

Final Checkpoint

Team #2

朝 星 暮 四

122576 김도현

134239 조성모

134292 한효정

1. 목표 어플리케이션 이름

돌직구 더 지니어스

2. 개발 동기

처음에 게임을 만들자고 정했을 때 게임의 목적을 먼저 생각했다. 바로 경쟁이다. 이 때문에 상대방과의 경쟁을 통해서 조금이라도 승리의 기쁨을 얻을 수 있는 게임을 생각했다. 슈팅게임(갤러그), 퍼즐버블 등 현재 우리가 할 수 있는 능력을 고려한 게임을 생각한 것이 몇 가지 있었다. 그 중에서 숫자야구가 문득 떠올랐고 목적에 적합했다. 그리고 2인이 게임을 함으로써 경쟁심을 끌어낼 수 있고 분석, 추리 등의 두뇌활동에도 도움이 된다. 기존의 숫자야구의 직접 필기해야하고 숫자를 외워야하는 등의 불편함을 개선하고 야구 관련된 효과와 부가적인 기능을 통해 pc게임으로 발전시키고자 하였다. 또 시각적, 청각적 효과를 넣어서 더 재미있게 하기 위하여 개발하기로 하였다.

3. 특징

두 명이 숫자야구를 대결구도로 하는 것이 특징이다. 물론 혼자서 하는 것이 가능하나 두 명이서 게임을 하여 경쟁심을 돋우는 것이 목적이다. 혼자 하는 기능은 랜덤으로 숫자를 저장해두고 그 숫자를 맞추는 방식이다. 설정을 통해 맞추려는 숫자의 개수를 3개와 4개로 설정을 할 수 있게 한다. 또 특별한 것이 있다. 평소에 우리가 숫자야구를 할 때는 종지와 펜을 이용하여 기록했다. 그러나 우리는 따로 공간을 두어 추측했던 내용들을 화면에 띄울 것이다. 이 밖에도 소소한 모션을 통해 재미를 더할 것이다.

4. Function Specification

1. 숫자의 개수를 선택할 수 있게 설정. bgm 음소거 기능.
2. 게임설명 : 게임을 하기 위한 조작 및 규칙이 나타남.

●게임 시작

1. 혼자하기 : 랜덤으로 설정된 숫자로 게임을 혼자 진행.
2. 같이하기 : 1대1 대결이며, 각자 게임 전에 숫자를 입력하여 상대방의 숫자를 추리하며 게임진행. 상대방의 숫자를 먼저 맞추는 player가 승리한다.

●설정

추리하는 숫자의 개수를 3개와 4개로 설정할 수 있다. 음소거 버튼을 통해 bgm을 없앨 수 있다.

●게임설명

숫자야구의 규칙을 그대로 유지하면서 게임을 이어갈 것이다. 스트라이크(S)는 숫자와 그 숫자의 위치(몇 번째인지)를 맞출 경우 인정이 된다. 볼(B)은 숫자는 맞췄으나 위치를 맞추지 못한 경우이다. 이를 토대로 만약 숫자가 4개인 모드로 게임을 할 경우, 최대 4S(4스트라이크: 4개의 숫자를 순서까지 모두 맞출 경우)부터 OUT(한 개의 숫자도 못 맞출 경우)까지 기록이 된다. 두 사람이 게임을 하면 player1이 먼저 숫자를 입력하고 그 결과가 화면의 전광판에 바로 기록이 된다. player2의 순서가 되면 player1이 입력한 내용은 미리 마련해 두었던 공간에 기록된다. 각 차례마다 시간이 주어지고 그 시간 안에 숫자를 입력하지 못할 시 다음 순서로 넘어가게 된다. 이러한 규칙 안에서 게임이 이뤄지고 player중 한 명이 4S가 나올 경우 승패가 가려지고 게임이 끝이 난다. 이 모든 과정을 간략한 그림으로 설명함.

●배경

게임시작 시 화면은 야구의 느낌을 주기 위해서 야구장을 배경으로 설정했다. 각 player가 숫자를 입력할 때마다 투수의 모션이 바뀌고 전광판에 숫자와 그에 대한 결과가 스트라이크, 볼, 아웃의 형태로 출력된다. 상대방의 차례로 넘어갈 시 전광판의 내용은 해당 player의 노트(따로 마련해둔 공간)에 기록된다. 승리 시 승리한player가 효과와 함께 화면에 나타난다. 그 후 새게임버튼(게임을 새로 시작)과 홈버튼(메인화면으로 이동)이 나타난다.

●음향

매 화면에 맞는 야구 관련 효과음을 삽입한다. 그 외에 bgm 삽입함.

5. Requirement Lists

1. 초기 화면은 야구장이 배경화면이고, 게임시작과 게임설명 버튼 우측 하단에 Sound Check Box가 있고, 우측 상단에 X버튼으로 게임 종료를 할 수 있다.
2. 게임설명 버튼을 누르면 게임설명창이 팝업되고, X버튼으로 끌 수 있다.
3. 초기 화면 우측 하단의 Sound Check Box의 Default 값은 Check이고, Check가 켜져있으면 Sound on, 꺼져있으면 Sound off가 된다.
4. 게임 시작 버튼을 누르면 1인모드, 2인모드, 3구, 4구 모드로 선택 할 수 있는 창이 나타난다. 한 칸에만 체크가 가능하고, 아무것도 체크하지 않으면 게임진행버튼이 활성화 되지 않는다.
5. 게임 진행 중에 우측 상단에는 음영진 글자로 PAUSE: ESC key가 나타난다.
6. ESC key를 누르면 게임이 중지되고 “홈으로”, “다시하기”, “돌아가기” 버튼과 Sound Check Box가 나타난다.
7. 모드를 선택 한 후 게임시작을 하면 1인 모드는 혼자서 임의의 수를 맞추면 게임이 끝나고 2인 모드는 각자의 수를 맞추게 되면 게임이 끝난다.
8. 2인 모드를 하게 되면 게임을 진행하기 전에 각 플레이어가 자신의 숫자를 3개 혹은 4개 입력 할 수 있어야 하고 그 숫자들이 저장되어야 한다.
9. 게임 진행 중 중앙 화면에는 유추한 숫자를 입력 할 수 있는 칸, 입력 한 결과가 Strike, Ball, Out으로 나타나는 전광판, 제한 시간을 나타내는 시계가 있어야 한다.
10. 게임 진행 중 화면 상단에서 투수캐릭터가 있고, 각 플레이어가 숫자를 유추할 때마다 와이드업 자세를 취하며, 공이 플레이어 쪽으로 날아와서, 그 공에 입력한 숫자와 결과가 나타난다.

