

최종보고서

Leap Motion을 이용한 가상 악기 연주 및 음악 창작 공유 플랫폼 (Harmony)

Ver. 1.2

2018. 12. 06

한국외국어대학교

정보통신공학과

2팀(ENSEMBLE)

최종보고서: Leap Motion을 이용한 가상 악기 연주 및 음악 창작 공유 플랫폼 (Harmony)

문서 정보

구 분	소 속	성 명	날 짜	서 명
작성자	한국외국어대학교	김세용	2018. 12. 06	
	한국외국어대학교	김준영	2018. 12. 06	
	한국외국어대학교	김진우	2018. 12. 06	
	한국외국어대학교	김정민	2018. 12. 06	
	한국외국어대학교	유진솔	2018. 12. 06	
검토자	한국외국어대학교	김세용	2018. 12. 06	
	한국외국어대학교	김준영	2018. 12. 06	
	한국외국어대학교	김진우	2018. 12. 06	
	한국외국어대학교	김정민	2018. 12. 06	
	한국외국어대학교	유진솔	2018. 12. 06	
사용자				
승인자	한국외국어대학교	홍진표	2018. 12. 06	

머리말

Leap Motion을 이용하여 가상 악기를 연주 할 수 있고 합주 모듈을 이용하여 각종 악기 사운드를 쌓음으로써 합주가 가능 하다. 또한 만든 음악을 공유 할 수 있는 공유 플랫폼을 웹으로 제공한다. 본 문서는 이러한 Harmony 서비스들을 구축하기 위한 시스템들의 소개와 설계 방법을 기술한다.

목 차

목 차	4
1. 개요	6
1.1 목적	6
1.2 범위	6
2. 서비스 개요	7
2.1 서비스 기획 배경	7
2.2 서비스 소개	8
2.2.1 서비스 기능	8
2.2.2 Class Diagram	9
2.2.3 시나리오	10
3. 시스템 구성	12
3.1 전체 시스템 구성도	12
3.2 사용한 소프트웨어	13
3.2.1 Unity	13
3.2.2 Django Web Framework	13
3.2.3 Amazon Elastic Beanstalk	13
3.2.5 MySQL	14
3.3 사용한 하드웨어	14
3.3.1 Leap Motion	14
3.3.2 Arduino & 압력센서(FSR-406)	15
3.3.3 추가 하드웨어	15
4. 시스템 상세설계	16
4.1 HM Software	16
4.1.1 인터페이스	16
4.1.2 합주 모듈	17
4.2 Drum	19
4.2.1 drum stick	19
4.2.2 drum	19
4.2.3 Kick drum	21
4.3 Keyboard	21
4.3.1 keyboard	22
4.3.2 Synthesizer	24
4.3.3 Sustain Pedal	24
4.4 Launch Pad	25

최종보고서: Leap Motion을 이용한 가상 악기 연주 및 음악 창작 공유 플랫폼 (Harmony)

4.4.1 launch pad	25
4.4.2 Loop station	27
.....	27
4.5 HM Web server + Database.....	28
4.5.1 Web Server Login & Logout.....	28
4.5.2 Web Server 회원가입	28
4.5.2 File Up/Download와 게시판 기능 제공'	28
5. 유저 인터페이스 및 가이드	30
6. 적용방안 및 기대효과	33
7. 프로젝트 세부추진계획 및 세부일정	34
8. 팀원 담당업무	35

1. 개요

본 장에서는 Leap motion을 이용한 악기 연주 및 음악 창작과 공유 플랫폼인 Harmony에 대한 목적과 범위, 참고자료, 용어 및 약어를 제시 한다.

1.1 목적

본 프로젝트는 언제 어디서나 가상으로 악기 연주가 가능한 HM software를 소개 및 설계하고 여러 사용자가 새롭게 창작한 음악의 업로드, 다운로드가 가능하도록 하는 HM Web site를 설계한다.

이를 통하여 사용자가 직접 악기를 구매하지 않아도 HM software, leap motion, PC가 있는 환경이라면 언제든지 가상으로 드럼, 건반악기, 런치패드 연주가 가능하고 녹음 기능 또한 제공하여 자신의 연주를 녹음할 수 있는 환경을 구축하는 데 목적을 둔다.

1.2 범위

“Harmony” 서비스는 leap motion, arduino와 압력센서로 구성된 HM kit을 이용하여 가상으로 악기를 연주하고, 음악을 창작하고 공유하는 환경을 제공한다.

본 프로젝트 개발 진행에 있어서 다음과 같은 범위를 둔다.

- Leap motion과 arduino를 이용한 악기 연주
: Leap motion으로 가상의 손을 화면에 띄우고, arduino로 압력 센서 값을 받아와 드럼에서는 킥 드럼의 역할로, 키보드에서는 서스테인 페달의 역할로, 런치 패드에서는 루프 스테이션의 역할로써 가상 악기 연주가 가능하다.
- 재생/정지, 녹음, mute/solo 기능으로 구성된 합주 모듈 제공
: HM software를 실행하여 만든 악기 연주 음원 뿐만 아니라 외부의 어떤 음악 파일과도 mixing이 가능하다.
- HM website에서 음악 공유 기능 제공
: HM website에 업로드 되어 있는 음악들을 다른 사용자와 공유할 수 있다.

2. 서비스 개요

Harmony 서비스는 Leap motion을 이용한 가상 악기를 제공하여 시공간의 제약 없이 연주를 가능하게 하고 그 연주에 대한 음악 창작 및 공유 플랫폼을 제공한다.

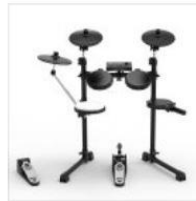
2.1 서비스 기획 배경

충간소음 줄이기, 서로 배려해야

오피니언 | 승인 2018.05.01 | 15면

댓글 0 트위터 페이스북

충간소음은 사람의 직접적인 행동으로 인한 소음과 각종 악기, 음향기기 등을 통해서 나오는 소음을 말한다. 아파트나 빌라 등 공동주택에서 거주하는 사람들이 제일 많이 겪는 황당함을 말한다면 아마 '충간 소음'일 것이다. 이 충간소음 때문에 이웃간에 사이가 멀어지거나 서로 스트레스를 받는 일이 많다고 한다. 물론, 사람들이 많이 모여 사는 곳에 살다보면 이런저런 마찰이 있을 수 있다. 특히, 요즘에는 이웃이 누구인지, 윗층



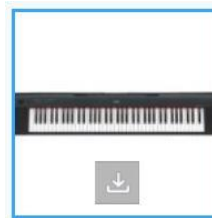
넥타 Hitman HD-4M 전자드럼

399,000원

생활/건강 > 악기 > 타악기 > 전자

우리+네이버페이 간편결제시 5% 할인

리뷰 136 · 등록일 2018.05. ·



야마하 NP-31

최저 322,500원 판매처 18

생활/건강 > 악기 > 건반악기 > 키보드

건반수 : 87이하건반 | 건반종류 : 5만, 10음역, 12W, 32동시발음, 듀얼/

리뷰 ★★★★★ 227 · 등록일 201



노베이션 터치패드 프로 에이블론

최저 317,990원 판매처 36

생활/건강 > 악기 > 건반악기 > 키보드/

DJ컨트롤러, 64버튼

리뷰 ★★★★★ 68 · 등록일 2015.11. ·

[Figure 1] 악기 관련 기사

- 악기를 연주 할 때에 시·공간적 제약에 의한 불편함이 많다.
 - (1) 공동주택지에는 악기 연주로 인한 소음 문제로 이웃에게 피해를 유발하는 경우가 많고, 이는 쉽게 이웃간 갈등으로 이어진다.
 - (2) 악기를 가지고 있지 않은 사람은 악기 대여를 위해 악기가 있는 곳으로 연습하러 가야만 한다.
 - (3) 합주를 할 때, 연습 할 만한 공간이나 시간을 맞추기 힘들다.
- 악기를 구매하기 위해 지불해야 하는 가격이 매우 부담스럽다.

기존의 아날로그 악기를 대체 할 만한 다른 악기와 합주 서비스가 필요하다.

2.2 서비스 소개

2.2.1 서비스 기능



[Figure 2] 하모니 서비스

- 드럼 연주
 - Harmony Software를 사용하는 사용자는 가상의 공간에서 드럼을 연주 할 수 있다.
 - 추가 구성품인 페달(arduino)을 밟으며 킥 드럼도 칠 수 있다.
- 키보드 연주
 - Harmony Software를 사용하는 사용자는 가상의 공간에서 건반악기를 연주 할 수 있다.
 - 건반 악기는 클래식 피아노와 일렉트릭 피아노 사운드를 선택 할 수 있으며 또한 신디사이저를 선택하여 원하는 사운드로 조절 가능하다.
 - 추가 구성품인 페달(arduino)을 밟으며 서스테인 페달을 밟은 것과 같이 리버브 효과를 낼 수 있다.
- 런치 패드 연주
 - Harmony Software를 사용하는 사용자는 가상의 공간에서 런치패드를 연주 할 수 있다.
 - 런치 패드는 각 패드들에 원하는 사운드를 넣을 수 있다.
 - 추가 구성품인 페달(arduino)을 밟으며 루프 스테이션 기능을 사용 할 수 있다.
- 합주
 - 모든 악기의 연주 화면에서는 기본적으로 3개의 트랙을 제공한다.
 - 각각의 트랙에 원하는 악기 사운드를 삽입하여 재생가능 하다.

