

# 한이음 공모전 2017 참가신청서

## 작품 정보

프로젝트명	국문	증강현실로 즐기는 SKY JUMP / 패러글라이딩
	영문	Augmented reality Sky Jump / Paragliding
프로젝트기간	2017. 03 . 27 . ~ 2017. 11 . 30 .	
작 품 명	I can fly	
작품소개	가상현실을 통해 스카이다이빙을 간접적으로 체험할 수 있는 프로그램임 다양한 장비를 활용하여 능동적으로 프로그램에 참여할 수 있게 하였음	
주제영역	<input type="checkbox"/> 건강 <input type="checkbox"/> 생산성 <input type="checkbox"/> 생활 <input type="checkbox"/> 안전 <input checked="" type="checkbox"/> 엔터테인먼트 <input type="checkbox"/> 기타 ( )	
타 대회참가 신청수상여부	<input checked="" type="checkbox"/> 미참가 <input type="checkbox"/> 참가신청 중 <input type="checkbox"/> 수상	

## 팀 정보

팀 명	증강현실로 즐기는 SKY JUMP / 패러글라이딩				
팀 원	이 름	소 속	부서/학과	직위/학년	
멘 토	김태은	FICO	Prefessional Service	부장	
지도교수					
멘 티 (참여학생)	멘티 1(팀장)	김지현	건국대학교	컴퓨터공학과	3
	멘티 2	박준성	건국대학교	컴퓨터공학과	4
	멘티 3	임성준	건국대학교	컴퓨터공학과	4
	멘티 4	정명훈	건국대학교	컴퓨터공학과	4
	멘티 5	정자경	건국대학교	컴퓨터공학과	4

본인은 「한이음 공모전 2017」의 제반규정 및 유의사항을 준수하고 제출된 서류의 모든 내용에 허위 사실이 없음을 서약합니다. 또한, 공모전 심사를 위한 평가에 성실히 응할 것이며 참가자와 관련된 정보 활용에 동의합니다.

### ★ 개인정보 수집·이용(개인정보보호법 제15조)

- \* [수집·이용목적] 한이음 공모전 및 한이음 엑스포 행사 운영/관리, 특허출원을 위한 선행기술조사, 언론홍보 및 행사안내, 한이음 사이트 등 사업 관련 자료 공개 및 홍보자료 활용
- \* [수집항목] 이름, 소속, 부서/학과, 직위/학년, 전화번호, 이메일, 프로젝트 수행내용
- \* [보유·이용기간] 사업 종료 후 5년

선정된 작품은 「한이음 엑스포 2017」 행사에 반드시 전시해야 하며, 전시 불참 시 평가 및 수상에서 제외됨을 확인하고 본 공모전에 참가 신청합니다.

본 참가신청서 제출 시, 위 모든 사항에 동의한 것으로 간주합니다.

2017년 9 월 3 일

[붙임] 개발보고서 1부

# 한이음 공모전 2017

## 개 발 보 고 서

2017. 9.

프로젝트명	국문	증강현실로 즐기는 SKY JUMP / 패러글라이딩 (부제: 가상현실로 즐기는 스카이다이빙)
	영문	Augmented reality Sky Jump / Paragliding (Virtual Reality Skydiving)
작 품 명	I can fly	
신 청 자	건국대학교 / 김지현	

# 요 약 본

## 팀 정보

팀 명	I Can Fly			
팀 원	이 름	소 속	부서/학과	직위/학년
멘 토	김태은	FICO	Professional Service	부장
지도교수				
멘티 1(팀장)	김지현	건국대학교	컴퓨터공학과	3
멘티 2	박준성	건국대학교	컴퓨터공학과	4
멘티 3	임성준	건국대학교	컴퓨터공학과	4
멘티 4	정명훈	건국대학교	컴퓨터공학과	4
멘티 5	정자경	건국대학교	컴퓨터공학과	4



작품 정보		
프로젝트명	국문	증강현실로 즐기는 Sky Jump / 패러글라이딩 (부제: 가상현실로 즐기는 스카이다이빙)
	영문	Augmented reality Sky Jump / Paragliding (Virtual Reality Skydiving )
작품명	I Can Fly	
작품 소개	<ul style="list-style-type: none"> <li>· 본 프로젝트는 VR Device와 Kinect Sensor를 이용하여 누구나 스카이다이빙을 간접 체험할 수 있는 기회를 제공함</li> <li>· VR Device를 통해 가상현실을 제공함으로써 실제 스카이다이빙시 발생하는 날씨, 지형, 가격, 안전상의 문제들을 보완함</li> <li>· Kinect Sensor를 이용하여 사용자의 움직임으로 작동하게 만들어 현실감을 극대화하며 운동 효과까지 주고자 함</li> </ul>	
작품 구성도	<pre> graph TD     User((User)) -- "모션정보 수집" --&gt; Kinect[Kinect Sensor]     Kinect -- "서버에게 모션 정보 전달" --&gt; Server[Server]     Server -- "클라이언트에게 모션정보 전달" --&gt; Client[Client]     Client -- "VR콘텐츠로 화면 출력" --&gt; VR[VR Headset]     VR -- "VR환경 제공" --&gt; User </pre>	
작품의 개발배경 및 필요성	<ul style="list-style-type: none"> <li>· 가상현실을 통해 일반적으로 쉽게 할 수 없는 스카이다이빙을 간접적으로 체험해보고자 하였음</li> <li>· 가상현실에 관련된 지식을 습득하고 이를 기반으로 프로그램을 기획하고 개발해보고자 하였음</li> </ul>	
작품의 특징점	<ul style="list-style-type: none"> <li>· 다양한 장비를 활용하여 능동적으로 프로그램에 참여할 수 있게 하였음</li> <li>· Kinect Sensor와 VR Device를 활용하여 프로그램 내에 신체의 움직임 또는 시선의 이동을 반영할 수 있음</li> </ul>	
작품 기능	<ul style="list-style-type: none"> <li>· 시선의 이동을 통해 버튼을 선택할 수 있음</li> <li>· 영상을 통해 프로그램을 사용하는 방법을 배울 수 있게 함</li> <li>· 버튼을 선택하거나 게임을 종료하여 다른 Scene으로 이동할 수 있음</li> <li>· 프로그램 내의 낙하속도, 고도, 랜덤으로 정해진 바람의 방향과 속도, 게임이 진행된 시간을 화면에 표시함</li> <li>· 낙하산을 펴야하는 고도에 도달하거나, 정해진 구역을 벗어날 경우 메시지를 전달 함</li> <li>· 사용자의 움직임에 따라 프로그램 내의 플레이어를 회전 또는 이동시킴</li> <li>· 사용자가 지정한 모션을 취할 경우 낙하산을 펼 수 있게 함</li> </ul>	
작품의 기대효과 및 활용분야	<ul style="list-style-type: none"> <li>· 일반인들이 스카이다이빙을 쉽게 체험해 볼 수 있게 함</li> <li>· 스카이다이빙 전 이수해야하는 안전교육의 효과를 높일 수 있음</li> <li>· 스카이다이빙 자격증을 따기 위한 연습용으로 활용 가능</li> <li>· 군사 훈련용으로 활용 가능</li> </ul>	

# 본 문

## I. 작품 개요

※ 평가항목 : 기획력 (필요성, 차별성)

### 1. 작품 소개

#### ○ 기획의도

- 본 프로젝트는 VR Device와 Kinect Sensor를 이용하여 누구나 스카이다이빙을 간접 체험할 수 있는 기회를 제공함
- VR Device를 통해 가상현실을 제공함으로써 실제 스카이다이빙시 발생하는 날씨, 지형, 가격, 안전상의 문제들을 보완함
- Kinect Sensor를 이용하여 사용자의 움직임으로 작동하게 만들어 현실감을 극대화하며 운동 효과까지 주고자 함