11. 8번에서 입력한 숫자의 결과는 중앙에 있는 전광판에 기록되어야 한다.
12. 3구모드는 9회, 4구모드는 12회까지 게임이 진행된다.
13. 특정 회에 플레이어 한명이 3Strike 혹은 4Strike를 하게 되면 그 플레이어가 승리하게 된다.
14. 같은 회에 두 플레이어 모두 3Strike 혹은 4Strike를 하거나 제한 회 내에 경기가 종료 되지 않으면 무승부가 되어 게임이 종료 된다.
15. Game end 화면 중앙에는 어떤 플레이어가 승리 했는지, 혹은 무승부인지를 알려주는 text가 나타나게 되고 홈버튼, 다시하기 버튼, 게임종료 버튼이 나타난다.

6. Usecases

UseCase

1. 메인 화면

1. 화면의 배경은 야구장이 배경으로 나타난다.
2. 화면의 우측 상단에는 X표시 버튼이 있고 누르면 게임이 종료 된다.
3. 화면의 우측 하단에는 Sound Check Box가 있고 기본적으로는 박스가 체크되어 있다. 체크 되어 있으면 배경음이 들리게 하고 체크 되어 있지 않으면 음소거가 되게 한다.
4. 화면의 중앙에는 게임시작 버튼, 게임 설명 버튼이 나타난다.
 - 4-1. 게임 설명 버튼을 누르면 게임을 설명하는 설명창이 팝업 되고 팝업창은 우측 상단의 x 버튼을 누르면 종료되게 한다.
 - 4-2. 게임 시작 버튼을 누르면 게임 모드 선택 창으로 넘어간다.

2. 게임 시작

1. 게임 시작 버튼을 누른 후에는 게임 모드를 설정 할 수 있는 창이 나타나게 되고 플레이어가 자신이 원하는 모드를 선택 한 후 하단의 게임 진행 버튼을 누르면 게임이 진행되고 만약에 모드를 선택하지 않은 경우에는 게임진행 버튼이 활성화 되지 않아서 게임이 진행되지 않는다.

3. 게임 진행

1. 1인 3구, 1인 4구 모드를 선택하게 되면 임의로 추측해야 할 숫자 3개 혹은 4개가 설정된다.
 - 1-1. 화면의 상단에는 투수 캐릭터, 중앙에는 입력창과 전광판, 제한 시간이 나타난다.
 - 1-2. 투수 캐릭터는 숫자 입력 중에는 와인드업을 하고 숫자 입력이 완료되면 투구를 한다.
 - 1-3. 투수가 던진 공은 투수를 떠나 중앙의 표와 중간에 한번 작게 나타나고 전광판 근처에 와서 크게 확대된 다음에 플레이어가 입력한 숫자의 결과를 출력한다.
 - 1-4. 플레이어가 추측한 숫자를 키보드로 입력하면 전광판 아래의 원형 칸에 숫자가 나타나게 되고 이 칸은 입력될 칸은 밝은 회색, 그렇지 않은 칸은 어두운 회색으로 나타나고 숫자를 지울 때는 키보드의 back space key를 누르고 숫자를 지우면 원형 칸의 숫자도 지워지며 모든 숫자를 다 입력 하고 enter를 누르면 추측해야 할 숫자와 비교하는 작업이 진행된다.
 - 1-5. 입력한 숫자는 투수가 던진 공에 나타나고 나타난 결과는 각 회마다 아래 전광판에

Strike와 Ball로 기록된다.

1-6. 처음에 임의로 설정된 숫자를 플레이어가 9회 혹은 12회 안에 맞추게 되면 플레이어가 게임을 승리하게 되고 그렇지 않으면 패배하게 된다.

2. 2인 3구, 2인 4구 모드를 선택하면 각 플레이어가 맞춰야 할 secret number를 3개 혹은 4개를 입력하는 창이 나타난다.

2-1. 게임이 진행 되면 다른 창은 1인 모드와 동일하고 전광판만 Player1과 Player2로 나뉘어져 나타난다.

2-2.가 다른 기능들은 3.1의 하위에서 진행된 것과 똑같은 기능을 한다.

2-3.나 처음에 각 플레이어가 설정한 숫자를 9회 혹은 12회 안의 특정 회에 두 플레이어 중 한 플레이어만 3Strike 혹은 4Strike를 하게 되면 그 플레이어가 승리하게 된다.

2-4.다 만약 9회 혹은 12회 내에 승리가 결정되지 않거나 한 회에 두 플레이어 모두 각 플레이어의 숫자를 전부 맞추게 되면 게임은 무승부가 된다.

4.게임 중지

1. 게임 진행 중 게임 화면의 우측 상단 쪽에는 음영진 회색 글씨로 PAUSE: ESC key가 나타난다.

2. ESC key를 키보드로 누르게 되면 진행되던 게임화면이 중지되고 화면의 중앙에 “홈으로”, “다시하기”, “돌아가기” 버튼과 Sound Check Box가 나타난다.

2-1. “홈으로”버튼을 누르면 현재 진행중이던 게임을 초기화 하고 초기 화면으로 돌아간다.

2-2. “다시하기”버튼을 누르면 현재 진행중이던 게임을 초기화하고 1인모드이면 새로운 secret number가 임의로 설정되고 2인 모드이면 각자의 숫자를 다시 입력하고 현재 모드로 1회부터 시작한다.

2-3. “돌아가기” 버튼을 누르면 현재 진행중이던 게임을 이어서 진행한다.

2-4. Sound Check Box를 체크하면 소리가 나고 체크하지 않으면 음소거를 한다. 게임을 중지하여도 게임의 배경음은 계속 play한다.