[Figure 3] Class Diagram

2.2.3 시나리오

Case1. 악기 연주

소프트웨어는 립모션을 통해 사용자의 모션을 인식하고, 가상 환경을 제공하여 사용자의 모션에 의해 악기가 연주되도록 한다. 또한 이를 녹음할 수 있도록 한다. 사용자의 선택에 따라 녹화된 음악 파일을 웹사이트에 업로드할 수 있다.

시나리오 이름	PlayHM
참여 액터	SeYong : Player HMSW : HMSoftware Leap : LeapMotion HMWST : HMWebSite
이벤트 플로우	<ol style="list-style-type: none"> 1. Harmony 이용자인 SeYong은 피아노 연주가 하고 싶어 HMSW를 켜고 손 인식을 잘 하게끔 Leap 위치를 미리 조정했다. 2. HMSW가 켜지고, Leap이 검지 손가락 방향을 인식해 만든 커서로 HMSW상의 모든 화면을 조작하기 때문에 마우스 없이 연주 모드를 클릭 후에 로그인 하였다. 3. 악기 선택 화면이 나타나고, SeYong은 건반 악기만 구매한 상태이므로 건반 악기만 선택할 수 있어서 건반 악기를 선택했다. 4. 연주모드를 선택하고 유니티 화면으로 자신의 연주 모습을 보면서 자유롭게 피아노 연주를 한다. 5. 다른 사운드로 연주해보고 싶은 SeYong은 제공되는 신디사이저 사운드셋으로 사운드를 변경하여 연주를 하다가 맘에 드는 연주를 '녹음'하고 컴퓨터에 저장하였다. 6. 자신의 연주를 업로드 하고 싶어진 SeYong은 HMWSite에 접속하여, '업로드' 버튼을 클릭해 자신의 연주 음원을 업로드하였다.

[Table 1] 시나리오 1

최종보고서: Leap Motion을 이용한 가상 악기 연주 및 음악 창작 공유 플랫폼 (Harmony)

Case2. 연주 병합

사용자는 웹사이트에서 다른 사용자의 녹음된 곡을 다운받을 수 있고, 이를 소프트웨어에서 재생시켜 곡을 덧입혀 연주할 수 있다.

시나리오 이름	<u>Merge Audio</u>
참여 액터	SeYong : Player JinSol: Player HMWST : HMWebSite HMSW : HMSSoftware
이벤트 플로우	<ol style="list-style-type: none"> 1. JinSol은 Harmony 드럼 악기 이용자이다. JinSol은 다른 Player들의 연주곡을 듣고 싶어 HMWebSite에 접속했다. 2. JinSol은 HMWSW에서 제공하는 게시판에 업로드된 Player들의 곡을 듣던 중, SeYong이 업로드한 신디사이저 연주에 자신의 곡을 더하고 싶어졌다. 3. JinSol은 SeYong의 글에서 '다운로드' 버튼을 클릭해 SeYong의 연주곡을 다운로드하고, HMSW의 합주 모듈에 곡을 재생시켰다. 4. 음악 파일에 맞춰 연주연습을 하다가 마음에 드는 연주를 녹음하여 컴퓨터에 저장하였다. 5. 병합된 음악파일은 HMWST에 업로드할 수 있고, 다운로드도 할 수 있어서 JinSol은 음악파일을 다운로드 받아 소장한다.

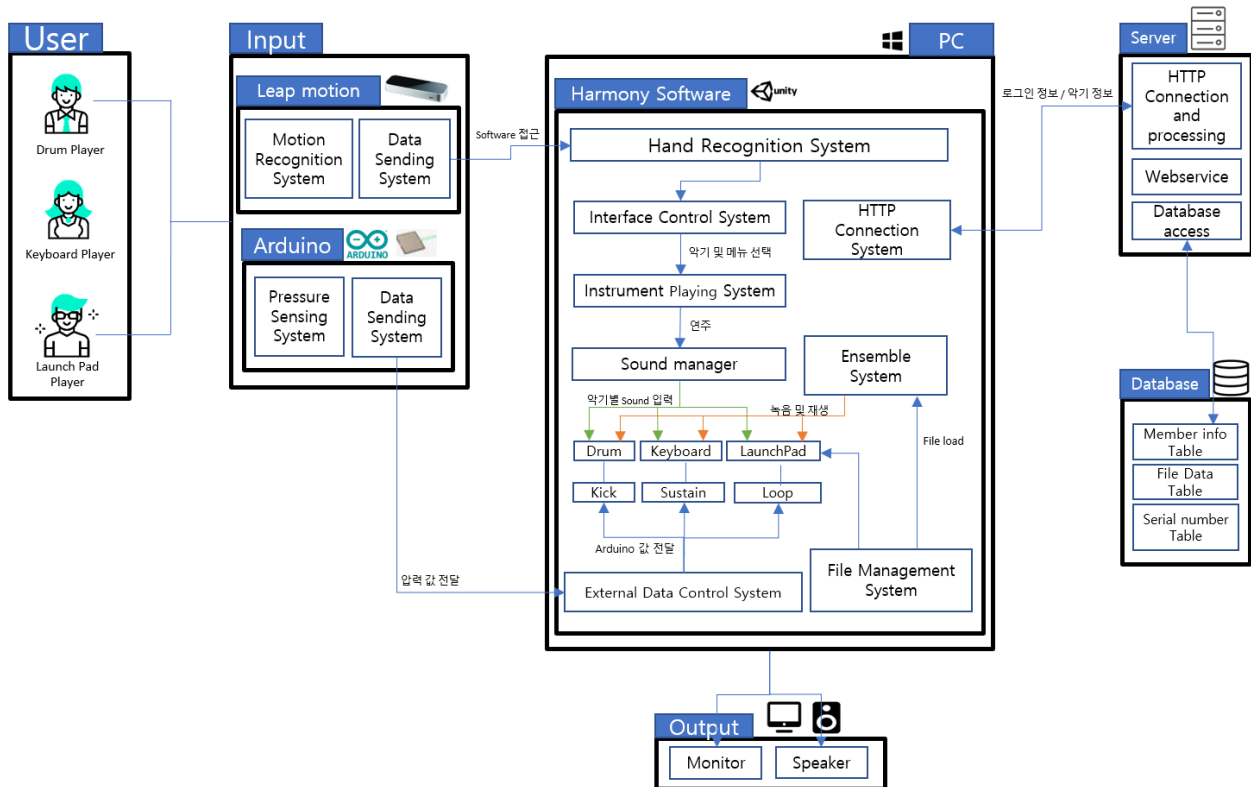
[Table 2] 시나리오 2

최종보고서: Leap Motion을 이용한 가상 악기 연주 및 음악 창작 공유 플랫폼 (Harmony)

3. 시스템 구성

본 장에서는 Leap motion을 이용한 가상 악기 연주 및 음악 창작 공유 플랫폼 Harmony에 대한 전체 시스템 구성 그리고 시스템을 구성하기 위한 개별 소프트웨어, 하드웨어를 기술한다.

3.1 전체 시스템 구성도



[Figure 4] 전체 시스템 구성도

구성 요소	설명
User	User는 Harmony Service를 받는 대상이다.
HM Kit	HM Kit(점선)은 Leap Motion과 Arduino(FSR-406)으로 구성되어 있다.
HM Software	HM Software에서는 가상 악기를 연주하고 녹음 할 수 있으며 다른 곡과 합주를 할 수 있다.
HM Web Server	HM Web Server는 로그인 기능, 음원파일 업/다운로드 기능 등을 통해 음악 공유 플랫폼 서비스를 제공한다.
Database	Database는 HM Web Server와 정보를 주고 받는다.

[Table 3] 전체 시스템 구성도

최종보고서: Leap Motion을 이용한 가상 악기 연주 및 음악 창작 공유 플랫폼 (Harmony)

3.2 사용한 소프트웨어

3.2.1 Unity



[Figure 5] Unity

Unity는 게임 엔진 기술이자 통합개발환경(Integrated Development Environment, IDE)이다. Harmony Software는 C# 스크립팅을 통해 객체들의 속성과 컴포넌트를 조작하고 립모션과 상호작용을 하기위해 Unity기반으로 제작되었다

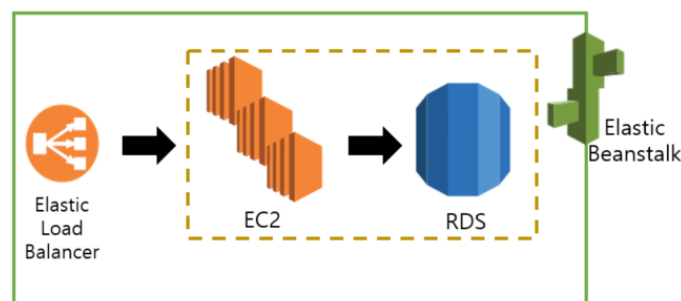
3.2.2 Django Web Framework



[Figure 6] django

Harmony 서비스에서 사용자가 직접 만든 음악을 공유할 수 있는 플랫폼은 Django 웹 애플리케이션 프레임워크를 이용하여 제작 하였다. Django는 파이썬을 사용하며 보다 빠르고 편리하게 Web Server를 설계할 수 있게 해준다.