#### ○ 작품내용

- VR Device의 헤드모션컨트롤 기능을 통해 360도 회전기능 및 가상현실을 제공함
- Kinect Sensor를 이용하여 사용자의 관절 값을 받아 회전 및 속도 등을 제어할 수 있음
- 스카이다이빙시 구름과 바람의 변화 같은 여러 특수효과들로 생동감과 오락적 요소를 추가하였음

### 2. 작품의 개발 배경 및 필요성

#### ○ 제작 동기: 가상현실을 통해 일반적으로 쉽게 할 수 없는 경험을 간접적으로 체험해보고자 하였음

- 가상현실은 객체와 배경 모두 가상의 이미지를 사용하기 때문에 현실에서의 제약을 많이 극복할 수 있음
- 이를 이용하여 스카이다이빙 체험장에 가지 않고도 스카이다이빙을 체험할 수 있는 프로그램을 개발하고자 하였음

#### ○ 제작 목적: 가상현실에 관련된 지식을 습득하고 이를 기반으로 프로그램을 기획하고 개발해보고자 하였음

- 가상현실의 개념, 증강현실과 가상현실의 차이점, 가상현실을 구현하는 기술, 국내외 활용사례, 가상현실 시장 현황 및 전망, 국내외 기술개발 동향 등을 조사하며 가상현실의 활용방안을 모색하고자 하였음
- 가상현실에 대한 여러 논문을 접해보며 가상현실에 대한 지식을 습득하고 이를 기반으로 프로그램을 기획하고자 하였음
- 가상현실관련 기기들과 무료 게임엔진인 Unity를 활용하여 프로그램을 작성해보고자 하였음
- Kinect Sensor의 모션캡처 기능을 활용하여 사용자의 움직임을 프로그램 내에 반영해보고자 하였음

### 3. 작품의 특징 및 장점

○ 다양한 장비를 활용하여 능동적으로 프로그램에 참여할 수 있게 하였음

- 프로그램 내에 자신의 움직임을 반영할 수 있음

수동적으로 프로그램을 시청하는 것이 아니라 Kinect Sensor로 모션캡처 기능을 구현하여 능동적으로 프로그램 내에 자신의 움직임을 반영하며 프로그램에 참여할 수 있음.

- 시선의 이동을 통해 프로그램을 조작할 수 있음

VR Device를 통해 프로그램을 출력하여 프로그램에 생동감을 부여하였을 뿐만 아니라 헤드모션컨트롤 기능을 구현하여 시선의 이동을 통해 프로그램을 조작할 수 있어 프로그램을 조작하는 동안 손발이 자유로워질 수 있음.

- 실제 스카이다이빙을 하는듯한 느낌이 들게 하는 장비를 착용함

해먹을 이용하여 프로그램을 사용하는 동안 실제로 하늘을 나는 듯한 느낌을 받을 수 있음.

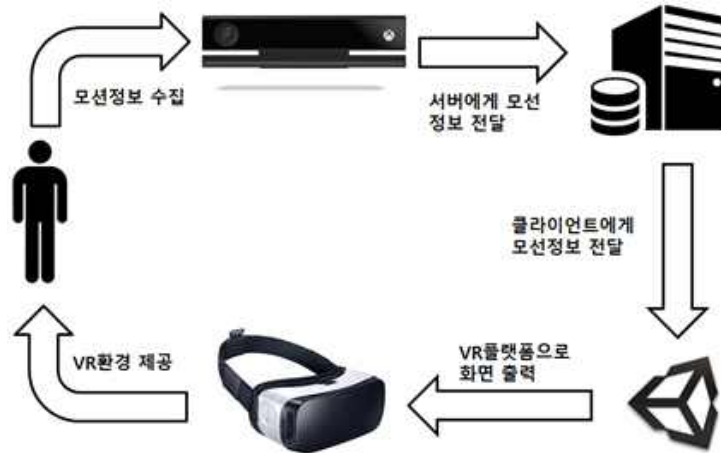
- 일반 스카이다이빙에 오락적 요소를 추가함

낮선 스카이다이빙에 다양한 이벤트를 추가하여 누구나 쉽고 재밌게 즐길 수 있게 함.

## II. 작품 내용

※ 평가항목 : 기술력 (기능구체성, 난이도, 완성도)

### 1. 작품 구성도



## 2. 작품 기능

### 2-1. 전체 기능 목록



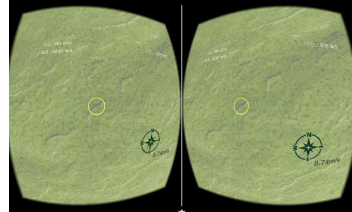

구분	기능	설명	현재진척도(%)
S/W	헤드모션컨트롤 기능 구현	VR Device를 통한 시선 이동	100%
	시작 동영상 표시	프로그램 시작 시 동영상 표시	100%
	버튼 선택	시선의 이동을 통해 버튼 선택	100%
	Scene 이동	버튼 선택 시 원하는 Scene으로 이동	100%
	낙하 속도 표시	변화되는 속도 표시	100%
	고도 표시	변화되는 고도 표시	100%
	풍향, 풍속 표시	현재 풍향과 풍속을 표시	100%
	풍향, 풍속 변경	임의의 값으로 풍향과 풍속 변경	100%

	게임 시간 표시	게임 플레이 시간 표시	100%
	낙하산 펴는 시점 설정	800m에서 자동으로 낙하산이 펴짐	100%
	정해진 코스를 따라 낙하 시 추가 점수 획득	링(정해진 코스) 통과 시 점수 획득	100%
	맵 이탈 경고	정해진 구역을 벗어날 경우 경고, 종료	100%
	결과 점수 표시	게임 Scene에서 받아온 값을 통해 최종 점수 표시	100%
	플레이어 움직임	Kinect를 이용해 플레이어를 움직임	100%
	스카이다이빙 시작	Kinect를 이용해 스카이다이빙 시작하기	100%
	낙하산 펴기	Kinect를 이용해 낙하산 펴기	100%
	설정 선택	변경하고자 하는 설정을 선택	100%
	모드 변경	설정을 통한 모드 변경	100%
	낙하산, 고도 설정 변경	원하는 낙하산과 고도를 선택	100%
	사용 설명 선택	보고자 하는 설명을 선택	100%
	사용 설명 동영상 재생/정지	설명 선택 시 동영상을 재생하거나 정지	100%
	시작 화면으로 이동	각 화면에서 처음 화면으로 돌아오기	100%
H/W	Kinect	사용자의 움직임을 파악	구매H/W
	VR Device	사용자의 시선 움직임을 파악하고 프로그램의 화면을 출력	구매H/W

## 2-2. S/W 주요 기능

기능	설명	작품실물사진
헤드모션컨트롤 기능 구현	VR Device와 연결된 Android의 시선 위치 값을 이용해 화면을 회전시키고 버튼 선택할 수 있음	
버튼 선택	버튼에 시선이 닿으면 사용자가 버튼을 선택하고 있다는 것을 인지할 수 있도록 Bar가 차오른다. 다 찰 경우 지정된 기능을 수행함	