5.게임 종료 화면

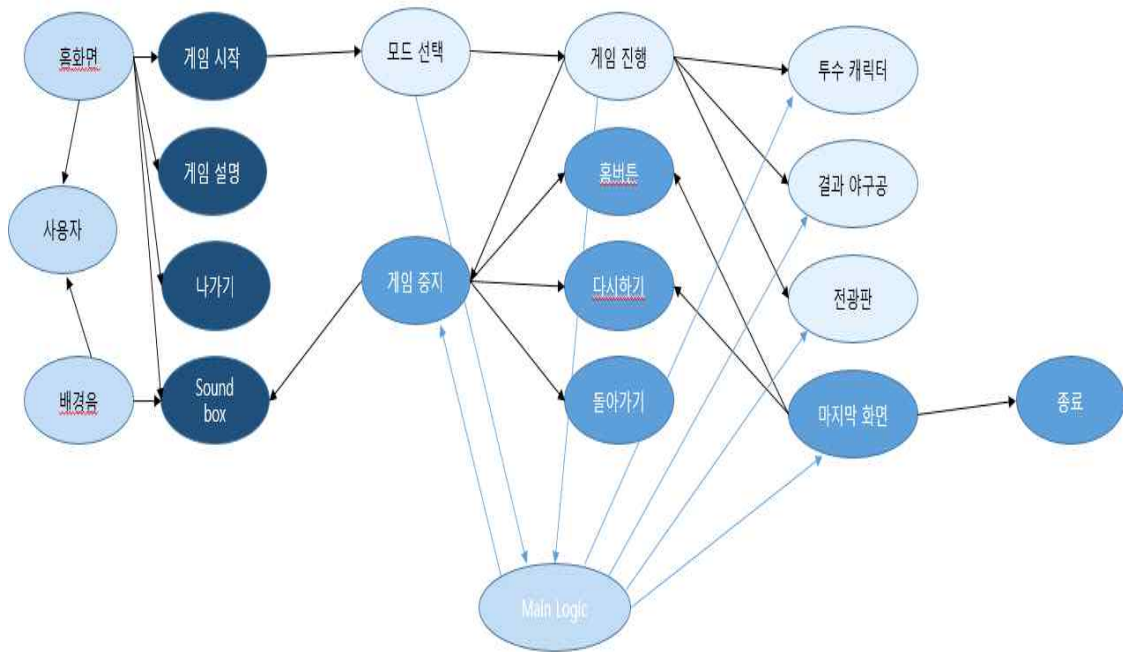
1. 게임의 결과가 결정되면 화면의 중앙에 누가 승리 하였는지, 무승부인지, 패배 하였는지를 알려주는 text가 출력되고 그 아래에 홈 버튼, 다시하기 버튼, 게임 종료 버튼이 활성화 된다.

1-1. 홈 버튼을 누르면 진행 되었던 게임의 데이터가 초기화 되고 초기 화면으로 돌아간다.

1-2. 다시하기 버튼을 누르면 현재 게임 모드로 게임을 1회부터 다시 시작한다.

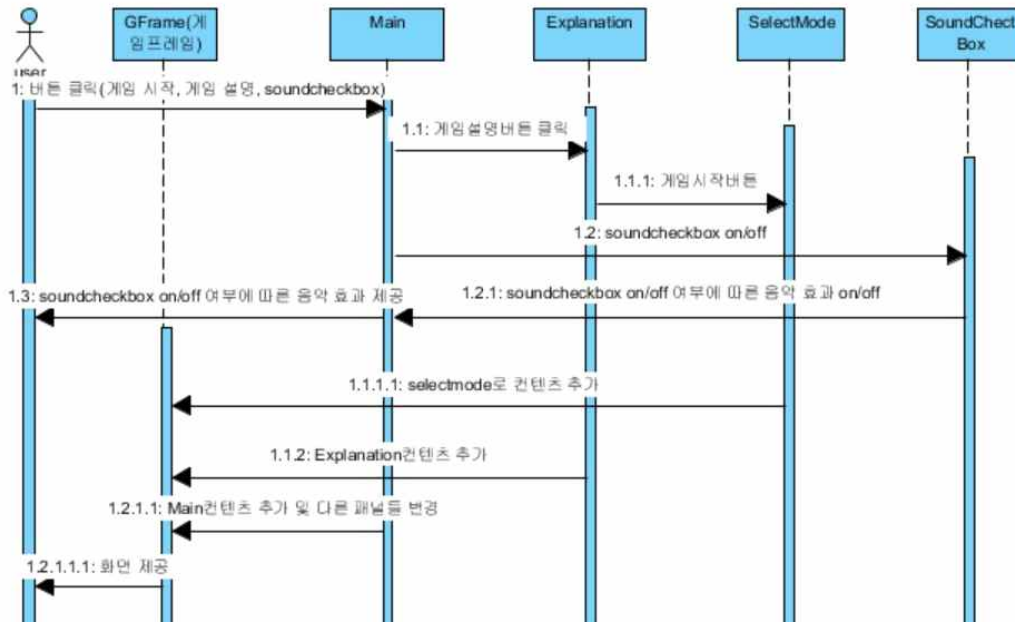
1-3. 게임 종료 버튼을 누르면 3초간 “게임을 종료합니다.”라는 text가 출력하고 게임 화면이 사라진다.