3.2.3 Amazon Elastic Beanstalk



[Figure 7] Amazon Elastic Beanstalk

Elastic Beanstalk는 웹 앱과 서비스를 배포, 관리 및 증감할 수 있는 서비스이다. Django로 제작된 음악 공유 플랫폼 기능의 Web App을 Elastic Beanstalk를 사용해 배포하였다. 이로써 Local이 아닌 많은 사람들이 자유롭게 Harmony Web Site를 사용할 수 있게하였다.

3.2.5 MySQL



[Figure 8] MySQL Database

관계형 데이터베이스 관리 시스템(RDBMS)인 MySQL은 Harmony 서비스를 제공하는 회원들의 정보와 게시판 정보, 오디오 파일의 정보를 저장하고 관리한다.

3.3 사용한 하드웨어

3.3.1 Leap Motion

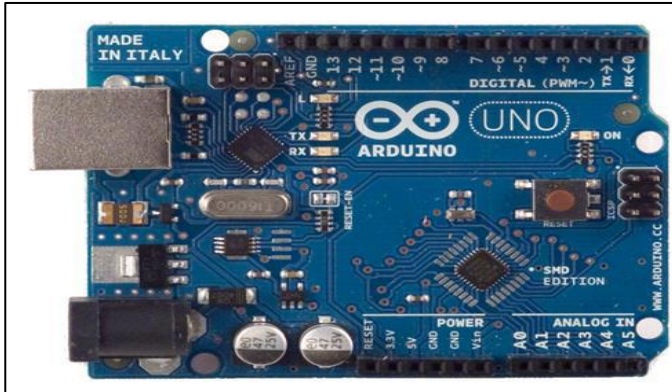


[Figure 9] Leap Motion

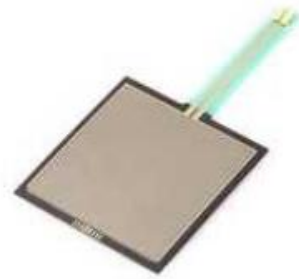
Harmony의 핵심 모션인식 장치이다. 초당 290 프레임, 0.01mm의 움직임까지 감지 및 입력이 가능하며 적외선을 이용해 감지하므로 불빛이 없는 곳에서도 문제없이 모션인식이 가능하다. 손 마디마디를 인식하므로 손을 흔들거나 위 아래로 움직이고 구부리고 움켜쥐는 등의 여러 동작을 자유롭게 표현, 인식이 가능하다.

최종보고서: Leap Motion을 이용한 가상 악기 연주 및 음악 창작 공유 플랫폼 (Harmony)

3.3.2 Arduino & 압력센서(FSR-406)



[Figure 10] Arduino Uno



[Figure 11] FSR-406

Harmony Software에서 악기를 연주할 때 사용하는 페달은 Arduino와 압력센서를 이용하여 제작하였다.

3.3.3 추가 하드웨어



[Figure 12] 립모션 거치대



[Figure 13] 페달 하드웨어

왼쪽 거치대는 Head Mounted 모드를 사용하는 Leapmotion을 고정하는데 사용된다. 오른쪽은 페달으로써 동그란 부분이 부착된 압력 센서와 보다 쉽게 부딪히게끔 해준다.

4. 시스템 상세설계

4.1 HM Software

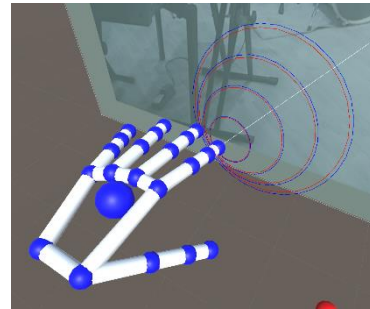
HM Software는 LeapMotion과 Arduino가 연결된 PC에서 실행된다. Software상에서 악기를 선택하여 가상 공간에서 연주를 할 수 있으며 연주한 곡을 녹음하거나, 녹음된 연주 위에 새로운 연주를 얹어 각각의 연주들을 병합해주는 기능을 제공한다. 또한 HM Software는 키보드와 마우스 그리고 Leap Motion을 통해 생성된 가상의 커서(Cursor)로도 메뉴 조작 가능하다.

4.1.1 인터페이스

▶ ‘모션인식을 통한 인터페이스 조작’

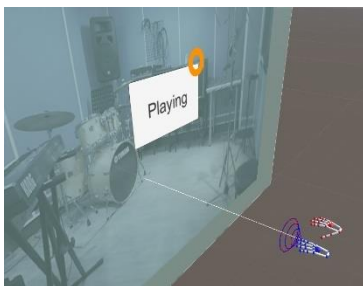


[Figure 14] finger direction(1)

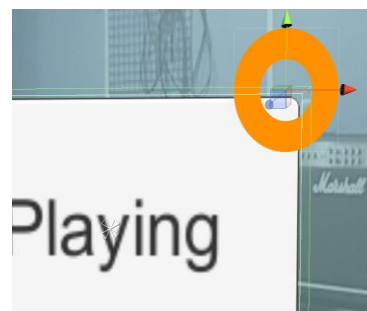


[Figure 15] finger direction(2)

Leap Motion의 Finger Class에서 검지 손가락(Index Finger)의 위치와 방향을 `finger.TipPosition()` 함수와 `selectedDirection(finger.TipPosition())` 함수를 통해 값을 받는다. 이 값은 `ToVector3()` 함수를 통해 벡터 값으로 변환한다. UI를 보여주는 가상의 공간에 위치한 2차원 Canvas 평면과 검지 손가락 끝의 직선 벡터의 교점을 구하여 Cursor를 생성한다.



[Figure 16] finger direction(3)



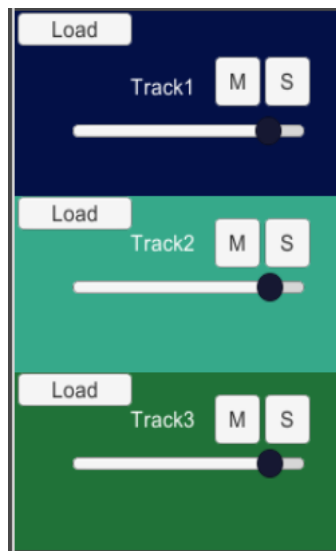
[Figure 17] finger direction(4)

클릭을 할 수 없는 가상의 커서(Cursor)는 `OnCollisionStay()`를 통해 UI BUTTON과 만나있음을 확인하고, 이 때 Timebar를 제공하여 Timeout이 되면 클릭 함수 `OnButtonClick()`를 호출하여 Button을 활성화 한다.

4.1.2 합주 모듈

합주 모듈은 각 악기의 연주화면에 숨어 있다가 연주화면의 좌측 상단에 합주 모듈 버튼을 클릭 할 때 나타난다. 합주 모듈은 사용자가 선택한 외부 음원 파일을 SW에 불러와 Mixer기능을 통해 생성된 트랙에 일대일 Mapping시킨다. 각 트랙을 독립적으로 제어(MUTE기능, SOLO기능, Volume Control기능) 할 수 있는 기능을 가지고, 이를 바탕으로 편리하게 사용자의 연주와 합주 할 수 있게 하고, 녹음 기능을 통해 외부 음원에 사용자의 연주가 덧입혀진 하나의 wav파일을 생성하여 사용자에게 제공한다. 합주 모듈은 각 악기 scene에 각각 독립적으로 제공된다.

▶ ‘사용자가 원하는 음원 파일 load기능 제공’



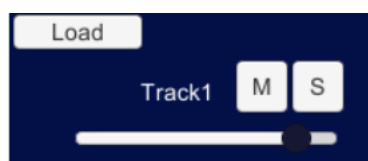
1. 그림과 같이 트랙 별로 Load버튼을 생성한다. Load버튼 클릭 시에 Unity의 OpenFilePanel() 함수를 실행시켜 File Browser를 띄우고, 사용자의 파일 선택에 의해 해당 파일의 Path를 얻어낸다.

2. 얻어낸 파일의 Path를 Unity의 www객체에게 Parameter로 전달하고, www객체는 Path를 바탕으로 파일을 다운로드하여 GetAudioClip()함수를 통해 SW에서 사용할 수 있는 오디오 파일로 불러온다.