Scene 이동	시작 화면에서 버튼을 선택 시 해당 Scene으로 이동할 수 있음	
낙하 속도 표시	게임 Scene에서 낙하 시 가속도에 의해 변화되는 낙하속도를 표시함	
고도 표시	게임 Scene에서 낙하 시 변화되는 고도를 표시함	
풍향, 풍속 표시	임의로 바람 방향과 바람 속도를 표시 함	
풍향, 풍속 변경	시작 시 풍향, 풍속에 임의의 값을 부여하고 일정 시간마다 값이 미세하게 변경되고 플레이어의 움직임에 영향을 주게 함	
게임 시간 표시	게임 실행 후 흘러간 시간을 표시함	
고도 경고	낙하산을 펴야하는 고도까지 오면 글자의 색이 빨간색으로 바뀌어 경고 메시지를 전달함	
정해진 코스를 따라 낙하 시 추가 점수 획득	링(정해진 코스)의 안쪽을 통과할 시 점수를 획득함	
맵 이탈 경고	정해진 구역을 벗어날 경우 2번의 경고 부여 경고 후에도 벗어나지 않을 경우 종료되게 하였음	

결과 점수 표시	게임 종료 후 자동으로 결과화면으로 넘어가게 함. 이때, 게임 Scene에서 받은 최종 점수를 보여줌	
플레이어 움직임	Kinect를 이용해 사용자의 관절 값을 받아 플레이어 캐릭터를 회전 또는 이동시킴	
스카이다이빙 시작	사용자가 지정된 모션을 취할 시 Kinect를 통해 그 값을 받아 비행기에서 뛰어 내리며 스카이다이빙을 시작하게 함	
낙하산 펴기	사용자가 지정된 모션을 취할 시 Kinect를 통해 그 값을 받아 낙하산을 펴는 기능을 수행하게 함	
설정 선택	버튼을 통해 맵, 낙하산, 고도 중 변경하고자 하는 것을 선택할 수 있게함	
맵 설정 변경	좌우 버튼을 통해 맵을 선택할 수 있음 맵을 변경하면 게임의 모드도 변경되게 함	

낙하산, 고도 설정 변경	좌우 버튼을 통해 낙하산, 고도의 원하는 설정 값을 선택할 수 있게 함	
사용 설명 선택	방향조절, 낙하산 펴기, 시작방법 중 설명을 보고자하는 것을 선택할 수 있게 함	
사용 설명 동영상 재생/정지	방향조절, 낙하산 펴기, 시작방법을 배울 수 있는 동영상을 재생, 정지할 수 있게 함	
시작 화면으로 이동	결과, 설정, 사용 설명 화면에서 처음으로 버튼을 통해 이동할 수 있게 함	

### 2-3. H/W 주요 기능

기능/부품	설명	작품실물사진
Kinect	<ul style="list-style-type: none"> <li>○ 사용자의 움직임을 PC에 전송 <ul style="list-style-type: none"> <li>- Kinect Sensor를 통해 사용자와 Sensor간의 거리를 측정하고 자세를 인식하여 PC에 전송함</li> <li>- 사용자의 움직임에 따라 프로그램내의 캐릭터를 조작할 수 있어 사용자가 캐릭터와 하나가 된 듯 실감나는 체험을 할 수 있음</li> </ul> </li> </ul>	
VR Device	<ul style="list-style-type: none"> <li>○ 프로그램을 출력하는 기능 <ul style="list-style-type: none"> <li>- 머리에 착용하여 모니터대신 화면을 출력해주는 HMD(Head Mounted Display)</li> <li>- 사용자의 머리움직임에 연동되어 화면이 움직이기 때문에 보다 생동감 있는 체험을 할 수 있도록 도와줌</li> </ul> </li> <li>○ 헤드모션컨트롤 기능 <ul style="list-style-type: none"> <li>- 머리의 움직임에 따라 포인터가 반응하는 헤드모션컨트롤기능을 추가했으며 포인터에 눈의 초점을 맞추면 메뉴가 클릭됨</li> </ul> </li> </ul>	

### 3. 주요 적용 기술

#### ○ Unity5

- Unity는 게임개발을 위해 제작된 엔진임
- 화면전환, 오브젝트 생성, 이동, 회전, 각종 물리 연산등 게임개발에 필요한 각종 기능들을 지원해줌
- 본 프로젝트에서는 카메라이동, 소리 재생, 오브젝트 생성, 이동, 가속 및 감속, 회전 및 충돌 감지 기능을 사용해서 기초적인 게임 구현을 하였음

#### ○ VR

- VR은 사용자가 마치 실제 주변 상황·환경과 상호작용을 하고 있는 것처럼 만드는 기술
- 사용자가 보는 화면을 양쪽 눈에 분할하여 보이게 하며, 360도 회전 기능을 제공함
- Unity에서 제공하는 Raycast를 이용하여 사용자의 시선으로 버튼을 클릭 할 수 있게 해줌
- 저렴한 ‘VR BOSS’ 기기를 사용하여 보다 많은 사용자들이 이용할 수 있게 함

#### ○ Kinect

- 마이크로소프트사에서 XBOX360 게임기 주변기기용으로 개발된 모션캡처 디바이스
- 최대 6명의 사용자를 인식해서 골격 및 관절의 위치, 거리, 얼굴 표정, 손동작 등을 측정할 수 있음
- 본 프로젝트에서는 관절의 위치 및 거리 측정, 제스처 기능을 활용함
- 관절의 위치, 거리측정으로 플레이어의 자세제어, 위치제어 및 낙하산 뿔 때의 플레이어의 회전제어 기능을 구현. 제스처 기능으로 낙하산의 개산 기능을 구현

#### ○ Unity Network

- 모바일 디바이스를 활용한 VR기능을 구현함에 따라 Windows 환경에서만 지원되는 Kinect 장비를 사용해서 VR환경을 구축하기에는 제약사항이 많음
- 이를 극복하기 위해 유니티 엔진에서 제공해주는 네트워크 기능을 활용하여 Kinect 장비에서 산출되는 정보들을 OS가 다른 기종으로 전송하는 기능을 사용함  
이기종지원이 사용 가능해짐

### 4. 작품 개발 환경

구분		상세내용
S/W 개발환경	OS	Window10, Android(최소 4.4 'KitKat'이상)
	개발환경(IDE)	MonoDevelop, Visual Studio 2015
	개발도구	Unity 5
	개발언어	C#
	기타사항	
H/W 구성장비	디바이스	VR Boss 또는 CardBoard, Android 휴대폰
	센서	Kinect V2
	통신	무선공유기
	개발언어	C#
	기타사항	
프로젝트 관리환경	형상관리	프로젝트 수정 후에는 반드시 다른 이름으로 저장 후 프로젝트명, 마지막수정일, 작성자명으로 이름변경
	이슈관리	매 회의 진행시 회의록 작성 후 카페에 업로드
	의사소통관리	1주일에 1회 이상 오프라인 회의 진행
	기타사항	

### III. 프로젝트 수행 내용

※ 평가항목 : 수행능력 (문제해결능력, 수행충실성)

#### 1. 멘티(참여학생) 업무분장

번호	이름	대학	학과	학년	역할	담당업무
1	김지현	건국대학교	컴퓨터공학과	3학년	팀장	총괄, 문서 작성, unity관련 기능구현
2	박준성	건국대학교	컴퓨터공학과	4학년	팀원	헤드 모션 컨트롤 기능 구현, unity관련 기능구현, 문서작성
3	정명훈	건국대학교	컴퓨터공학과	4학년	팀원	헤드 모션 컨트롤 기능 구현, unity관련 기능 구현, 문서작성
4	임성준	건국대학교	컴퓨터공학과	4학년	팀원	unity, kinect 방향전환 기능 구현, kinect 서버연결, 문서작성
5	정자경	건국대학교	컴퓨터공학과	4학년	팀원	unity, kinect gesture 기능 구현, kinect 서버연결, 문서작성