7. Usecase Diagram

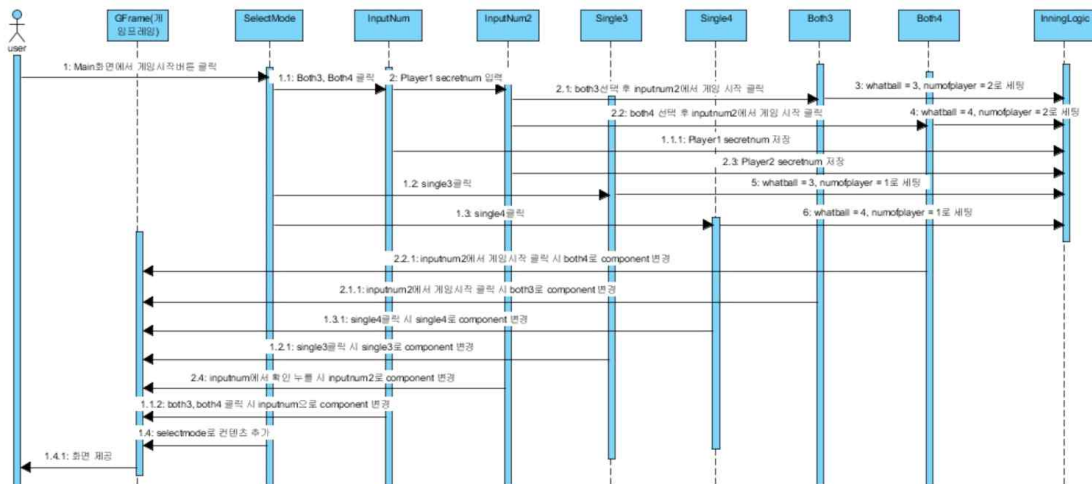


8. Sequence Diagram

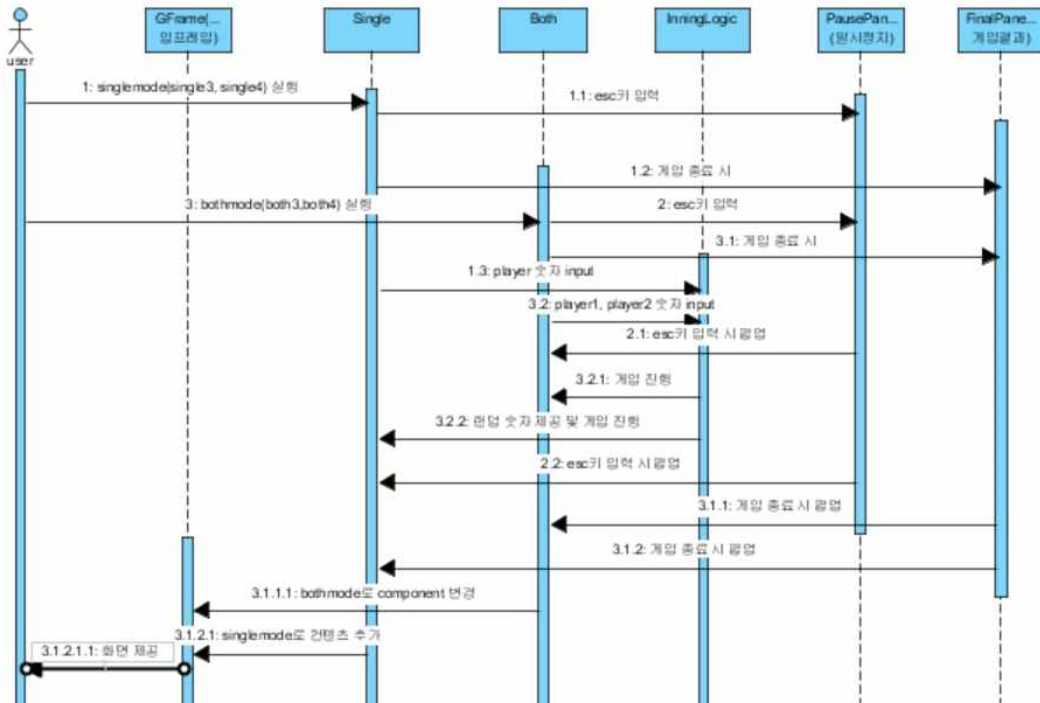
1-게임 시작 화면



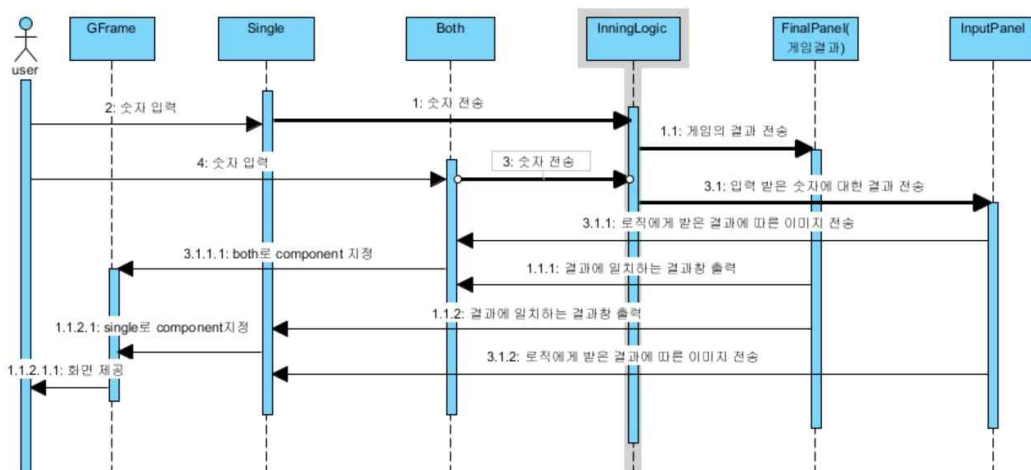
2-게임 모드 선택



3-게임 진행 중



4-Main Logic



9. User Interface Designs

1. 게임 시작 화면


볼직구

홈 화면

- Sound Check box**
- 체크가 되어 있으면 소리가 나고 체크가 없으면 음소거
- 게임시작 버튼**
- 클릭 시 모드 선택창으로 이동
- 게임설명 버튼**
- 클릭 시 게임 이용에 관한 설명이 있는 창이 팝업
- 스fx 버튼**
- 클릭 시 프로그래밍 종료



2. 게임 설명

게임설명 화면


- 게임 설명창**
- 그림과 함께 이해하기 쉬운 게임 설명창이 팝업 되어 시각적으로 보여진다.



3. 모드 선택 창

모드 선택 화면

- 1인 3구 모드**
- 혼자서 하는 3자리수 모드이다.
- 1인 4구 모드**
- 혼자서 하는 4자리수 모드이다.
- 2인 3구 모드**
- 둘이서 하는 3자리수 모드이다.
- 2인 4구 모드**
- 둘이서 하는 4자리수 모드이다.
- 게임 진행 버튼**
- 누를 시 게임진행창으로 변경