3. 불러온 오디오 파일을 해당 트랙의 AudioClip으로 추가하여 오디오를 사용자가 제어할 수 있도록 한다.

[Figure 18] Track 종합

▶ ‘트랙 별 독립적 제어 기능 제공’



[Figure 19] Track 1

트랙 별 독립적 제어를 하기 위해, 기본적으로 3개의 트랙을 Mixer에서 생성한다.

1. Mute 기능

- 그림과 같이 트랙 별로 Mute버튼을 생성한다. Mute 버튼 클릭 시에 해당 트랙의 현재 Volume을 저장하고 Volume을 Unity Mixer상의 최솟값인 -80으로 변경한다.
- Mute 버튼을 한 번 더 클릭 시에 저장했었던 Volume으로 변경한다.

최종보고서: Leap Motion을 이용한 가상 악기 연주 및 음악 창작 공유 플랫폼 (Harmony)

2. Solo 기능

- 그림과 같이 트랙 별로 Solo버튼을 생성한다. Solo 버튼 클릭 시에 나머지 트랙의 현재 Volume을 저장하고 Volume을 Unity Mixer상의 최솟값인 -80으로 변경한다.
- Solo 버튼을 한 번 더 클릭 시에 나머지 트랙의 Volume을 저장했었던 Volume으로 변경한다.

3. Volume Control 기능

- Volume Slider를 생성하고, Mixer에서의 해당 트랙 Volume Fader와 Mapping하여 Slider를 조절하면 트랙의 볼륨이 조절되도록 한다.

▶ '소프트웨어 내부 소리 녹음 기능 제공'



[Figure 20] 합주 모듈 기본

1. Unity Main Camera에 AudioListener는 소프트웨어에서 나는 모든 소리를 출력해주는 역할을 하므로 AudioListener에서 출력되는 사운드를 wav파일로 변환하는 Script를 Component로 추가한다.
2. Script는 OnAudioFilterRead를 통해 오디오 청크(chunk)가 해당 필터에 통과될 때마다 오디오 데이터를 받아들이고 FileStream.write를 이용해 이를 wav파일로 변환하는 함수를 추가한다.

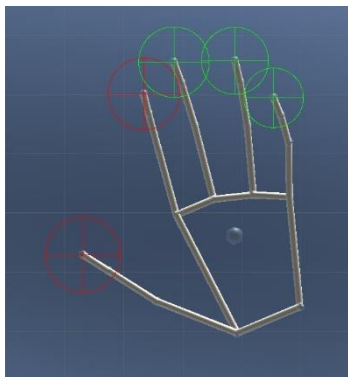
4.2 Drum

HM Software의 드럼은 립 모션 위에서 주먹을 쥐면, 드럼 스틱 객체를 얻는다. HM Software는 얻어진 스틱 객체를 이용하여, 가상환경에 띄워진 드럼 객체들을 연주할 수 있다. HM Software의 드럼은 심벌 3개와 북 4개를 가지며 드럼 스틱으로 내려치는 세기에 따라 악기 볼륨이 조절된다.

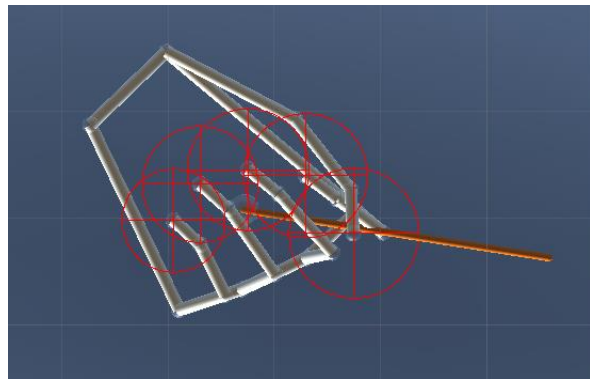
4.2.1 drum stick

사용자는 드럼 악기를 연주하기 위해서는 가상 환경의 손에 드럼 스틱을 쥐어야 한다.

▶ '드럼 스틱 조작'



[Figure 21] 손 객체(손가락 폼 때)



[Figure 22] 손 객체(손가락 접었을 때)

ExtendedFingerDetector 함수를 통해 각 손가락 정보(thumb, index, middle, ring, pinky)들의 상태를 stateArray에 담아 펼쳐진 손가락의 개수를 판단한다. Extended finger의 개수가 0이 되면 주먹을 쥔 것으로 activate하고 가상의 손에 가상 드럼 스틱 객체를 생성하여 주먹에 쥐어준다. 스틱의 생성 좌표를 고정하여, 사용자가 원하는 방향과 위치에 스틱을 생성할 수 있도록 하여 맞춤 연주가 가능하다.

4.2.2 drum

▶ '드럼 악기 연주 제공'

1. 연주 방법

HM Software는 사용자의 손을 립 모션을 통해 인식하여 가상 환경에 가상 손 객체를 생성한다. HM Software는 드럼스틱 객체와 가상환경에 구현된 드럼 객체들의 충돌을 감지하여 사운드를 출력한다. 드럼 스틱과 각 드럼 객체 간의 충돌에 의한 사운드 출력은 드럼 스틱 객체의 rigidbody컴포넌트와 드럼 객체의 mesh collider의 충돌을 감지하여 일으킨다.



[Figure 23] 드럼 연주 화면

2. 악기 구조



[Figure 24] 드럼 연주 화면

HM Software는 다음과 같이 1인칭 시점에서 가상 악기를 보여준다. HM Software는 LeftTom, RightTom, Snare, FloorTom, Hi-hat Cymbal, Crash Cymbal, Ride Cymbal으로 드럼을 구성한다. 악기의 현실성을 높이기 위해 Cymbal들에 물리적 특성 중의 하나인 spring joint를 드럼에 적용한다. 이를 통해 HM Software는 치는 방향과 세기에 따라 다른 움직임을 보여준다.

3. 소리 세기

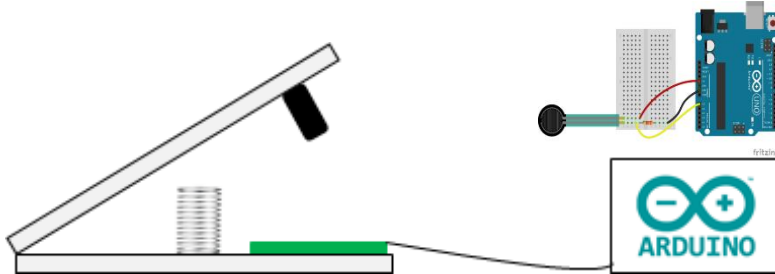
소리의 세기는 정적인 드럼 객체와 동적인 드럼스틱 객체의 상대 속도에 비례하여 출력한다.

```
audioSource.volume = collision.relativeVelocity.magnitude
```

4.2.3 Kick drum

Harmony의 Drum은 leap motion으로 연주가 불가능한 kick 드럼 소리를 아두이노를 통해 연주 환경이 있다.

▶ ‘아두이노를 이용한 드럼 Kick sound 출력’

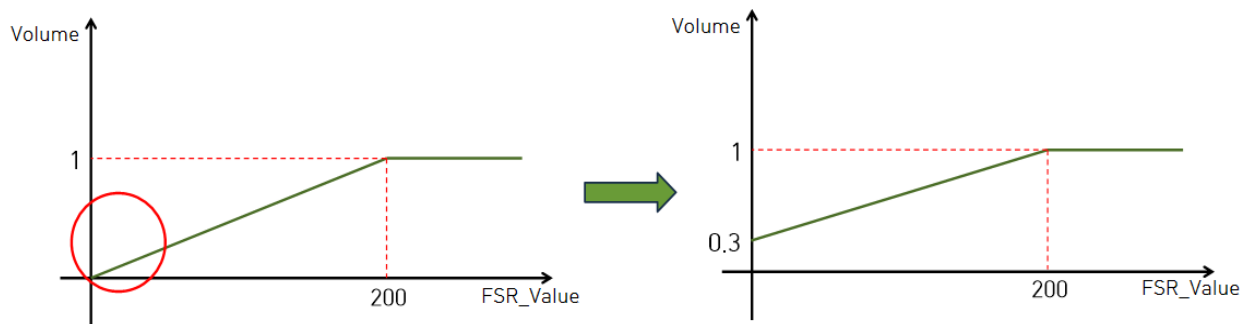


[Figure 25] Kick hardware 구성



[Figure 26] Kick

1. Unity와, Arduino에서 각각 Connection을 Open한다.
2. Unity → Arduino : MyDevice.Write(comm_start_msg)를 통해 Serial communication의 시작을 알린다.
3. 위와 같은 페달에 FSR-406 센서를 붙여 발로 밟으면서 센서에 압력을 가한다.
4. Arduino → Unity : Serial.SendData(FSR_Value)를 loop()에서 data를 보낸다.
5. Unity : FSR_Value를 mapping하여 volume 조절 후 Sound 출력한다



페달을 약하게 쳤을 때, 소리가 거의 들리지 않기 때문에 Volume의 값을 0.3 ~ 1사이로 다시 mapping한다.