## 2. 프로젝트 수행일정

프로젝트 기간 (한이음 사이트 기준)		2017.03.27. ~ 2017.11.30.											
구분	추진내용	프로젝트 기간											
		1월	2월	3월	4월	5월	6월	7월	8월	9월	10월	11월	12월
계획	프로그램 기획 및 수행 계획												
분석	기능 정의												
설계	UI 설계				→								
	프로그램 플로우차트 작성												
개발	모션캡처기능 구현												
	헤드모션컨트롤기능 구현												
	Unity를 통한 화면 구성 및 효과 삽입									→			
테스트	프로그램 테스트 및 수정												
종료	종료 보고												

→ : 변경된 일정

## 3. 프로젝트 추진 과정에서의 문제점 및 해결방안

### 3-1. 프로젝트 관리 측면

- 문제점 : 사용하려던 배경의 용량이 커 다운로드가 진행되지 않아 계획된 일정이 뒤로 미루어짐
  - 프로그램의 배경을 실제와 가까운 지형을 사용하려 하였으나 배경의 용량에 비해 PC 성능이 좋지 않아 배경을 만드는데 문제가 발생함
- 해결방안 : 변경할 배경을 찾는 동안 다른 계획을 먼저 수행하였음
  
- 문제점: VR과 Kinect 연동의 어려움
  - 해외구매가 지원 되지 않아 VR과 Kinect를 자체적으로 연결 할 수 있는 기능을 가진 ‘Oculus Rift’ 를 구매 할 수 없었음
- 해결방안 : 국내 기기인 ‘Kinect v2 VR Examples’ 를 구매해 서버를 이용해 VR Device의 핸드폰과 연동하여 해결함
  
- 문제점 : 하나뿐인 ‘Kinect Xbox’ Sensor의 사용문제
  - 프로젝트의 많은 부분이 ‘Kinect Xbox’ 모션캡처 Sensor가 필요하지만 20만원

이라는 비싼 가격에 하나만 구매 할 수밖에 없어 보관과 사용 문제 발생함

- 해결방안 : 프로젝트를 크게 VR과 Kinect 부분으로 나누고 Kinect 부분을 사는 곳이 가까운 팀원 두 명이 맡아 공동 개발하고 합치는 방식으로 진행해 해결하였음

- 문제점 : 장비(해먹) 보관의 어려움.

- 프로그램을 좀 더 리얼하게 실행하기 위해 필요한 해먹의 부피가 커서 보관과 이동의 불편함을 겪음

- 해결방안 : 대부분 해먹을 보관하는 친구의 집근처에서 만나며, 조립식으로 만들어 이동의 불편함을 최소화 함

### 3-2. 작품 개발 측면

- 문제점: 스카이다이빙 영상을 재생하고 물리적인 장비개발에 초점을 맞춤

- 현실감을 위해서 화면에 보이는 것보다 실제 스카이다이빙 장비들을 착용하는 것이 중요하다고 생각함

- Unity에서 스카이다이빙 영상을 재생하는 기능만 구현하고 물리적인 장비에 초점을 맞추어 개발을 진행함

- 사용자가 이동할 수 있는 범위와 시야가 촬영된 영상으로 한정되어 오히려 현실감이 떨어지게 됨

- 해결방안: 실제 스카이다이빙 영상을 가상의 배경과 모션캡처 기능으로 대체함

- Unity에 스카이다이빙 영상이 아닌 지형과 하늘을 삽입하고 모션캡처 기능을 추가하여 문제를 해결함

- 프로그램 내에서 사용자가 원하는 방향으로 이동할 수 있게 되어 더욱 현실감을 느낄 수 있게 됨

## 4. 프로젝트를 통해 배우거나 느낀 점

- 끈기를 기르고 깊이 있는 학습을 함

- 하나의 프로젝트를 1년 이상 진행하며 끈기를 기를 수 있었고 가상현실과 Unity에 대해 깊이 있는 학습을 하였음.

- 프로젝트의 진행과정과 프로젝트의 내용을 더욱 확실하게 알 수 있었음.

- 프로젝트에 필요한 장비를 신청하기 위해 가상현실 관련 기기를 조사하고 실습 장비 신청서를 작성하며, 가상현실에 더 많은 관심을 가지게 됨.

○ 협업능력을 신장시킴

- 프로젝트를 함께 프로젝트를 진행해가는 즐거움을 알게 됨.
- 하나의 기능을 구현하기 위해 모든 팀원들이 모여 함께 진행할 때도 있었지만 개발의 효율성을 위해 역할을 분담하기도 함. 역할을 분담했을 때도 자신이 맡은 일을 끝냈음에도 팀원들끼리 서로 도와주며 개발을 진행. 이러한 분위기속에서 협업능력을 기르게 됨.

## IV. 작품의 기대효과 및 활용분야

### ※ 평가항목 : 기획력 (활용가능성)

#### 1. 작품의 기대효과

○ 기대효과

- 스카이다이빙의 대중화

스카이다이빙은 1회 평균 50만원 상당에 보험료 및 장비 대여료까지 포함하면 금전적 부담이 큰 스포츠이다. 하지만 'I Can Fly' Application을 이용하면 30만('I Can Fly' Application 이용료 무료, Kinect Xbox 20만 원, VR Device 10만 원 이하)원 이라는 상대적으로 저렴한 가격으로 계속해서 사용 가능하다. 그리고 시간과 장소, 날씨의 영향을 받지 않아 바쁜 현대인에게 유용하다. 또한 나이, 키, 몸무게 등 제한적 요소가 거의 없어서 누구나 가능하다. 특히 고속공포증이 있거나, 몸이 불편한 환자들도 이용 할 수 있다는 장점이 있다. 그래서 보다 많은 사람들이 스카이다이빙을 즐길 수 있을 것이라 기대된다.

	실제 Sky Diving	I Can Fly
1회 비용	50만원 내외	무료
장비 비용	160만원 내외 [아날로그식 고도 측정기 22만원+고글 5만원+ 스카이다이빙복 23만원 +헬멧 7만원 +장갑 1만원+ 낙하산110만원]	30만원 내외(Kinect 20만원+ Cardboard 혹은 VR Device 5만원 내외)
이용조건	안전 교육자 동반 OR 자격증 소유 130kg 이하	나이 제한 없음, 언제든지 가능
장소	1600m이상의 제한된 장소	센서만 있다면 어디서나



		가능
--	--	----

- 운동 및 건강관리

건강이 중요하다는 것은 누구나 알지만, 힘이 들고 재미가 없다는 이유로 게을리하게 되는 경우가 많다. 하지만 가상현실기술로 시각적 흥미를 돋우고 모션캡처 통한 조작법으로 운동효과를 주어 건강관리에 도움이 될 것으로 기대된다.

- 스카이다이빙 연습 및 오락 제공

일반 맵은 스카이다이빙 자격증을 획득하기 위한 사전 연습 기회를 제공한다.

그리고 전쟁테마 맵은 군사 훈련용으로 사용 가능하다. 두 가지 경우 모두 비용 절감효과는 물론 전쟁테마 맵은 군사력 증강까지 기대 할 수 있다. 게임테마 맵은 재미와 스트레스 해소를 극대화하는 효과를 불러온다.

- VR 가상현실 체험

VR Device를 통해 3D 공간을 현실 세계처럼 바라보며, Kinect 모션캡처를 통해 사용자의 움직임이 게임 속 주인공에 실시간으로 적용되는 색다른 경험을 제공한다.

○ 차별성

- 기존에 출시 되어있는 스카이다이빙 대부분이 화면 터치 게임 방식이다. 하지만 ‘I Can Fly’ Application은 Google VR SDK for Unity 연동을 통해 가상현실을 구현하였다. 이를 통해 사용자의 시선이동이 자유로워 원하는 곳을 바라 볼 수 있다. 또한 Kinect v2 with MS-SDK를 연동하여 플레이어를 기존의 화면 터치로 이동하는 것이 아니라 사용자 모션캡처를 실시간으로 적용시켜 움직이게 하였다. 또한 일반맵과 게임맵, 전쟁맵 총 세 가지 테마로 구성 되어 있어 사용자 목적에 맞는 테마를 선택해 사용할 수 있게 하였다.