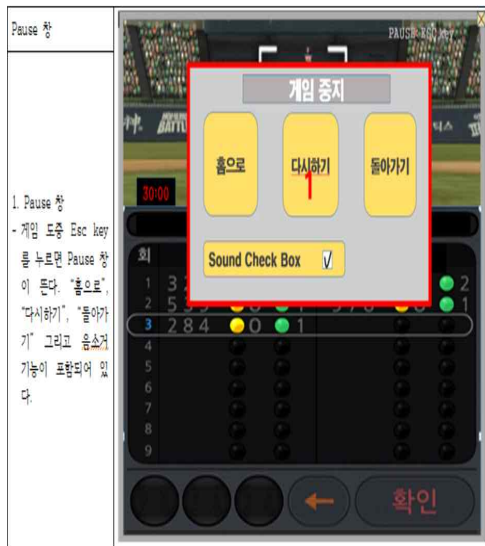
4. 게임 진행 창

게임 진행 화면

- Pause**
- Esc key 를 누르면 Pause창이 뜬다.
- 투수 그림**
- 숫자 입력 중 투수의 그림이 움직이고 입력 완료시 투수가 투구를 할과 동시에 야수들의 결과가 뜬다.
- 타이머**
- 30초 안에 투수한 숫자를 입력해야한다.
- 차례 알림이**
- 누구의 차례인지 알려준다.
- 전광판**
- 투수한 숫자의 결과에 대한 내용이 저장된다. 5는 노란색 또는 초록색, 이닝 정보는 가지고 있다.
- 숫자 입력기**
- 숫자를 입력한다.



5. 게임 중지 창



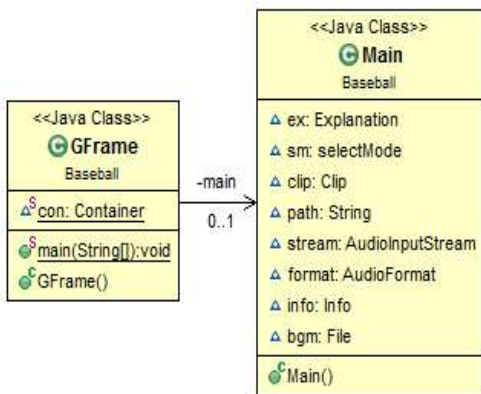
6. 게임 종료 창



10. Class Diagram

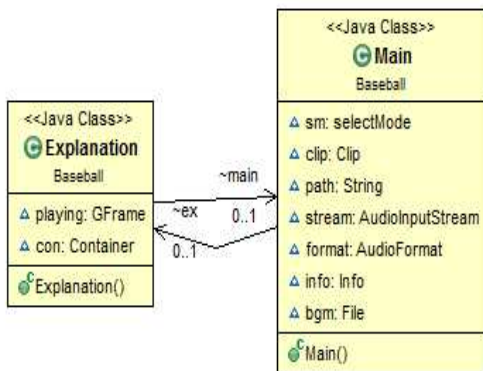
1.Main

프로그램이 시작 될 시 처음으로 나타나는 메인 화면을 정의한 클래스로 게임시작 버튼을 누르면 게임 모드 선택 창으로 가게 되고, 게임 설명 버튼을 누르면 게임 설명 팝업창이 뜨고 우측 하단의 soundCheckBox를 체크하면 배경음이 들리고 체크가 안되었으면 배경음이 들리지 않게 한다.



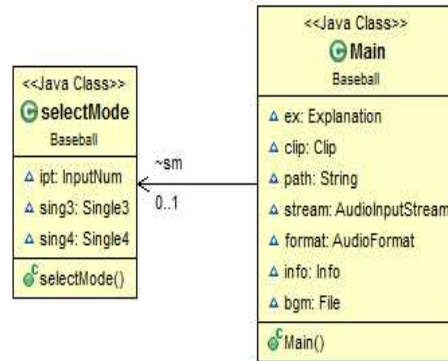
2.Explanation

메인 화면에서 게임 설명 버튼을 누르면 팝업된 설명 창을 구현한 클래스로 게임의 규칙을 설명하는 이미지를 보여주고, 우측 상단의 x버튼을 누르면 설명 창이 사라지게 한다.



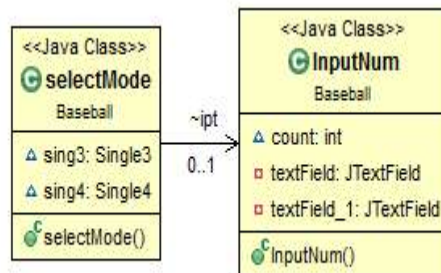
3.selectMode

게임 선택창을 구현한 클래스로 총 4개의 버튼에서 각 버튼을 누를 때 1인 3구, 1인 4구, 2인 3구, 2인 4구 모드로 게임이 진행된다.



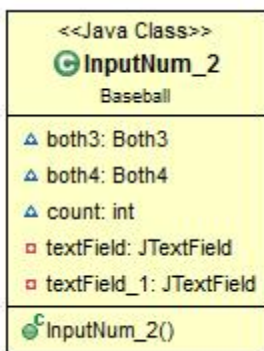
4.InputNum

2인 모드 선택시에 Player1의 초기 숫자 입력을 받기 위해 정의한 클래스로 마우스가 숫자 버튼을 누를 때 그 숫자를 메인 로직의 벡터에 저장하고 게임에 필요한 초기숫자를 모두 입력한 후에는 확인 버튼이 활성화 되어 누르면 Player2의 초기 숫자를 입력 받기 위한 창으로 전환된다.



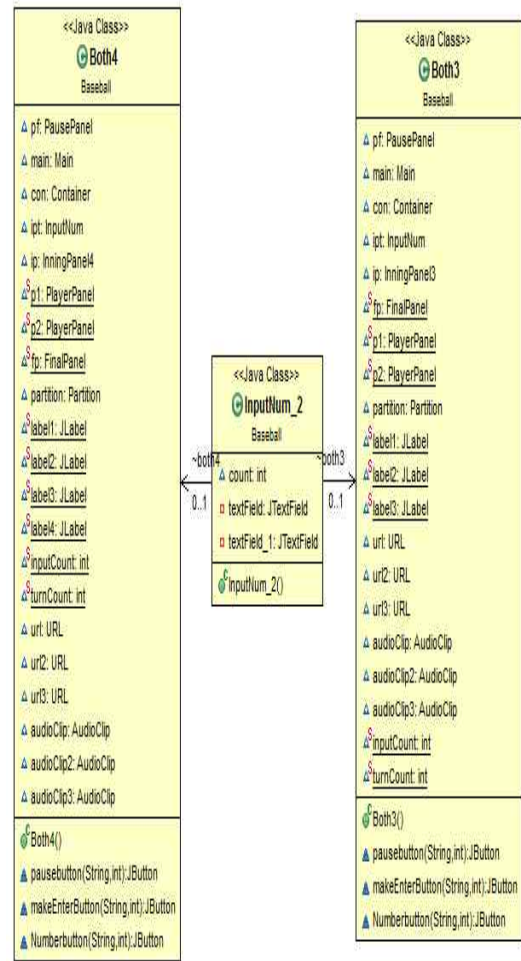
5.InputNum_2

2인 모드 선택시에 Player1의 숫자를 입력한 후에 Player2의 초기 숫자 입력을 받기 위해 정의한 클래스로 마우스가 숫자 버튼을 누를 때 그 숫자를 메인 로직의 벡터에 저장하고 게임에 필요한 초기숫자를 모두 입력한 후에는 게임시작 버튼이 활성화되어 누르면 게임이 시작 된다.