[Figure 25] kick sound graph

4.3 Keyboard

HM Software는 키보드는 Classic Piano, Electric Piano와 Synthesizer로 이루어진다. 사용자는 유저 인터페이스를 통해 세가지 악기 중 원하는 악기를 선택해 연주 할 수 있고, Synthesizer는 LFO Modulation Ratio, Echo Ratio, Harmonic Ratio 슬라이더를 통해 소리를 사용자가 Customize할 수 있다. 기본적으로 건반을 손 객체가 눌렀을 때 눌리는 정도를 건반의 회전각으로 판단하여 소리를 출력하며, 키보드로도 연주가 가능하다. 또한 서스테인 페달 기능도 포함한다.

4.3.1 keyboard

▶ '키보드 연주 제공'



[Figure 27] 건반악기 연주 화면

1. 연주 방식

(1) Classic Piano와 Electric Piano 연주 방식

- 각 건반과 손 객체에 물리적 특성을 가지도록 Rigidbody 컴포넌트와 Collider를 추가한다.
- 손 객체가 가상환경에 구현된 건반 객체들을 물리적인 힘으로 누르면 건반이 회전하고, 건반이 눌렸다고 판단될 정도로 회전하면 사운드를 출력하는 방식으로 연주한다.
- 건반이 눌러 소리가 출력되고 난 후엔 시간에 따라 누르기 전 상태의 각도로 되돌린다.

(2) Synthesizer 연주 방식

- 건반은 자신의 음에 맞는 주파수를 갖는다. 사용자가 신디사이저 악기를 선택하고, LFO Modulation Ratio, Echo Ratio, Harmonic Ratio 슬라이더를 통해 소리를 Customize한다.
- Sin파, Pulse파, Triangle파 중 선택된 파형으로 신디사이징된 사운드를 onAudioFilterRead()를 통해 구현한 Oscillator를 통해 건반이 눌러 회전각이 변해 있는 동안 출력해 준다.

최종보고서: Leap Motion을 이용한 가상 악기 연주 및 음악 창작 공유 플랫폼 (Harmony)

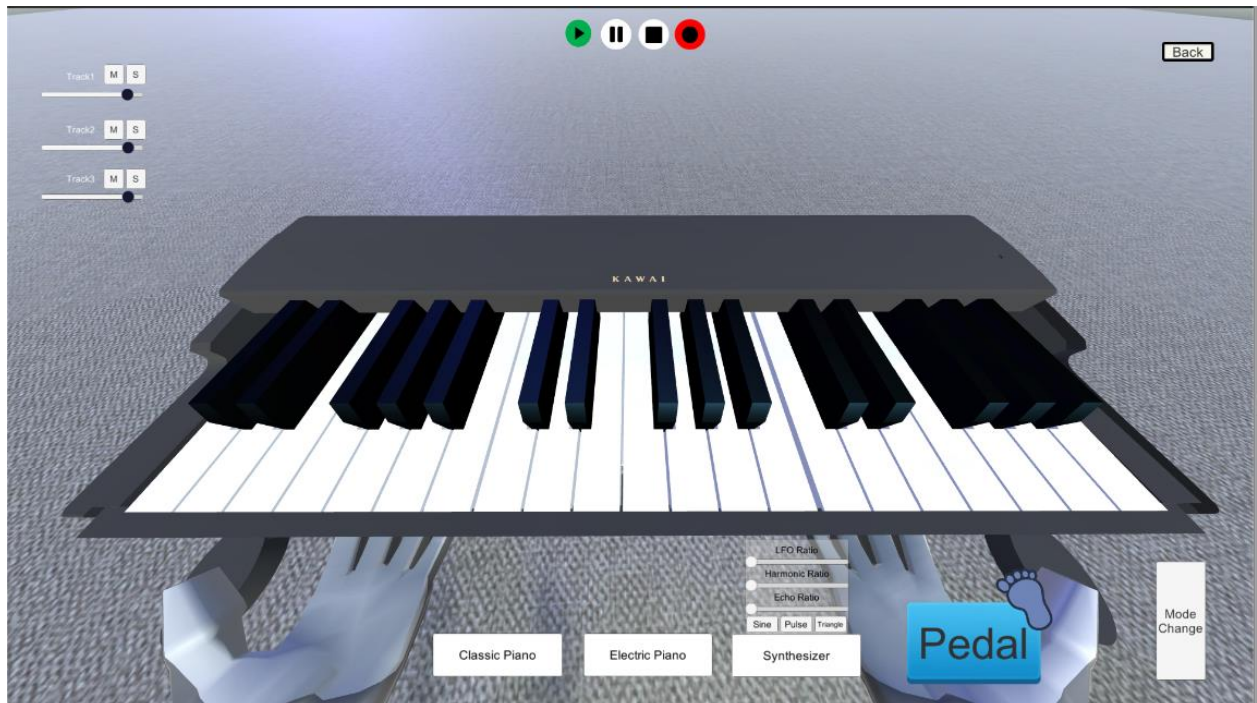
(3) 추가적인 연주법



[Figure 28] Keyboard 조작법

- 건반 악기 연주 화면에서는 립모션으로 연주를 하지만 키보드로도 보다 원활한 연주가 가능하다.
- Classic Piano와 Electric Piano, Synthesizer에서 위 사진과 같은 방법으로 연주 가능하다.
- 건반은 [Figure 27]과 같이 mapping되어 있으며 키보드상의 키를 누르게 되면 립모션으로 연주하는 것과 같이 건반이 눌리게 된다.

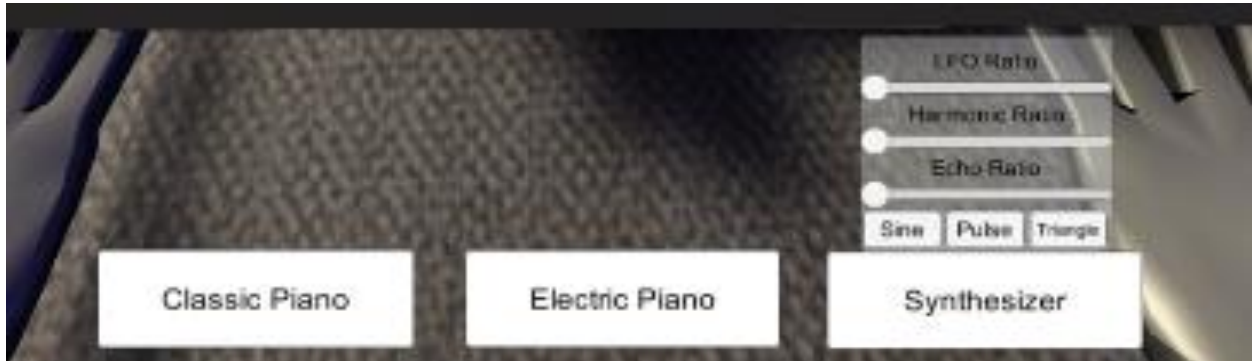
2. 악기 구조



[Figure 29] 건반악기 화면

Keyboard는 3옥타브 36건반으로 구성된다. 유저인터페이스에서 Classic Piano, Electric Piano, Synthesizer를 선택할 수 있다.

4.3.2 Synthesizer



[Figure 30] Synthesizer

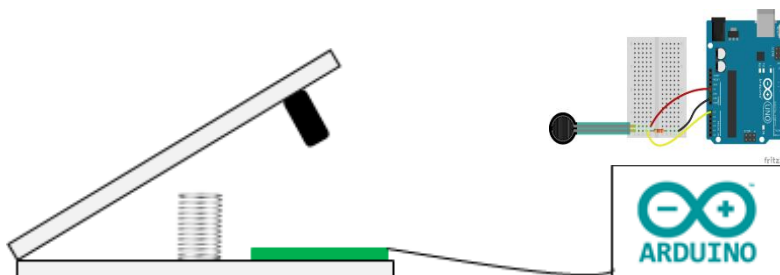
신디사이저에서는 Oscillator를 통해 사용자가 Sine wave, Pulse wave, Triangle wave 중 선택한 파형을 기본 파형으로 출력한다. 파형을 조작할 수 있는 슬라이더는 LFO Ratio, Harmonic Ratio, Echo Ratio가 있다.

- LFO ratio : 저주파 발진기를 통해 0 ~ 10 Hz의 저주파 Sine 파형을 생성하여 기본파형의 주파수를 조작한다. 이로 인해 음이 위아래로 떨리는 사운드를 생성한다.
- Harmonic Ratio : 기음인 주파수로 생성한 파형에 n 배수의 배음의 주파수로 생성한 파형을 합성하는 비율을 설정한다. 이로 인해 기음에 색을 더한 사운드가 제공된다.
- Echo Ratio : 신디사이저 특성상 사운드가 기계적이므로 Echo기능을 추가하여 좀 더 자연스러운 소리를 내도록 한다.