## 2. 작품의 활용분야

○ 게임 및 가상현실 체험

- VR은 미래 육성산업중 하나로, 아직 초기 단계이지만 생태계가 구축되면서 빠른 성장이 기대된다. ‘정보통신기술진흥센터’에 따르면 해외시장 대비 게임 콘텐츠 경쟁력은 평균 4.32점(7점 만점)으로 나타났으며 대기업·소기업은 보통, 중기업은 높은 편이라는 응답이 다수를 차지하였다.

그림 III-5 게임 콘텐츠 경쟁력

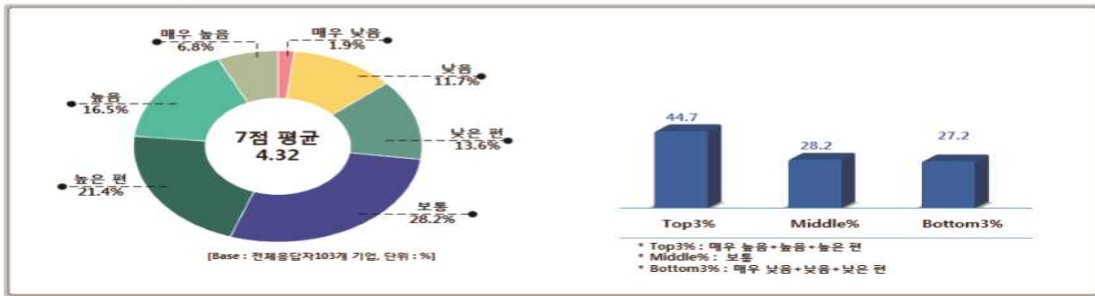


표 III-5 게임 콘텐츠 경쟁력 [단위 : %, 점]

구분	사례수	매우 낮음	낮음	낮은 편	보통	높은 편	높음	매우 높음	Top3%	100점 평균
전체	(103)	1.9	11.7	13.6	28.2	21.4	16.5	6.8	44.7	55.34
산업 분야										
콘텐츠	(76)	1.3	10.5	11.8	32.9	17.1	18.4	7.9	43.4	56.80
플랫폼/솔루션	(10)	10.0	0.0	40.0	20.0	30.0	0.0	0.0	30.0	43.33
디바이스	(15)	0.0	20.0	6.7	13.3	33.3	20.0	6.7	60.0	57.78
기타	(2)	0.0	50.0	0.0	0.0	50.0	0.0	0.0	50.0	41.67
법정 유형										
대기업	(7)	0.0	28.6	0.0	42.9	28.6	0.0	0.0	28.6	45.24
중기업	(12)	0.0	16.7	16.7	16.7	33.3	8.3	8.3	50.0	54.17
소기업	(84)	2.4	9.5	14.3	28.6	19.0	19.0	7.1	45.2	56.35

출처 : 정보통신기술진흥센터

2017년 8월 4일 인천 송도에 국내 최대 규모의 VR 테마파크가 개장하였고, 2017년 7월 28일 삼성서울병원에 환자들 스트레스 해소를 위한 ‘힐링 유 VR’ Application이 출시되었다. 이와 같이 ‘I can fly’ 도 VR 체험장이나 병원, 공공기관에 VR 가상체험과 스트레스 해소용 게임으로 활용 가능 할 것으로 예상된다.



출처 : 인천 송도 트리플 스트리트 ‘Monster VR’



출처 : 삼성서울병원 암병원의 가상현실 체험 앱 ‘힐링 유 VR’

- 스카이다이빙 자격증을 따기 위한 연습용
- 스카이다이빙을 혼자 뛰어 내리기 위해서는 A-Licenece가 필요하다. 하지만 50만원이라는 비싼 가격 때문에 여러 횟수에 거쳐 연습하기가 쉽지 않다. 그러나 ‘I Can Fly’ Application은 모션캡처 센서와 VR Device만 있다면 무료로 가능하다. ‘I Can Fly’ Application은 실제 스카이다이빙 환경과 최대한 유사하게

만들었으며, 바람도 임의로 바뀌며, 사람의 움직임 또한 실제 환경과 같은 물리작용을 적용 하였다. 그래서 운전면허 연습장과 같은 스카이다이빙 연습장에서 활용하면 큰 비용 절감을 가져 올 것이라 기대 된다.

○ 군사 훈련용으로 활용

- 훈련이 곧 실전이다. 이스라엘과, 영국, 미국 등 많은 나라에서 이미 VR을 군사 훈련용으로 사용 하고 있다. 우리나라도 최근 국내 벤처기업들이 개발한 VR 모의 훈련 체계가 우리 군에 적용되고 있다. ‘I Can Fly’ 도 날아오는 총알을 피하며 바닥까지 무사히 착륙하는 공군 군사 훈련용으로 사용이 가능 하다. 특히 훈련병이나 스카이다이빙을 한 번도 해보지 않은 공군에게 유용하게 사용 될 것이다.

○ 실제 스카이다이빙 전 안전교육용으로 활용

- 스카이다이빙은 위험이 따르는 종목이기 때문에 스카이다이빙을 하기 전 반드시 안전교육을 이수해야 한다. 하지만 안전교육은 말로 설명하는 것에 불과하기 때문에 안전 교육 후에 가상현실로 스카이다이빙을 미리 체험을 할 수 있다면 안전교육의 효과를 높일 수 있을 것이다.

○ 스카이다이빙을 할 수 없는 분들에게 스카이다이빙을 체험

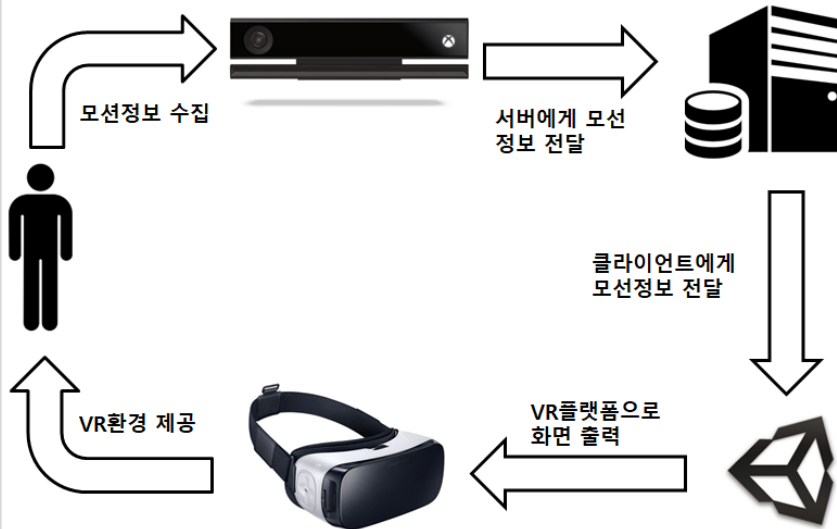
- 모션캡처 Sensor를 이용하여 양쪽 어깨와 팔의 움직임에 대한 기능을 제공함. Sensor와 콘텐츠를 분리하여 거동이 불편한 사람들도 스카이다이빙을 간접적으로 체험 가능