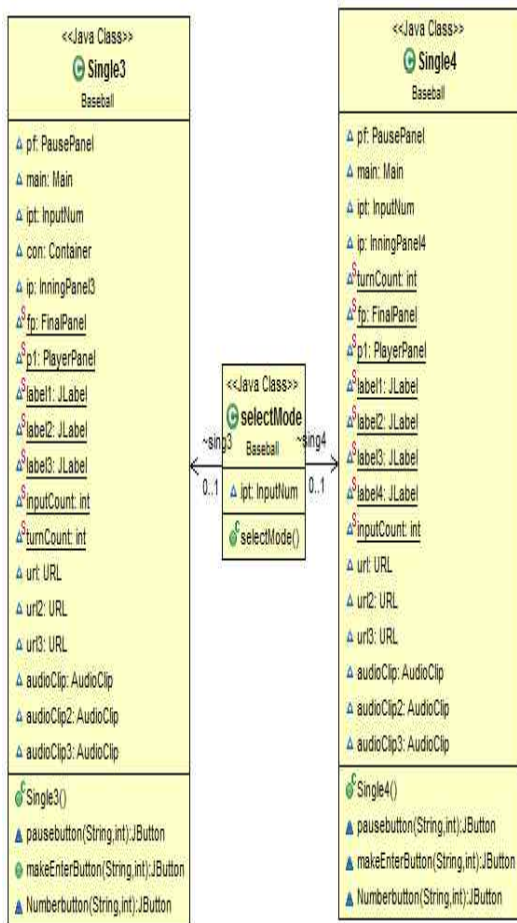
6.Both3 / Both4

selectMode에서 2인 모드 설정 시 Player2의 초기숫자를 입력받고 게임시작 버튼을 눌렀을 때 2인 3구, 2인 4구모드 게임이 진행되는 화면을 구현한 클래스로 게임 진행에 필요한 2명의 Player의 게임 진행 상황을 보여주는 전광판, 추측한 숫자 입력 기능, 게임 중지 창, 상황에 따라 모션의 변화를 일으키는 투수와 공, 게임 결과 창으로 구성되어 있다.

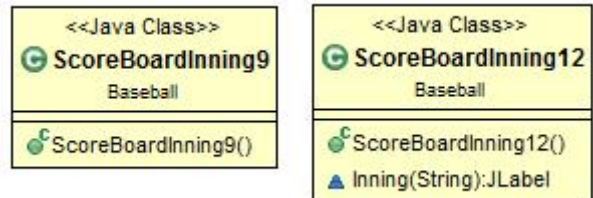


7.Single3 / Single4

selectMode에서 1인 모드 설정 시 바로 진행되는 1인 3구, 2인 4구모드 게임이 진행되는 화면을 구현한 클래스로 게임 진행에 필요한 단일 Player의 게임 진행 상황을 보여주는 전광판, 추측한 숫자 입력 기능, 게임 중지 창, 상황에 따라 모션의 변화를 일으키는 투수와 공, 게임 결과 창으로 구성되어 있다.

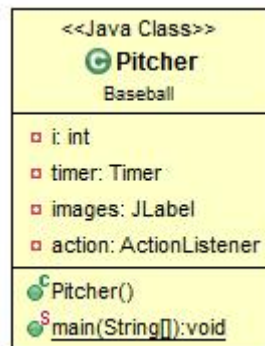


8.ScoreBoardInning9 / ScoreBoardInning12
 3구 모드, 4구 모드에 따라 9회 전광판, 12회 전광판의 바탕을 형성하는 클래스



9.Pitcher

Player가 추측한 숫자를 입력하고 enter key 를 눌렀을 때 투수가 공을 던지는 모션을 구현하는 클래스



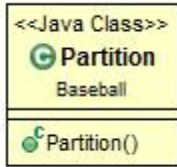
10.PlayerPanel

전광판에서 각 플레이어의 추측 결과가 나타나는 바탕이 되는 Panel을 구현한 클래스



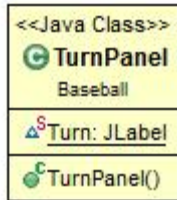
11.Partition

2인 모드에서 각 PlayerPanel을 구분하기 위한 Panel을 구현한 클래스



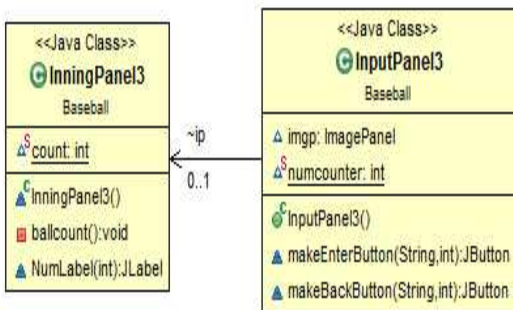
12. TurnPanel

2인 모드에서 어떤 player가 숫자를 입력할 차례인지를 보여주기 위해 전광판 상단에 존재하는 Panel을 구현한 클래스



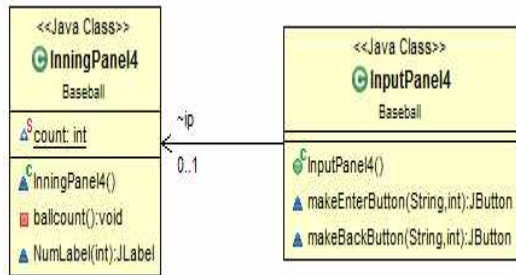
13.InputPanel3/InningPanel3

3구 모드에서 InputPanel3는 숫자를 입력하고 지우기 위한 기능을 구현한 클래스 / InningPanel3 는 그렇게 입력 받은 숫자와 초기에 입력한 숫자를 비교한 결과에 따라 PlayerPanel에 황등, 청등으로 이미지를 보여주는 것을 구현하는 클래스