4.3.3 Sustain Pedal

Harmony의 Keyboard는 아두이노를 통해 손쉬운 조작으로 leap motion으로 연주가 불가능한 sustain pedal의 기능을 대신한다.

▶ 'Arduino를 이용한 Keyboard의 Sustain pedal 제어'



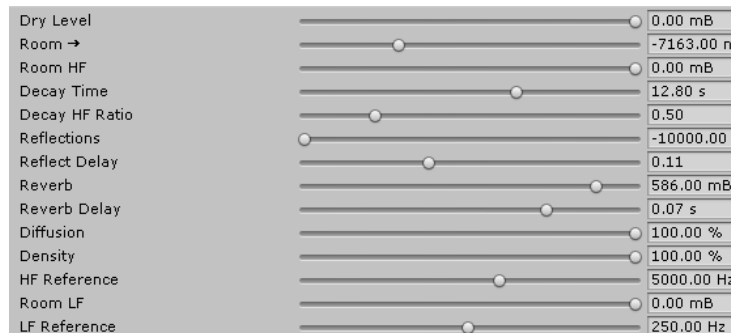
[Figure 31] Pedal hardware 구성



[Figure 32] Sustain pedal

최종보고서: Leap Motion을 이용한 가상 악기 연주 및 음악 창작 공유 플랫폼 (Harmony)

1. Unity와, Arduino에서 각각 Connection을 Open한다.
2. Unity → Arduino : MyDevice.Write(comm_start_msg)를 통해 Serial communication의 시작을 알린다.
3. Arduino → Unity : Serial.SendData(FSR_Value)를 loop()에서 data를 보낸다.
4. Unity :
 - (1) FSR_Value가 특정 값 이상일 때(Pedal을 밟고 있을 때), ReverbLevel과 Decay time을 늘려 Reverb 효과를 ON 시킨다.
 - (2) FSR_Value가 특정 값 이상이 아닐 때(Pedal을 밟지 않고 있을 때), ReverbLevel과 Decay time을 원래 상태로 조절해 Reverb 효과를 OFF 시킨다.



[Figure 33] Reverb filter

4.4 Launch Pad

HM Software의 런치패드는 16개의 pads로 이루어져 있다. 각 패드에는 사용자가 원하는 소리를 file load를 통해 넣을 수 있고, 손객체와 pad의 접촉으로 소리를 내게 된다. 런치패드를 치는 손의 속도를 감지하여 사운드 볼륨을 조절하며 연주가 일어나는 도중에도 사용자 임의로 개체에 속해 있는 음악파일을 변경할 수 있고 루프스테이션 기능 또한 포함한다.

4.4.1 launch pad

1. 연주 방식

Leap motion을 통해 사용자의 손을 인식하고 가상환경에 손 객체를 생성한다. Leap motion이 인식한 물리적 성질을 갖는 손가락인 rigidfinger에 런치패드 객체의 콜라이더와 반응할 tag를 심어준다.

```
if (bones[i] != null) {
    CapsuleCollider capsule = bones[i].GetComponent<CapsuleCollider>();
    bones[i].tag = "target";
}
```

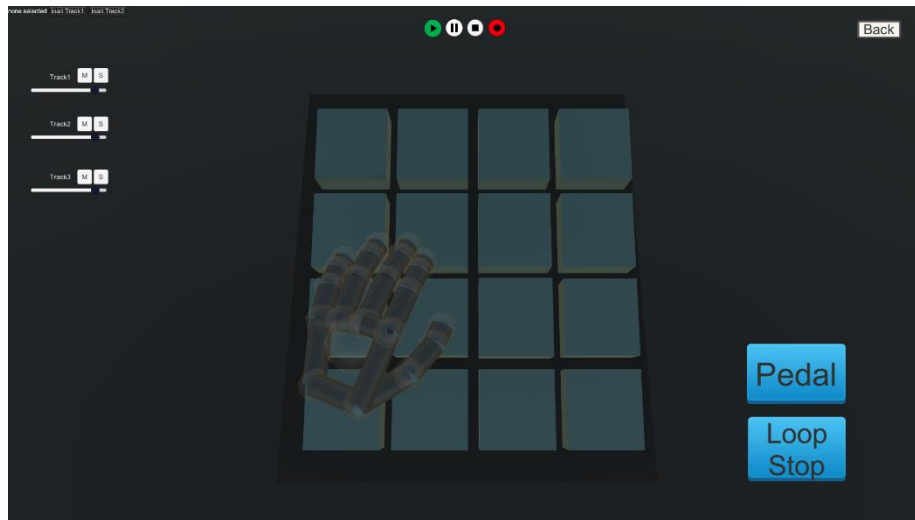
런치패드의 mesh collider와 손 객체의 target tag가 충돌했다는 값을 받으면 sound를 출력한다.

```
if (collision.gameObject.CompareTag("target"))
{
    AudioSource.PlayOneShot(play, audioSource.volume);
}
```



[Figure34] 가상 런치패드 위치

2. 악기 구조



[Figure 35] Launch pad 연주 화면

HM Software는 다음과 같은 런치패드의 1인칭 연주화면을 제공한다. 총 9개의 연주 패드가 있고 각각의 패드에는 사용자의 선택에 따른 audiosource를 받고 그에 따라 사운드를 출력한다. 메뉴에 있는 또 다른 소스들이 패드의 소스들과 변경되어 다양한 사운드들이 출력된다.

3. 세기

소리의 세기는 정적인 런치패드 객체와 동적인 손 객체의 상대 속도에 비례하여 출력한다.

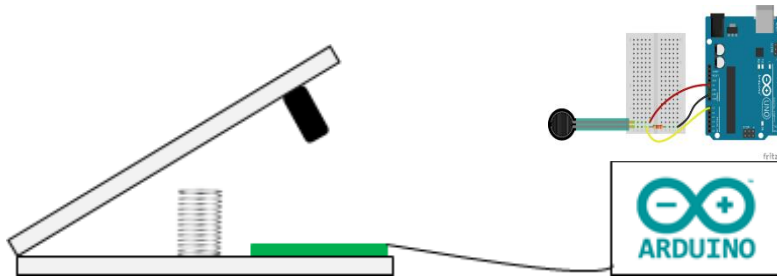
```
audioSource.volume = collision.relativeVelocity.magnitude
```

▶ '런치패드 악기 연주 제공'

4.4.2 Loop station

Harmony의 launch pad는 아두이노를 통해 손쉬운 조작으로 반복 동작을 할 수 있는 loop station 기능이 있다.

▶ 'Loop station 기능 제공'

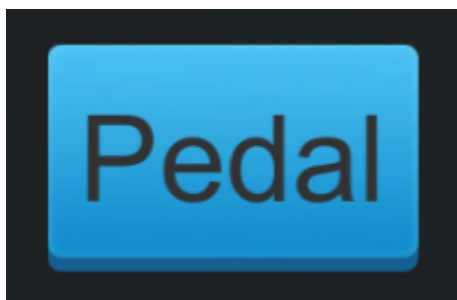


[Figure 36] Loopstation hardware 구성

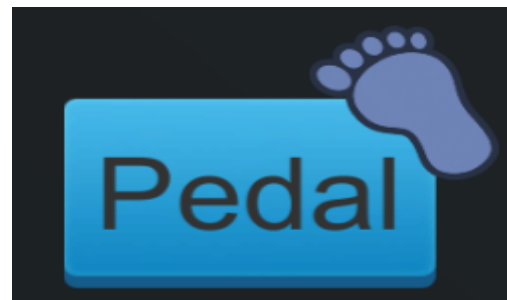


[Figure 37] Loop station

1. Arduino에서는 Sensor값을 루프를 통해 계속해서 압력 값을 보낸다.
2. 연주자가 페달을 한번 밟게 되면(특정 값 이상이 들어오게 되면) HM Software에서는 Loop Station기능이 On 되어 녹음이 시작된다.
3. 녹음이 시작되면 현재 연주되고 있는 악기 사운드가 Audio Listener로 들어오게 되고 실시간으로 녹음이 된다.
4. 연주자가 페달을 한번 더 밟게 되면 현재까지 녹음된 파일이 무한 재생되며 연주자는 또 다른 사운드를 추가할 수 있다.
5. 위와 같은 행위를 반복하여 박자 위에 박자를, 박자 위에 음을 쌓을 수 있다.
6. 모든 연주를 마친 사용자는 Loop Stop 버튼을 눌러 최종적으로 녹음된 파일을 저장한다



[Figure 38] loop 비활성화



[Figure 39] loop 활성화

최종보고서: Leap Motion을 이용한 가상 악기 연주 및 음악 창작 공유 플랫폼 (Harmony)

4.5 HM Web server + Database



[Figure 40] Harmony Web Site 홈화면

4.5.1 Web Server Login & Logout

- ① 사용자가 자신의 ID, PW를 담은 로그인 정보를 POST method로 서버에 보낸다.
- ② 서버는 사용자로부터 받은 로그인 정보를 mysql DB에 있는 회원정보 테이블과 비교하여 유효한지 검사하여 html파일로 response 메시지를 보낸다.
- ③ 사용자가 logout 요청을 POST method로 서버에 보내면 기존의 login session 끊고 logout된 html파일을 response 메시지로 보낸다.

