 있다. Option Menu를 사용해 두 가지 기능을 만들었다.

먼저, 오른쪽 상단의 메뉴를 누르면 background1부터 background8까지 항목이 나온다. 이 항목들을 터치하면 그 항목에 해당하는 이미지로 메인화면의 배경이 변하게 된다.

그렇게 중요하거나 의미있는 기능은 아니지만, Option Menu에서 터치했을 때 여러 항목이 뜨도록 하는 것을 구현해 보았다.

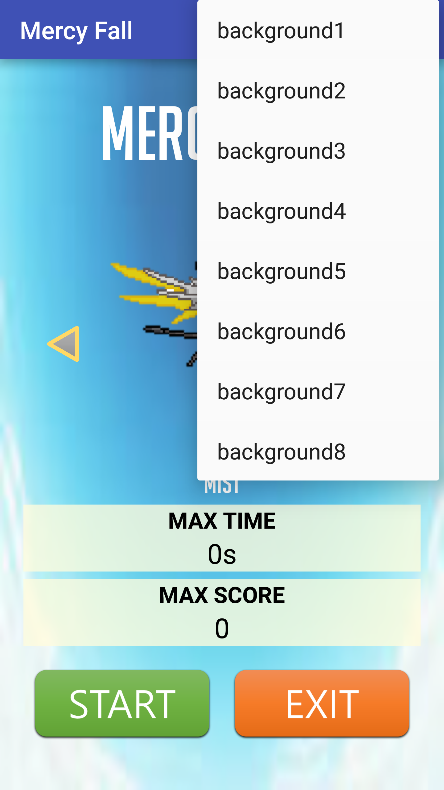
  

Figure 5,6,7 기존 배경 이미지(왼쪽)에서 Option Menu를 통해(가운데)

새로운 배경 이미지를 적용한 화면(오른쪽)

그런데 아쉽게도 이 기능은 여러 기종의 스마트폰에서 실행시켜 본 결과, android 6.0에서만 제대로 실행되는 것 같다. 그 밑의 버전에서는 점수와 시간도 제대로 뜨지 않는다는 문제점을 발견했다. 이 문제는 해결할 방법을 찾지 못했다.

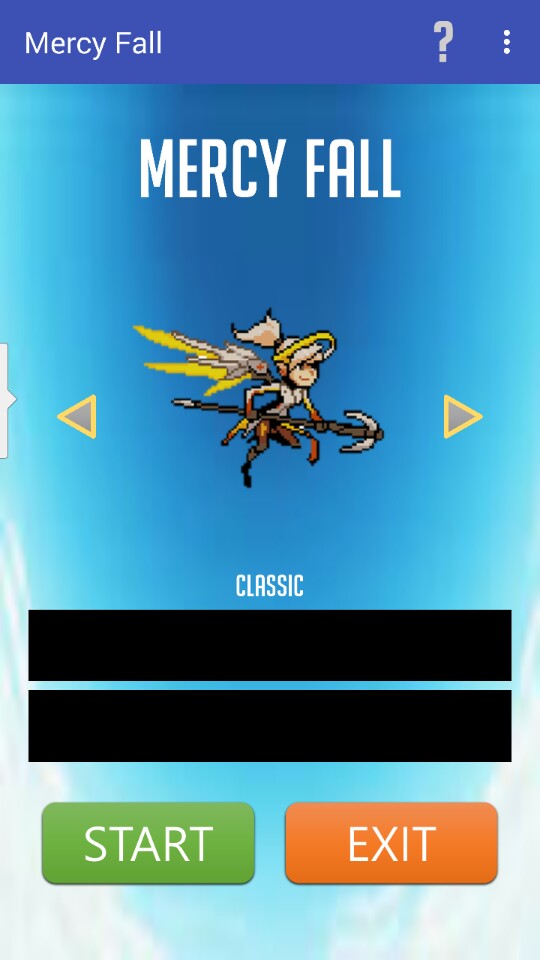
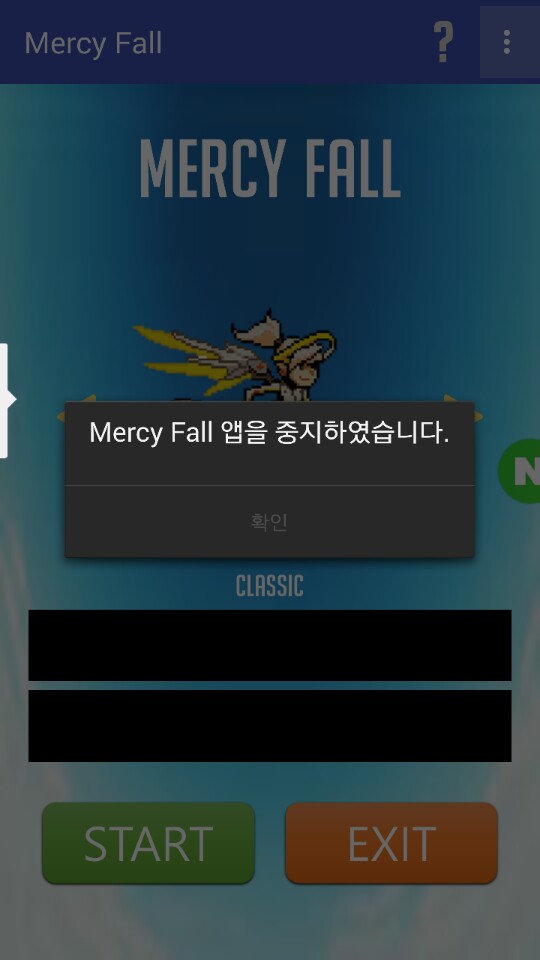
 