## V. 개발산출물

※ 평가항목 : 평가 전반에 참고

## | 1. 시스템 구성도

### 시스템 구성도



### 설명

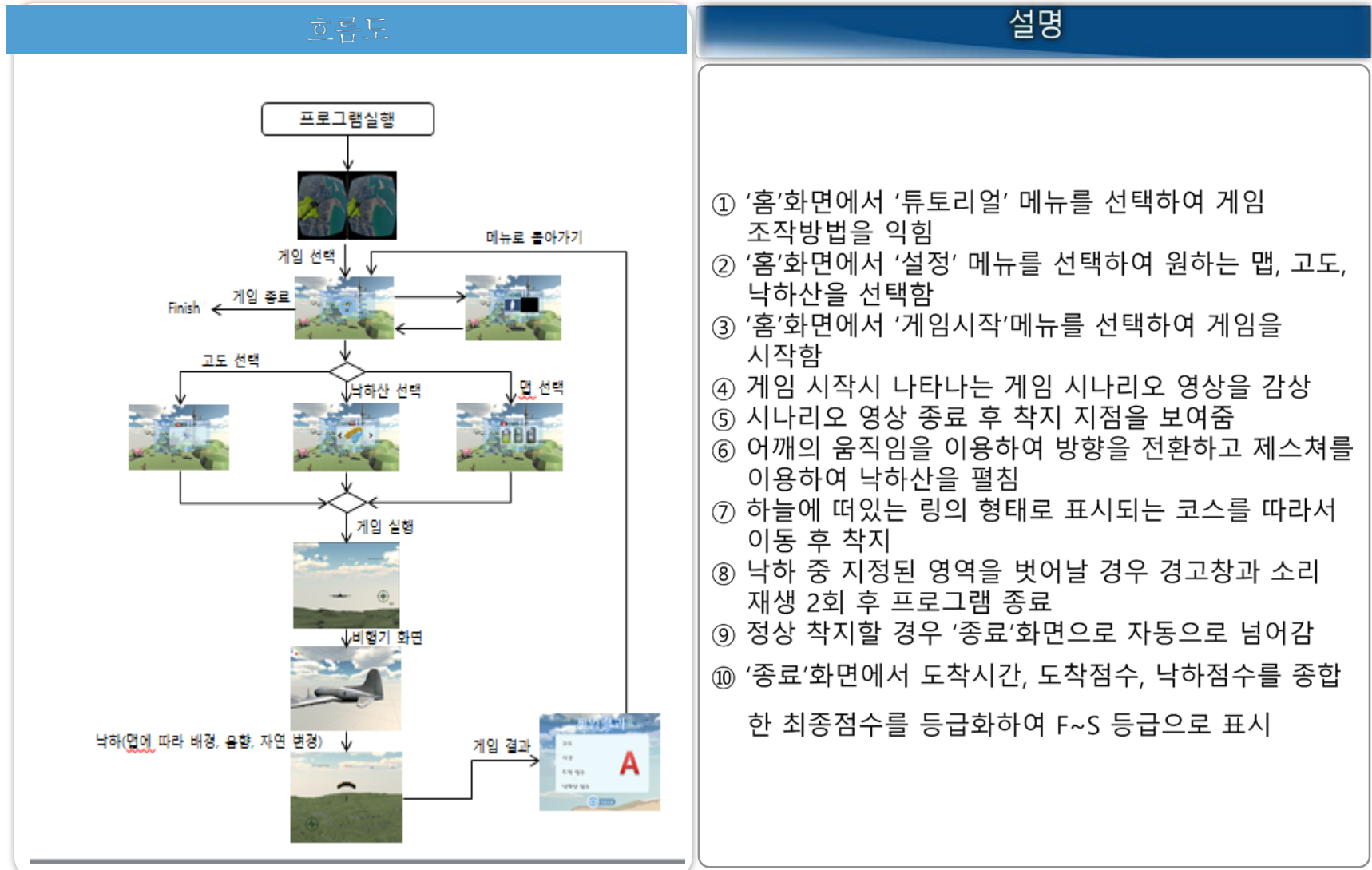
- I Can Fly 프로그램은 I Can Fly App, VR Device, Kinect Sensor 로 구성되어있다.



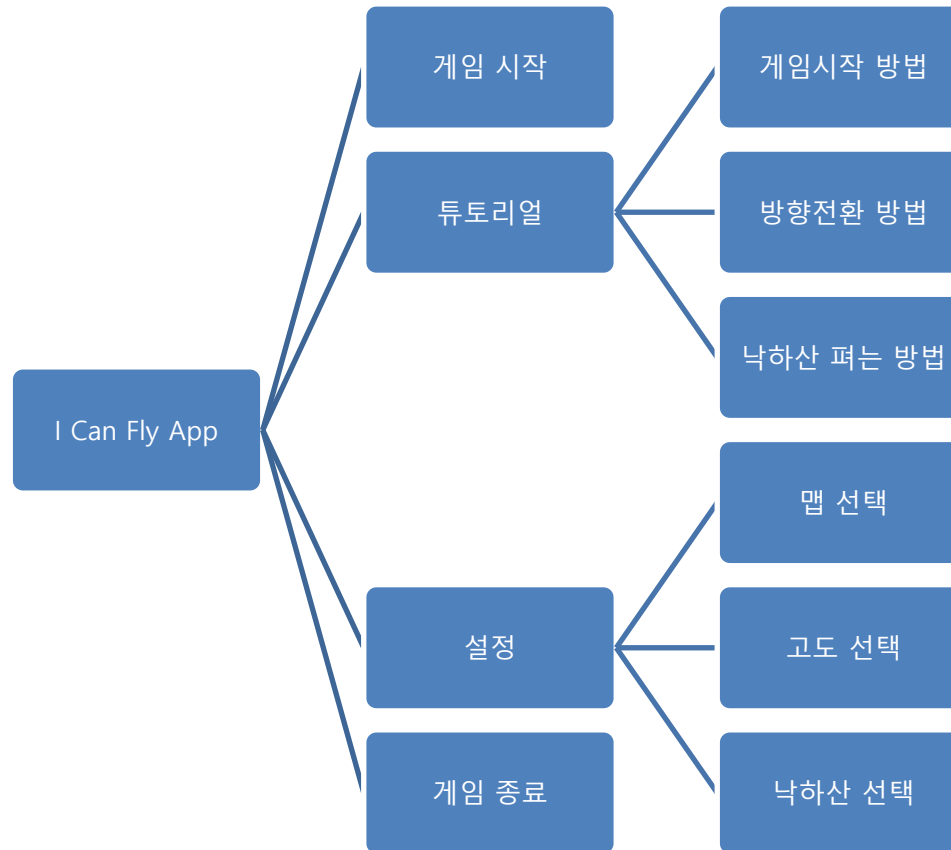
- VR Device를 통해 I can Fly App의 화면을 볼 수 있으며 시선의 이동을 통해 버튼을 선택 할 수 있다.
- 원하는 방향으로 신체를 움직이면 Kinect Sensor를 통해 I Can Fly App에 움직임이 전송되어 프로그램내의 캐릭터에 적용된다.