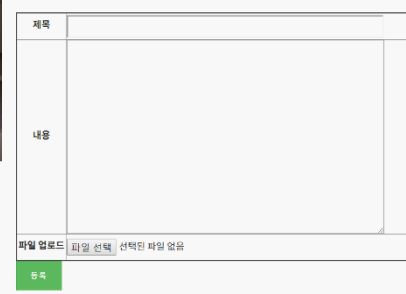
4.5.2 Web Server 회원가입

- ① 사용자가 자신의 ID, PW, 악기선택에 대한 정보를 담은 메시지를 POST method로 서버에 보낸다.
- ② 서버는 사용자로부터 받은 회원가입 정보를 mysql DB의 회원 테이블에 저장한다. 이때, 아이디를 primary key로 하여 중복된 아이디가 있으면 다른 아이디를 만들라는 메시지를 알린다.
- ③ 회원가입이 성공적으로 되면 성공했음을 알리기 위해 web site의 main page를 response 메시지에 담아 보낸다.

4.5.2 File Up/Download와 게시판 기능 제공'



[Figure 41] 게시글



[Figure 42]글쓰기 화면

- ① HM website에 접속하면 서버에 업로드 된 게시글들의 목록이 보여진다.
- ② 게시글의 제목을 누르면 게시글의 내용을 보여준다.

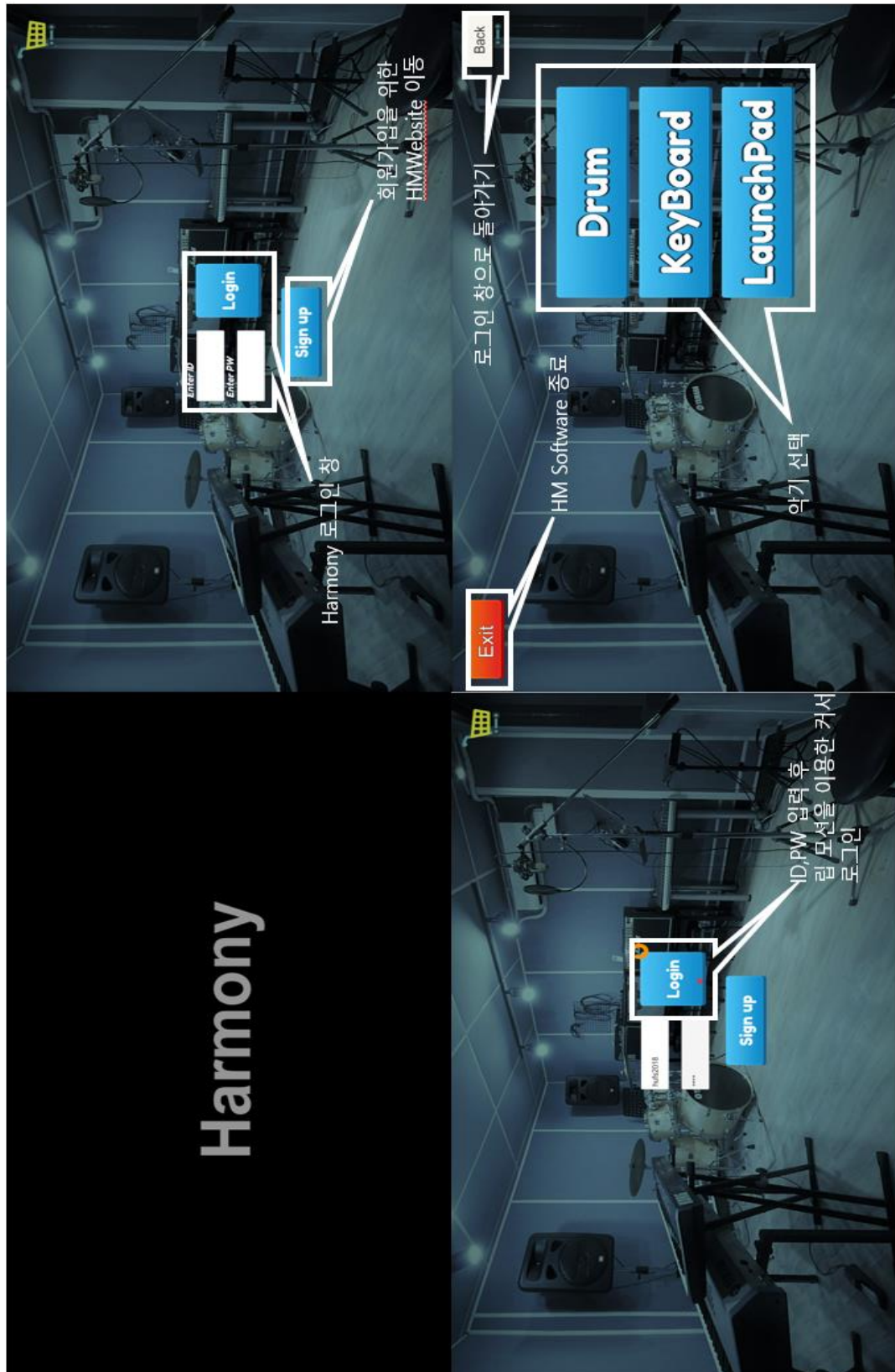
최종보고서: Leap Motion을 이용한 가상 악기 연주 및 음악 창작 공유 플랫폼 (Harmony)

- ③ HM website 홈 화면에서 글쓰기 버튼을 누르면 게시글에 포함될 내용들의 형식이 나타나고 그 중에 파일선택 버튼을 누르면 ID, 그리고 filename이 포함된 업로드 되는 file의 path가 local 변수에 저장되고 file이 업로드 된다.
- ④ 글쓰기를 완료하고 등록 버튼을 누르면 게시글의 번호와 내용, 그리고 위의 local변수에 저장된 ID, filename이 포함된 업로드 되는 file의 path가 Database에 테이블 형태로 저장이 되고 게시글이 업로드 된다.
- ⑤ 사용자가 게시글의 '파일 다운로드'를 누르면 다음 페이지로 넘어가며 음원 자동 재생 및 다운로드 버튼으로 음원을 다운로드 할 수 있게 된다.
- ⑥ Web Server는 DB에 저장된 해당 게시글 파일의 경로를 통해 음원을 로딩해 스트리밍 및 다운로드 기능을 제공해 주며 사용자는 탐색기를 통해 자신의 로컬 디렉터리에 파일을 저장한다.