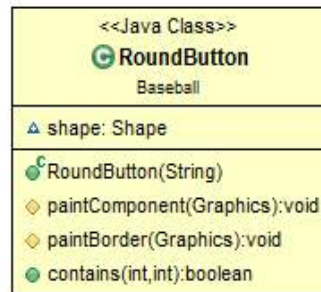
14.InputPanel4/InningPanel4

4구 모드에서 InputPanel4는 숫자를 입력하고 지우기 위한 기능을 구현한 클래스 / InningPanel4 는 그렇게 입력 받은 숫자와 초기에 입력한 숫자를 비교한 결과에 따라 각각의 PlayerPanel에 황등, 청등으로 이미지를 보여주는 것을 구현하는 클래스



15.RoundButton

동그란 모양의 버튼인 enter버튼과 back space 버튼에 동그란 모양으로 색깔을 배치하는 것을 구현한 클래스



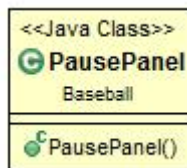
16.ImagePanel

전광판 위쪽의 야구장 배경과 투수 사진을 나타내기 위한 클래스



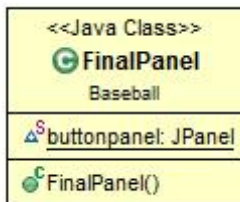
17. PausePanel

게임 진행 중에 esc key를 눌렀을 때 메인화면으로 가는 홈 버튼, 원래 게임창으로 돌아가는 돌아가기 버튼, 게임을 종료하는 게임 종료 버튼을 포함한 게임중지 창을 구현한 클래스



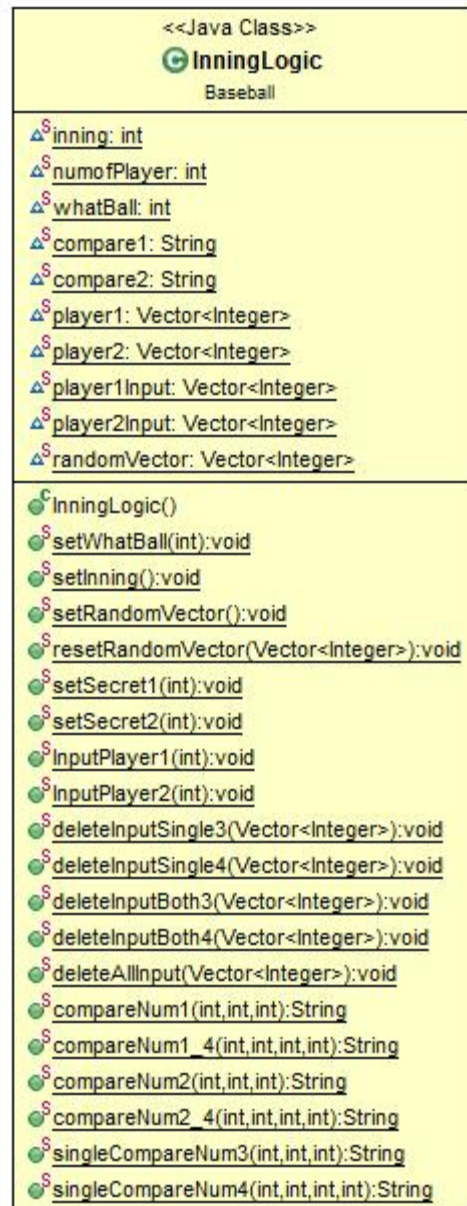
18. FinalPanel

게임이 한 player가, 혹은 두 player가 동시에 3스트라이크 혹은 4스트라이크를 하거나 제한 inning수를 넘길 시 게임의 승, 패, 무승부 여부를 알려주고 메인 화면으로 가는 홈버튼, 같은 모드로 재시작 하는 다시하기 버튼, 게임을 종료하는 게임종료 버튼을 구현하는 클래스



19. InningLogic

게임의 진행, 결과 도출을 위한 멤버들과 숫자를 저장하고, 지우고, 비교하는 등의 메소드들이 구현된 게임의 main logic을 담당하는 클래스



11. Screenshot of API Manual

1. 초기 화면

PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

PREV CLASS NEXT CLASS FRAMES NO FRAMES ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Class InningLogic

java.lang.Object
InningLogic

public class **InningLogic**
extends java.lang.Object

게임에 필요한 페인 로직 1인 3구, 2인 3구, 1인 4구, 2인 4구 등의 로직이 정의되어 있다.

Constructor Summary

Constructors

Constructor and Description

InningLogic()

Method Summary

All Methods Static Methods Concrete Methods

Modifier and Type Method and Description

2. 메인 화면

Class Main

java.lang.Object
java.awt.Component
java.awt.Container
javax.swing.JComponent
javax.swing.JPanel
Main

All Implemented Interfaces:
java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable, javax.accessibility.Accessible

public class **Main**
extends javax.swing.JPanel

메인 화면의 패널 클래스 Explanation, selectMode를 포함 wav를 통해 소리를 나게 하기 위한 Clip, path, AudioInputStream, AudioFormat, DataLine.Info, bgm 등을 포함

See Also:
Serialized Form

Nested Class Summary

Nested classes/interfaces inherited from class javax.swing.JComponent

javax.swing.JComponent.AccessibleJComponent

3.Single3

Class Single3

```
java.lang.Object
  java.awt.Component
    java.awt.Container
      javax.swing.JComponent
        javax.swing.JPanel
          Single3
```

All Implemented Interfaces:

java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable, javax.accessibility.Accessible

```
public class Single3
  extends javax.swing.JPanel
```

게임모드 1인 3구 모드 화면인 패널클래스 Single3 PausePanel, Main, InputNum, InningPanel3, Partition을 포함 Sound 효과를내기 위한 URL, AudioClip을 포함

See Also:

Serialized Form

Nested Class Summary

Nested classes/interfaces inherited from class javax.swing.JComponent

12. Screenshot of application demo

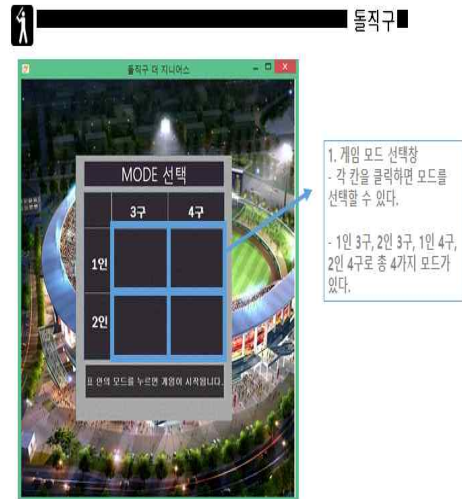
1. 게임 시작 화면



2. 게임 설명 창



3. 게임 모드 선택 창



4. 초기 숫자 입력 창



5. 게임 진행 창



7. 게임 결과 창



6. 게임 중지 창



13. 각 팀원 별 Work Distribution 및 진행 결과

1. 각 팀원 별 Work Distribution

1. 홈화면, 윈도우 화면, 게임시작, 게임설명 버튼 // 게임설명 팝업 - 도현, 효정
2. 게임모드선택 - 성모
3. 일시정지 팝업 - 도현, 효정
4. 이미지(배경, 모션 등등) - 도현, 성모, 효정
5. 게임진행 (전광판에 숫자출력) - 효정, 성모
6. 게임결과 창 - 성모, 효정
7. Main Logic - 성모
8. Sound - 도현

2. 진행 결과

김도현-Line Length:1049

저는 메인화면, 그리고 메인 화면에 있는 게임시작 버튼, 게임설명 버튼, 사운드 체크박스를 구현하였고, 또 게임 중간에 esc 키를 누르면 나오는 pause 팝업(PauseFrame class)과 게임 내에 나오는 Sound들을 삽입하였습니다.

메인화면의 경우는 한효정 학생에게 도움을 많이 받았고, 게임설명 버튼을 누르면 나오는 게임 설명창의 이미지를 구현하였습니다.

또, 배경음악, 게임 중에 나오는 효과음, 게임결과 창이 나올 때 나오는 배경음 등의 wav파일을 추출하여서 각 배경에 맞는 효과음, 배경음 등을 삽입하였습니다.

배경음악의 경우, Sound Check Box의 Default 값이 check된 상태이고, Sound Check Box가 check가 되어 있으면 배경음악이 나오게 하고, check를 끄면 배경음악이 나오지 않게 하였습니다.

그리고 게임시작이나 게임설명버튼을 누르면 배경음악이 꺼지게끔 구현하였습니다.

처음에는 AudioClip으로 구현하였지만, 그렇게 하면 체크박스를 껐다 켜면 음악이 처음부터 재생되는 문제가 발생하여서 다른 방법인 AudioInputStream, AudioFormat, Clip, DataLine.Info, File 등을 이용하는 방법으로 구현하였고, 그렇게 하면 체크박스를 껐다 켜도 음악이 처음부터 나오지 않고, 이어서 재생이 됩니다. Main 클래스와, pause 팝업에 있는 SoundCheckBox가 게임 중간에 나오는 효과음에도 영향을 주기 위해 isMute라는 메소드를 구현하여서 실행해봤는데 잘 되지 않았습니다.

그 이유는 버튼들을 객체로 만들어 놓지 않아서, 패널 별로 연결이 잘 되지 않은 것 같습니다.

또, pause 팝업에 있는 디자인 등을 맡아서 하였고, 돌아가기, 홈버튼, 다시시작 등의 버튼, 액션등을 구현하였습니다. 그리고 한효정 학생이 다른 패널로 이동하게끔 setVisible 하는 방법으로 구현하였습니다. 마지막으로 게임 결과 창에 나오는 배경음악을 게임 결과창이 뜨면 재생되게끔 구현하였습니다.

조성모-Line Length:1730

저는 게임 시작 화면에서 게임시작 버튼을 누를 시 player가 어떤 mode로 게임을 진행 할 것인지를 선택하는 게임 선택창을 구현하였습니다.

그리고 게임 진행 중에 필요한 숫자를 입력하고 숫자를 입력 했을 때 그 숫자가 이미지 상에 나타나게 하고, 숫자를 입력 할 때 쓰이는 enter, backspace기능을 구현 하였습니다.

3번째 발표때는 키리스너에 대한 공부가 완전히 되지 않아서 키를 입력 받았을 때 입력 받은 것이 무엇인지 프로그램에 전달 해 줄 수 없었기 때문에 그것을 화면에 출력하지도 못했지만 키리스너에 대해 공부하고 이벤트 핸들링에 대해서 더 연구하여서 최종적으로 숫자를 입력받고 화면에 출력해 줄 수 있었습니다.

그리고 게임이 최종적으로 종료 되었을 때 플레이어의 승, 패, 무승부 여부를 알려주고 게임을 재시작 하거나 처음 메인화면으로 돌아가거나 게임을 종료하는 버튼이 존재하는 게임 결과 화면을 구현하였습니다.

그리고 main logic인 InningLogic에는 게임 시작부터 게임이 종료하는데 까지 필요한 멤버들, 숫자를 입력하고 저장하고 비교하고 삭제하는 기능의 메소드들을 구현 하였습니다.

한효정-Line Length:2084

저는 전체적인 디자인 및 게임진행창(Single3, Single4, Both3, Both4)를 구현하였습니다. 저는 특히 각 패널들을 클래스로 만들어서 container에 넣어주는 방식을 사용했습니다.

각 패널들 역시 세부 패널클래스(inningpanel, pitcher, scoreboardinning, imagepanel 등)들을 정의 하여 각각의 큰 패널의 생성자에서 객체를 생성하여 추가하는 방식을 사용하여 게임진행창을 구현하였습니다.

아쉬운 점이 있다면 타이머를 구현하기로 했었지만 타이머 구현 중간에 오류가 발생하여 결국엔 해결하지 못했습니다.

그에 반해 투수이미지가 움직인다거나 전광판에 결과물을 출력해주는 것은 구현에 성공했고 프로그램도 잘 돌아갔습니다.

소스를 작성하면서 객체지향에 대해 계속 생각하였고 독립적인 클래스들을 정의하여 객체를 불러온 방식을 사용한 것에 대하여 객체지향에 꽤나 가까워졌다고 생각합니다.