## | 2. 시스템 흐름도

※ Embedded SW인 경우 HW와 SW 구분하여 작성



### | 3. 메뉴 구성도



## | 4. 프로그램(기능) 목록

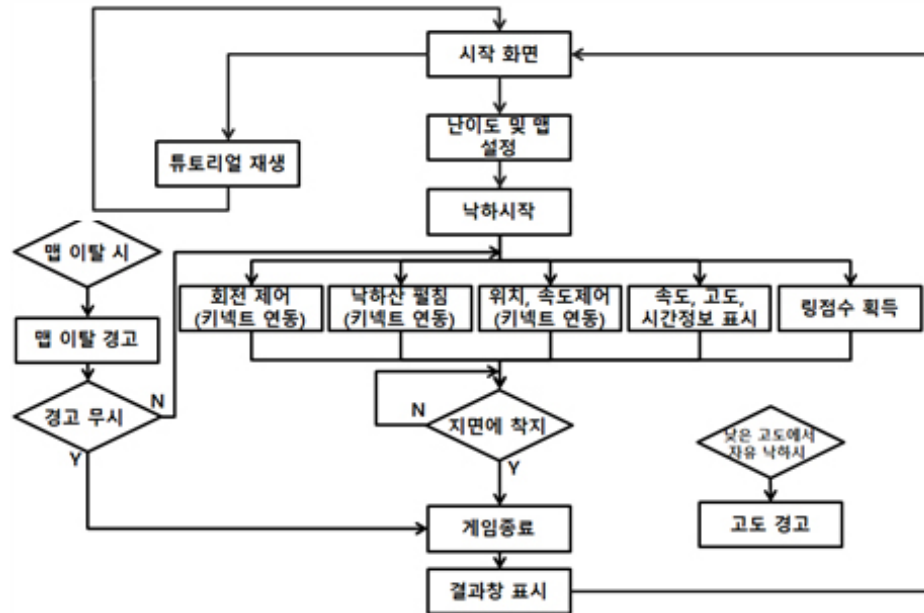
구분	기능	설명
S/W	헤드모션컨트롤 기능 구현	VR Device를 통한 시선 이동
	버튼 선택	시선의 이동을 통해 버튼 선택
	Scene 이동	버튼 선택 시 원하는 Scene으로 이동
	낙하 속도 표시	변화되는 속도 표시
	고도 표시	변화되는 고도 표시
	고도 경고	낙하산을 펴야하는 고도가 됐을 경우 표시
	풍향, 풍속 표시	현재 풍향과 풍속을 표시
	풍향, 풍속 변경	임의의 값으로 풍향과 풍속 변경
	게임 시간 표시	게임 플레이 시간 표시
	정해진 코스를 따라 낙하시 추가 점수 획득	링(정해진 코스) 통과시 점수 획득
	맵 이탈 경고	정해진 구역을 벗어나기 전 경고화면과 소리출력, 정해진 구역을 벗어날 경우 게임 종료
	착지지점 표시	게임을 시작하기 전 착지할 지점을 표시

구분	기능	설명
S/W	캐릭터 이동	키넥트의 모션정보를 바탕으로 캐릭터의 위치 이동
	캐릭터 회전	키넥트의 모션정보를 바탕으로 캐릭터 회전
H/W	Kinect (구매H/W)	사용자의 움직임을 파악
	VR기기 (구매H/W)	사용자의 시선 움직임을 파악하고 프로그램의 화면을 출력

## | 5. 기능 흐름도

실무 산출물 형식

프로그램 ID	I Can Fly	프로그램 명	스카이다이빙 프로그램	작성일	2017. 08. 29	Page	1/1
개요	키넥트기반의 모션인식 기술을 통해 가상현실로 스카이다이빙을 체험하는 콘텐츠를 제공하는 프로그램					작성자	




- 1 I Can Fly APP 을 실행하여 VR Device를 통해 프로그램을 실행하며 시선이동을 통해 버튼을 선택
- 2 '튜토리얼' 메뉴를 통해 조작방법을 익힌 후 '설정' 메뉴를 선택하여 맵, 고도, 낙하산을 선택
- 3 '게임시작' 메뉴를 통해 원하는 맵, 고도, 낙하산으로 게임을 진행




## | 6. 화면 설계서

	기능 번호	ICF00
	기능 명	헤드모션컨트롤 기능 구현
	기능설명	앱의 첫 화면, 프로그램 시작시 영상을 보여주어 사용자에게 긴장감과 흥미를 유발
	처리내용	■가상현실 체험 기존 스카이다이빙 영상을 VR 360도 회전기능을 통해 보여주어 실제 스카이다이빙하는 느낌을 줌
	비고	프로그램을 시작할 때 한번만 실행시킴 프로그램 실행중에는 항상 헤드모션컨트롤 기능이 포함되어있음
	요구사항 명	스카이다이빙 가상현실 체험

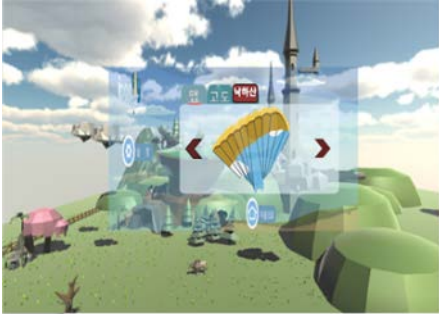
## | 6. 화면 설계서

	기능 번호	ICF01
	기능 명	메뉴 선택
	기능설명	앱의 홈 화면, 게임 시작, 게임 종료, 설정, 튜토리얼 메뉴를 선택할 수 있음
	처리내용	<b>■게임 시작</b> '게임 시작' 버튼을 통해 게임을 시작할 수 있음 <b>■튜토리얼</b> '사용설명' 버튼을 통해 게임 조작방법에 대한 설명을 들을 수 있음 <b>■설정</b> '설정' 버튼을 통해 플레이할 맵, 고도, 낙하산 및 장비를 선택할 수 있음 <b>■종료</b> '종료' 버튼을 통해 프로그램을 종료할 수 있음
	비고	<b>■버튼 클릭 방법</b> 버튼을 1~2초동안 응시하면 버튼의 슬라이더 게이지가 채워지기시작하며 게이지가 가득채워지면 버튼이 선택됨
	요구사항 명	게임시작, 게임종료 및 화면 전환


## | 6. 화면 설계서

	기능 번호	ICF02
	기능 명	튜토리얼
	기능설명	앱의 사용설명 화면, '홈'화면의 '사용설명'버튼을 통해 진입할 수 있음 게임상에서 방향전환 방법, 낙하산 펴는 방법을 영상을 보고 따라해볼 수 있음
	처리내용	<b>■시작 방법 설명</b> '시작방법' 버튼을 선택하면 시작하는 방법과 해당 영상이 나타남 '실행'버튼을 선택하면 해당영상과 함께 자신의 움직임을 볼 수 있음 <b>■방향 전환 방법 설명</b> '방향조절'버튼을 선택하면 방향조절 방법과 해당 영상이 나타남 '실행'버튼을 선택하면 해당영상과 함께 자신의 움직임을 볼 수 있음 <b>■낙하산 펴는 방법 설명</b> '낙하산 펴기'버튼을 선택하면 낙하산 펴는 방법과 해당 영상이 나타남 '실행'버튼을 선택하면 해당영상과 함께 자신의 움직임을 볼 수 있음 <b>■처음화면으로</b> '처음으로'버튼을 선택하면 홈화면으로 돌아감
	비고	
	요구사항 명	게임 조작 방법 설명


## | 6. 화면 설계서

	기능 번호	ICF03
	기능 명	게임 설정
	기능설명	앱의 설정 화면, '홈'화면의 '설정'버튼을 통해 진입할 수 있음 플레이할 맵, 고도, 낙하산을 선택할 수 있음
	처리내용	<b>■맵 선택</b> 일반 맵, 게임 맵, 전쟁 맵으로 구성되어있음 사용자의 목적에 맞는 맵을 선택하여 사용할 수 있음 <b>■고도 선택</b> 사용자가 원하는 고도를 선택하여 플레이할 수 있음 <b>■낙하산 및 장비 선택</b> 사용자가 원하는 낙하산 및 슈트를 선택할 수 있음 <b>■처음화면으로</b> '처음으로'버튼을 선택하면 홈화면으로 돌아감
	비고	
	요구사항 명	게임 설정 및 장비 선택