Figure 8,9 android 6.0보다 낮은 버전의 폰에서 실행한 화면

위에서 설명한 메뉴 옆에 물음표 아이콘의 메뉴가 하나 더 있다. 이 메뉴는 도움말 메뉴로, 터치하면 도움말을 보여주는 Help Activity를 실행시킨다.

아래는 도움말 메뉴를 눌렀을 때 나오는 Help Activity의 화면이다. 중간과제 때는 두 개의 페이지로 이루어져 있었으나, 새로운 아이템 하나와 장애물 두 개가 추가되면서 총 세 개의 페이지로 이루어 지게 되었다. 화면 하단에 현재의 페이지를 알려주는 수 양 옆에 배치된 Image Button을 이용해 페이지를 이동할 수 있다. 페이지는 Frame Layout을 이용해 쌓아놓고 각 페이지의 layout의 visibility를 조절해 보여주는 방법을 사용했다.

첫 번째 페이지에서는 차례대로 게임방법, 게임에서 등장하는 기능성 아이템의 이미지와 기능에 대한 설명이 나와있다. 따라서 아이템에 대한 설명은 여기서는 따로 하지 않겠다. 아래 그림에서 충분히 읽어볼 수 있다.

두 번째 페이지에서는 점수형 아이템과 장애물에 대한 설명이 나와있다. 점수형 아이템이 두 개가 되어서 각각 점수를 표시했다.

세 번째 페이지에서는 Main Activity에서 선택할 수 있는 메르시 스킨을 보여준다. 총 12개의 스킨 도트 이미지가 있다.

Help Activity의 Title Bar에는 X표 메뉴를 넣어, 터치하면 메인화면으로 돌아올 수 있도록 했다.