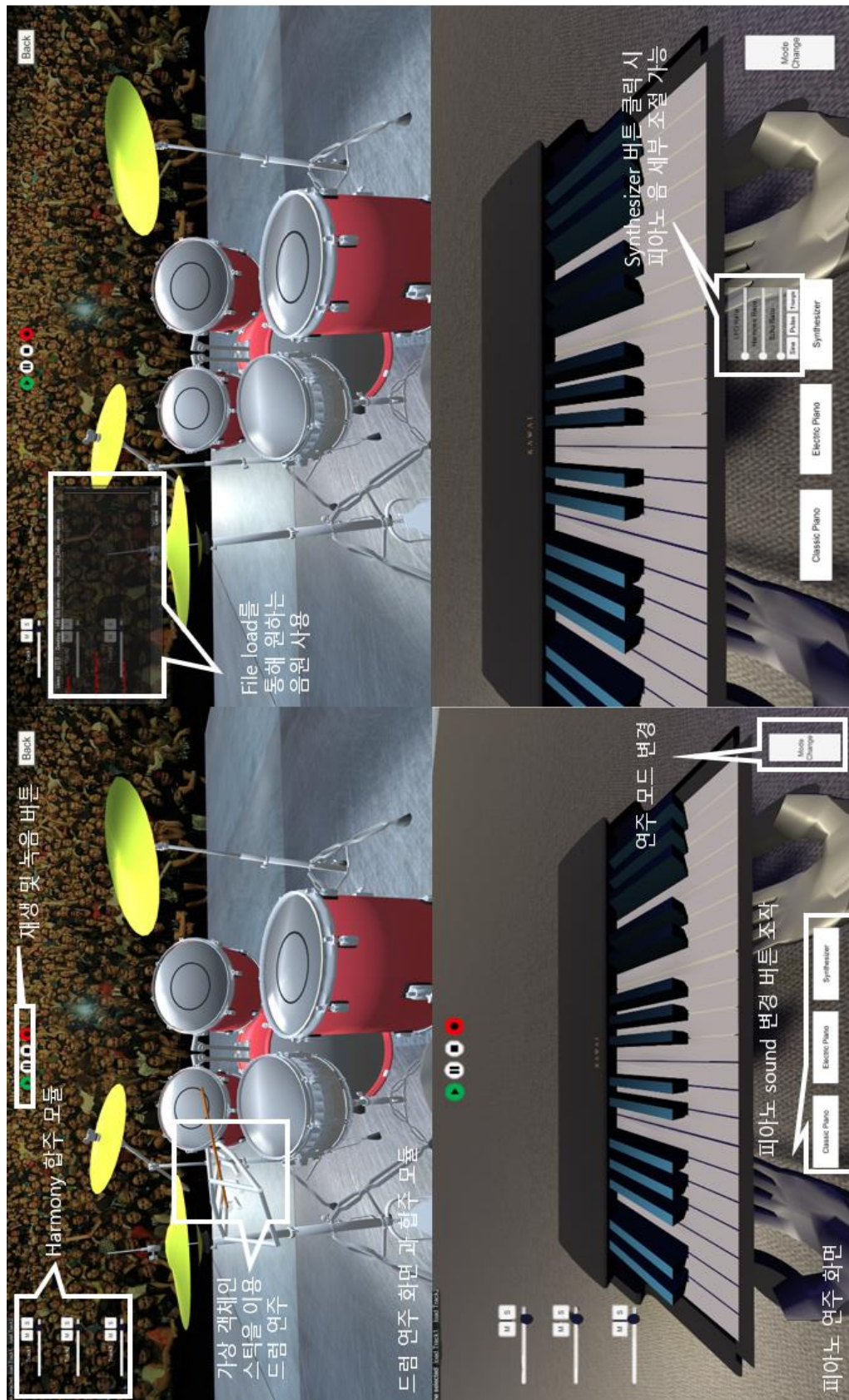
[Figure 43] Database Table 구조

5. 유저 인터페이스 및 가이드



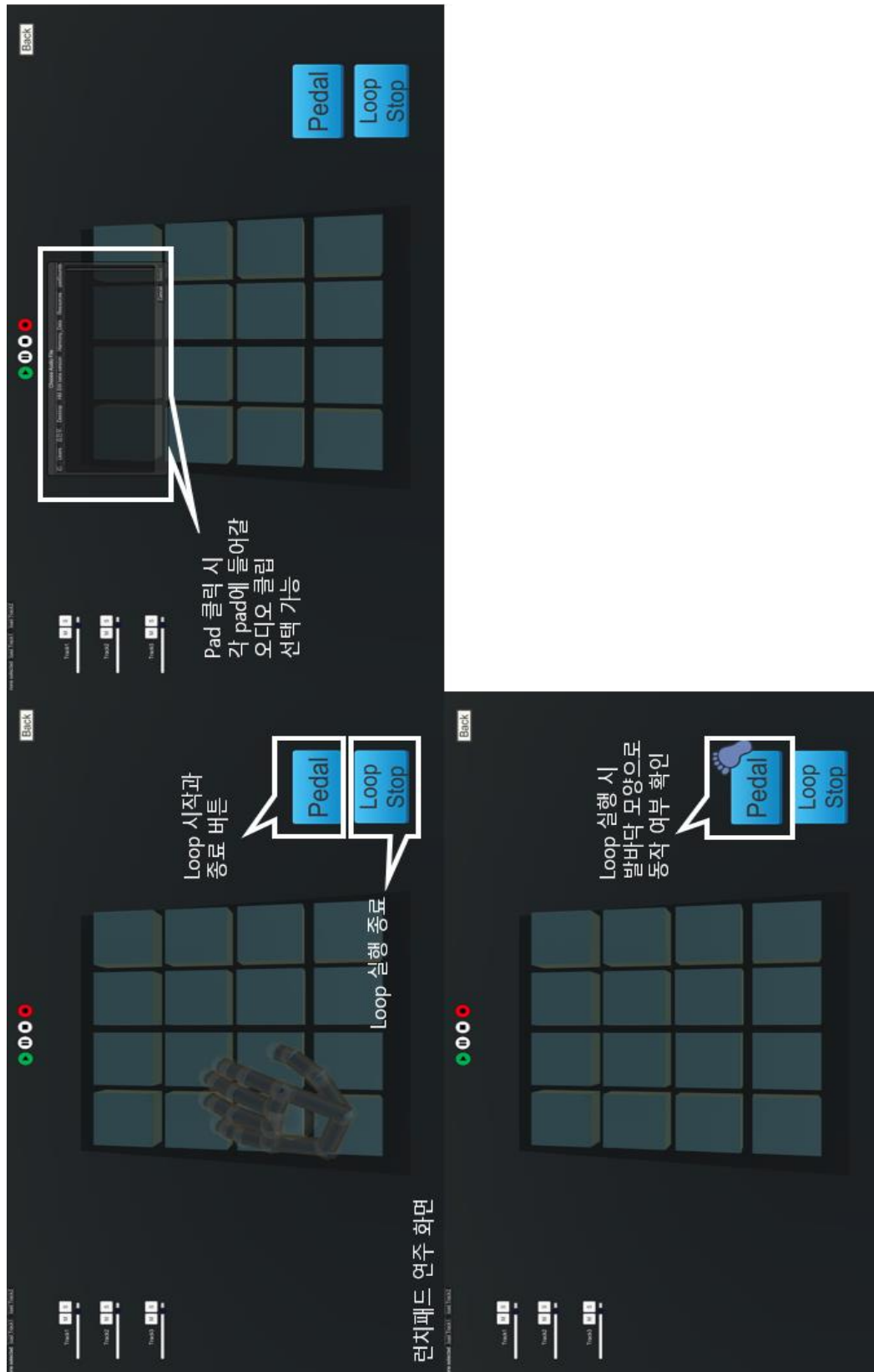
[Figure 44] 인터페이스(1)

최종보고서: Leap Motion을 이용한 가상 악기 연주 및 음악 창작 공유 플랫폼 (Harmony)



[Figure 45] 인터페이스(2)

최종보고서: Leap Motion을 이용한 가상 악기 연주 및 음악 창작 공유 플랫폼 (Harmony)



[Figure 46] 인터페이스(3)

6. 적용방안 및 기대효과

1. Harmony의 드럼, 키보드, 런치패드를 이용하는 사용자는 값이 비싸 접하기 힘든 실제 악기들을 시간과 공간의 제약을 받지 않는 환경에서 자유롭게 연주할 수 있다.
2. Harmony software 사용자는 보다 더 다채로운 창작 활동을 할 수 있다. 특히 합주 모듈을 사용자 임의로 자유롭게 음원을 넣어 재생 또는 녹음을 할 수 있고 루프스테이션을 이용해 빠르고 편하게 음을 더해나갈 수 있다.
3. Harmony website를 통해, 사용자들은 자신이 창작한 곡들을 게시할 수 있다. 또한 다른 사용자들의 곡을 다운로드 받아, HM Software에 load하여 자신의 연주를 더해나갈 수 있다.

강점(Strength)	약점(Weakness)
<ul style="list-style-type: none"> • 시공간의 제약을 받지 않는 악기이다. • 곡에 음을 더하는 것이 자유롭다. • 창작활동을 보장한다. • 다른 사용자와 곡을 공유할 수 있다. • 값이 실제 악기보다 저렴하다. 	<ul style="list-style-type: none"> • 모션 인식 장치의 한계가 있다. • 모션 인식 장치가 그림자의 영향을 받는다. • 더 많은 악기의 추가가 필요하다. • PC와의 연결이 필요하다.
기회(Opportunity)	위협(Threat)
<ul style="list-style-type: none"> • 1인 1악기 연주능력이 대세가 되고있다. • 바쁜 현대인들도 악기를 연주하고싶지만 여건이 안되어 하지 못하는 사람이 많다. 	<ul style="list-style-type: none"> • 단일 악기 서비스를 하는 경쟁사가 늘어나고 있다.

[Table 4] SWOT 분석



최종보고서: Leap Motion을 이용한 가상 악기 연주 및 음악 창작 공유 플랫폼 (Harmony)

7. 프로젝트 세부추진계획 및 세부일정

구분	내용	9주차	10주차	11주차	12주차	13주차	14주차	15주차
설계	Server 기본구성							
	Server 핵심기능							
	WebSite 기본구성							
	Web 디자인							
	Software 기본구성							
	Software 디자인							
	드럼							
	런치패드							
	건반악기							
	Arduino							
테스트	악기보완							
	점검							
종료	최종 시연							

[Table 5] 프로젝트 추진 일정



최종보고서: Leap Motion을 이용한 가상 악기 연주 및 음악 창작 공유 플랫폼 (Harmony)

8. 팀원 담당업무

이름	정	부	비고
김세용	Software 구축 및 UI , Software 병합 및 관리	Unity & Arduino 연동, 하드웨어 제작	팀장
김진우	Drum, Launchpad 구현	하드웨어, 문서, UI	
김준영	합주 모듈 구현, Keyboard	Project Sound 책임	
김정민	Unity & Arduino 연동 , Arduino	Web Server, Database	
유진솔	Unity & Arduino 연동 , Web Server	Arduino, Database	

[Table 6] 팀원 별 담당업무