## | 6. 화면 설계서

	기능 번호	ICF04
	기능 명	게임 시작
	기능설명	사용자가 플레이할 때 보이는 화면이다. 화면에 스카이다이빙시 유용한 플레이어 정보들이 나타남. 게임하는 플레이 환경과 맵에 따른 각종 효과들도 볼 수 있음.
	처리내용	<ul style="list-style-type: none"> <li>■ 낙하속도 표시 중력을 받아 낙하하는 플레이어의 수직 방향의 속도를 나타냄</li> <li>■ 고도 표시 플레이어의 현재 고도를 나타냄</li> <li>■ 풍속 및 풍향 표시 게임을 시작할 때 랜덤으로 정해진 풍속 및 풍향을 나타냄 주기적으로 몇 초 마다 값이 변경됨 플레이어는 풍속 및 풍향의 영향을 받아 움직임.</li> <li>■ 맵 이름 표시 현재 플레이 하고 있는 맵의 정보를 확인 할 수 있음</li> <li>■ 지형 플레이하고 있는 게임 환경을 볼 수 있음</li> </ul>
	비고	화면 UI를 VR 환경에 맞게 설정하고 플레이어 시선에 고정시킴
	요구사항 명	게임플레이 화면

## | 6. 화면 설계서

	기능 번호	ICF05
	기능 명	게임 종료
	기능설명	앱의 종료 화면, 게임 종료 후 각종 결과를 확인할 수 있음
	처리내용	<ul style="list-style-type: none"> <li>■ 고도 표시 사용자가 플레이했던 고도를 보여줌</li> <li>■ 맵 표시 사용자가 플레이했던 맵을 보여줌</li> <li>■ 시간 표시 사용자가 플레이한 시간을 나타냄</li> <li>■ 착지 점수 플레이어가 얼마나 안전하게 도착했는지 점수화하여 보여줌</li> <li>■ 낙하 점수 낙하산을 정해진 고도 내에서 폼는지 점수화하여 보여줌</li> <li>■ 최종 점수 도착시간, 도착점수, 낙하점수를 종합한 최종점수를 등급화하여 보여줌 F~S 등급으로 표시</li> <li>■ 처음화면으로 '처음으로'버튼을 선택하면 홈화면으로 돌아감</li> </ul>
	비고	맵마다 화면이 다름
	요구사항 명	이음 ▶ 프로그램 설계서 게임 결과 화면

## | 8. 프로그램 상세 로직

프로그램 ID	VR Button	프로그램 명	헤드모션컨트롤 기능 구현	작성일	2017. 08. 25	Page	1/3
개요	VR Device의 시선이동으로 Button(Slider)을 클릭					작성자	박준성

### 상세 로직

```

void Update ()
{
    if(triggerEnter)
    {
        progress = progress + Time.deltaTime;
        GetComponent<Slider>().value = progress;

        if(progress >= 1.2f)
        {
            selectEvent.Invoke ();
            OnGazeExit ();
            GameStart ();
        }
    }
}

public void GameStart()
{
    SceneManager.LoadScene ("Game");
}

void OnGazeEnter()
{
    triggerEnter = true;
}

void OnGazeExit()
{
    triggerEnter = false;
    progress = 0f;
    GetComponent<Slider>().value = 0f;
}

```

## | 8. 프로그램 상세 로직

프로그램 ID	Player move Control	프로그램 명	플레이어 방향조절 및 이동	작성일	2017. 08. 25	Page	2/3
개요	키넥트 장비로부터 모션정보를 받아 게임내 플레이어의 방향 조절 및 이동하는 기능					작성자	임성준

### 상세 로직

```

if (((leftShoulderPos.z - rightShoulderPos.z) > 0.1f))
{
    transform.Rotate(Vector3.up, -15 * Time.deltaTime, Space.Self);
}
else if (((rightShoulderPos.z - leftShoulderPos.z) > 0.1f))
{
    transform.Rotate(Vector3.up, 15 * Time.deltaTime, Space.Self);
}
if (((leftShoulderPos.z - rightShoulderPos.z) > 0.1f))
{
    transform.Translate(Vector3.right * Time.deltaTime * 10, Space.Self);
}
if (((rightShoulderPos.z - leftShoulderPos.z) > 0.1f))
{
    transform.Translate(Vector3.left * Time.deltaTime * 10, Space.Self);
}
if (((spineMidPos.z - headPos.z) > 0.1f))
{
    isTilt = true;
    Quaternion target = Quaternion.Euler(140, 0, 0);
    transform.rotation = Quaternion.Slerp(transform.rotation, target, Time.deltaTime * 2.0f);
    fallSpeed = 170;
}

```



## | 8. 프로그램 상세 로직

프로그램 ID	Kinect Gesture	프로그램 명	플레이어 낙하산 제스처	작성일	2017. 08. 27	Page	3/3
개요	팔의 각도와 깊이를 인식하여 범위 안에 팔이 올라가면 낙하산이 퍼지도록 함					작성자	정자경

### 상세 로직

```

if(!gestureListener) return
if(slideChangeWithKeys) {
    if(Input.GetKeyDown(KeyCode.PageDown)) {
        a = true
        currentRotate = transform.rotation;
        futureRotate = Quaternion.Euler(new Vector3(-90, 0, 0));
        futureRotate = currentRotate * futureRotate;
        parachute.active = true
    } else if(Input.GetKeyDown(KeyCode.PageUp)) {
        a = true
        currentRotate = transform.rotation;
        futureRotate = Quaternion.Euler(new Vector3(-90, 0, 0));
        futureRotate = currentRotate * futureRotate;
        parachute.active = true
    }
}
if(slideChangeWithGestures && gestureListener) {
    if(gestureListener.IsSwipeLeft()) {
        a = true
        currentRotate = transform.rotation;
        futureRotate = Quaternion.Euler(new Vector3(-90, 0, 0));
        futureRotate = currentRotate * futureRotate;
        parachute.active = true
    } else if(gestureListener.IsSwipeRight()) {
        a = true
        currentRotate = transform.rotation;
        futureRotate = Quaternion.Euler(new Vector3(-90, 0, 0));
        futureRotate = currentRotate * futureRotate;
    }
}

```

## | 9. 개발 환경 및 설명

구분		항목	적용내역
S/W 개발환경	OS	Windows 10, Android	<ul style="list-style-type: none"> <li>- Kinect센서가 Windows 7이상 지원</li> <li>- Windows 10과 VR Device를 연결할 모바일 기기</li> </ul>
	개발환경(IDE)	MonoDevelop, Visual Studio 2015	<ul style="list-style-type: none"> <li>- Unity 코드 작성 및 개발 도구 사용</li> </ul>
	개발도구	Unity 5	<ul style="list-style-type: none"> <li>- 가상현실 프로그램 개발과 모바일기기와의 연결을 지원하는 개발도구 사용</li> </ul>
	개발언어	C#	<ul style="list-style-type: none"> <li>- 개발 툴이 C#언어 기반이므로 C#언어로 작성</li> </ul>
H/W 구성장비	디바이스	Android 4.4 이상, VR Boss, Card Board	<ul style="list-style-type: none"> <li>- Unity로 개발한 어플리케이션을 구동 및 실험</li> <li>- 가상현실 프로그램을 생동감있게 출력할 장치</li> </ul>
	센서	Kinect V2	<ul style="list-style-type: none"> <li>- 사용자의 움직임 또는 제스처를 관절값을 통해 입력 받아 PC에 전달하는 장치</li> </ul>
	통신	무선공유기	<ul style="list-style-type: none"> <li>- Kinect Sensor를 통해 입력받은 값을 Unity Engine에 전달</li> </ul>
	개발언어	C#언어	<ul style="list-style-type: none"> <li>- 개발 툴이 C#언어 기반이므로 C#언어로 작성</li> </ul>