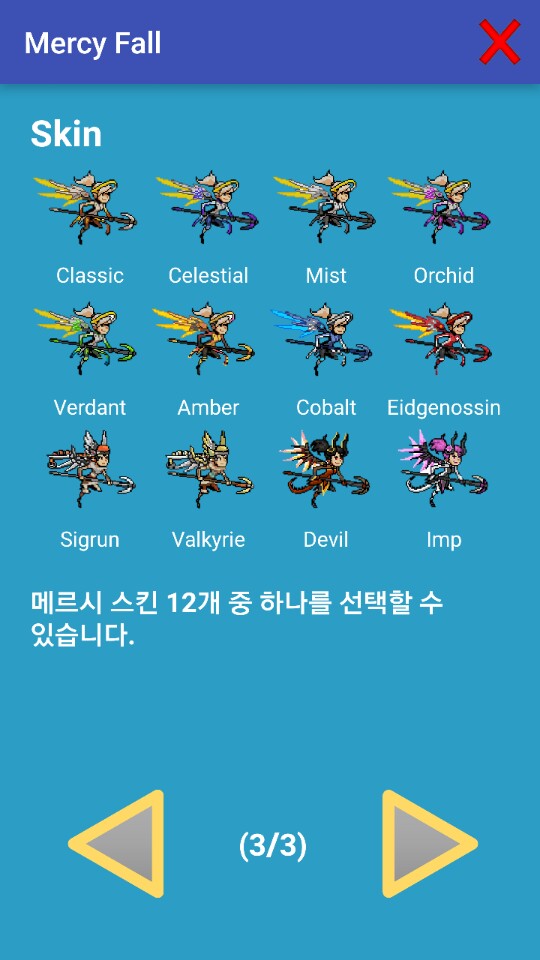
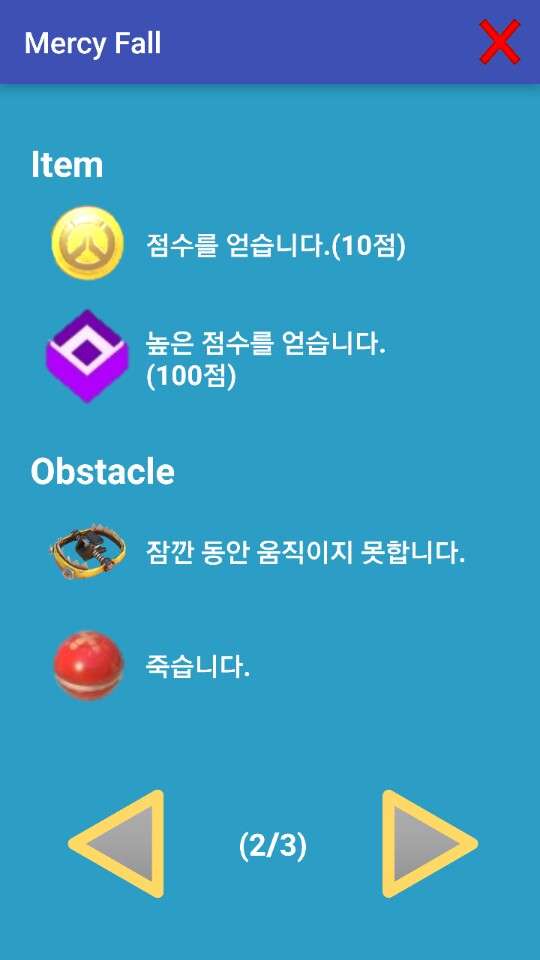
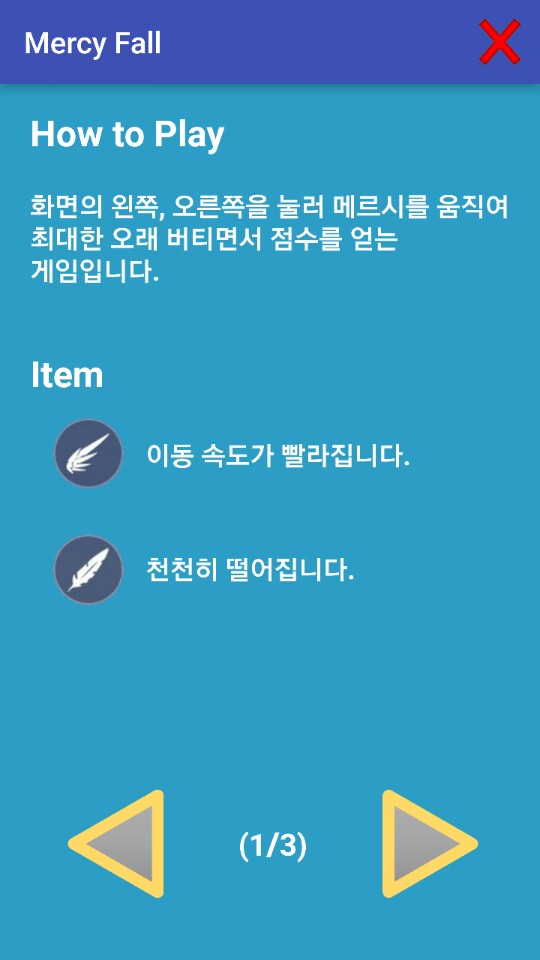


Figure 10,11,12 Help Activity의 첫 페이지(왼쪽)과 두 번째 페이지(오른쪽)

마지막으로 실제 게임을 플레이하는 Game Activity의 화면이다.

왼쪽 상단 Title Bar 밑에는 플레이 시간과 얻은 점수가 나와 있다.

화면 상단의 가운데에는 메르시가 떨어질 남은 거리가 나와 있다.

Game Activity의 Title Bar에도 마찬가지로 X표 메뉴를 넣어, 터치하면 메인화면으로 돌아올 수 있도록 했다.

Game Activity는 xml 대신 view를 만들고 onDraw로 canvas에 그리는 방법을 사용했다. 확실히 xml보다는 canvas가 비트맵 이미지를 그리는 것이 훨씬 편했다.



Figure 13 실제 게임 플레이 화면

게임을 플레이해서 메르시가 땅에 떨어지거나, 폭탄에 맞아서 죽거나 해서 게임이 끝나게 되면 아래 화면처럼 기록을 보여준다.





Figure 14,15 기록을 갱신한 경우(왼쪽)과 기록을 갱신하지 못한 경우(오른쪽)

기말과제의 핵심인 DB 사용은 당연히 최고 점수와 시간을 기록해 놓는데 사용했다. 최고 점수와 최대 시간만 저장해 놓으면 되므로 DB table에 단 하나의 데이터만 만들어서 값만 계속 갱신해주는 방식으로 했다.



Figure 16 게임을 다시 실행시키면 이제 예전의 기록이 남아있게 된다

**5. 수학 과학적 원리**

메르시는 하늘에서 떨어지면서 중력과 공기저항력을 받는다. 코드에서는 메르시가 떨어지는 속력에 중력에서 공기저항력을 뺀 값을 계속 더해주는 식으로 계산을 해서 적용시켰다. 공기저항력이 속력에 비례하므로 어느 순간부터는 중력과 공기저항력이 같아져 일정한 속도를 유지하게 된다.

천사 강림(천천히 떨어지게 하는 아이템)을 먹게 되면 공기저항을 늘어나게 해서 조금 더 천천히 떨어지게 했다. 이를 이용하면 플레이 시간을 늘리고, 점수를 더 많이 낼 수 있다.

**6. 차별화되는 점**

이런 비슷한 류의 모바일 게임들은 캐릭터가 아래에 있고 위에서 떨어지는 아이템을 먹게 되는데, 이 게임은 오히려 캐릭터가 위에서 떨어지고 하늘에 떠 있는 아이템을 먹는 방식을 선택했다. 또한 떨어지는 높이도 제한을 두어서 플레이 시간에 제한을 두었다. 이를 통해 플레이 시간과 점수 두 가지를 최대화하게 했다. 그리고 여러 아이템과 장애물을 추가하였다.

.

**7. 자랑할 만한 부분**

onTouchEvent가 같은 위치를 계속 누르고 있으면 호출되지 않아 꾹 누르고 있을 때 메르시가 움직이지 않는다. 이를 해결하기 위해 Handler와 Runnable을 사용해 화면의 같은 위치를 누르고 있어도 메르시가 방향에 맞게 움직이도록 했다.

메르시가 좌우로 움직일 때 움직이는 방향을 보도록 하고 싶었는데 12개나 되는 도트 이미지를 뒤집어서 넣기가 귀찮았다. 그래서 flip이라는 method를 만들어 행렬을 이용해 뒤집힌 비트맵 이미지를 반환하게 했다.

Linear Layout, Relative Layout, Table Layout, Frame Layout 등 xml에서 수업 시간에 배운 다양한layout을 사용했다.

글자를 넣을 때 기본 폰트가 마음에 들지 않아서 app/src/main에 assets라는 android resource directory를 만들고 그 안에 fonts라는 directory를 만들었다. 여기에 내가 사용하고 싶은 폰트를 넣어 가져다 쓸 수 있게 했다. Koverwatch 폰트의 ttf 파일을 넣어놓고 Typeface라는 것을 이용해 메인 화면의 제목이나 시간, 점수 등을 표시하는 Text View에 적용시켰다.

예쁜 비트맵 이미지를 구해 onDraw() method로 canvas에 그렸다.

DB를 추가하여 이제 예전 플레이 기록이 남아있도록 하였다.

**8. 중간과제와 달라진 점**

점수형 아이템이 하나 추가되었다.

그리고 장애물을 두 가지 새로 만들었다.

Handler 시간 간격을 100으로 해놨더니 약간 느려서 50으로 줄였다. 때문에 게임 관련 수치들을 다 뜯어 고쳐야 했다.

아이템과 장애물이 추가되면서 등장 확률을 약간 조절했다.

아이템과 장애물을 추가하는 바람에 도움말 페이지도 다시 만들어야 했다. 페이지 수가 세 페이지로 늘어나면서, 페이지를 넘어가게 하는 부분도 고쳐야 했다. FrameLayout을 사용했기 때문에 LinearLayout 배열을 만들어서 보이는 페이지를 바꾸는 것은 어렵지 않았다.

가장 중요한 DB의 추가는 당연히 했다. Table에 겨우 데이터를 하나만 넣지만 DB를 쓰긴 했다.