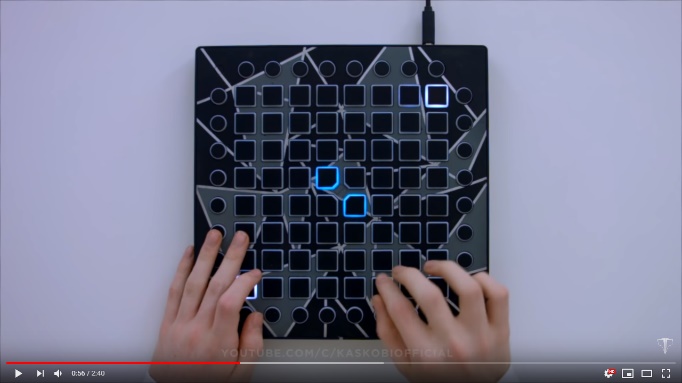
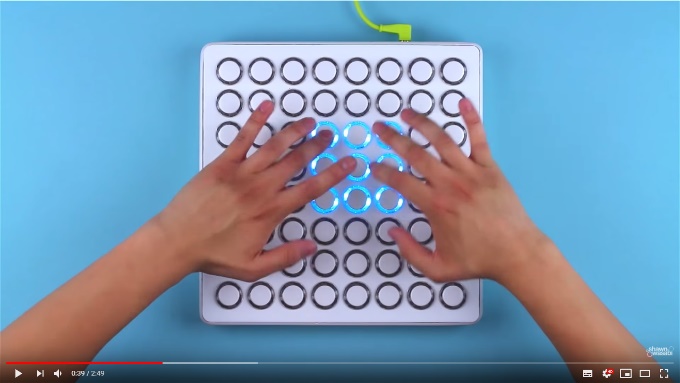
*Keyboard LaunchPad (키보드 런치패드)*

*김용현*

*서민택*

*이상승*

1. **런치패드(LaunchPad)란?**



‘런치패드(LaunchPad)’라는 악기가 있습니다. 런치패드에 대하여 간략히 설명하자면, 위의 사진에서 보이는 동그라미 혹은 네모 버튼에 각각 사운드소스(Sound source)를 할당하여 버튼을 누르게 되면 특정 소리가 나게 하는 악기입니다.

런치패드를 사용하는 목적으로는 크게 2가지의 경우가 있는데, 첫째로는 음악 프로듀서들이 미디(MIDI) 작업 혹은 시퀀싱(Sequencing)작업을 할 때 작업의 흐름을 좀 더 원할하고 용이하게 하기 위하여 사용하는 경우가 있고, 둘째로는 유튜브(Youtube)등의 동영상 플랫폼에서 자주 볼 수 있는 퍼포먼스(Performance) 용도로 사용하는 경우가 있습니다

저희는 프로그램을 개발하고자 할 때 전자보다 후자의 경우에 더 집중하였고 이에 대한 동기는 2번에서마저 이어가겠습니다.

1. **프로그램 개발동기**
   * 저희가 Keyboard LaunchPad 프로그램을 개발하고자 하는 데에는 크게 2가지 이유가 있는데 차례로 써내려 가자면

첫째로, 가격부담 문제입니다.

가격 부담 문제도 크게 2가지로 나눌 수 있는데 그중 첫째는 런치패드 그 자체 가격 문제와 그 다음으로는 런치패드에 사용하는 사운드 소스를 관리하는 데 필요한 DAW(Digital Audio Workstation, 간단히 말해서 음악을 제작하고 편집하는 소프트웨어 프로그램)의 가격 문제입니다. 런치패드 그 자체의 가격으로는 20~40만원 선으로 쉽게 접근하기에는 어려운 가격대를 형성하고 있으며, 이를 관리하는 DAW역시 최소 20~80만원 선으로 상당한 가격대를 형성하고 있습니다. 따라서, 런치패드를 단지 간단한 퍼포먼스 용도와 놀이 혹은 오락의 개념으로 접근하고자 하는 분들에게 이 가격대는 상당한 부담으로 다가올 수 있습니다.

둘째로, 어려운 조작 방법입니다.

런치패드 하드웨어의 경우는 악기 치고 상당히 간단한 조작방법으로 별 문제가 없을 수도 있지만 DAW의 사용방법의 경우에는 입문자분들이나 초보자분들이 접근하기에는 상당한 어려움을 겪을 수 있습니다. DAW에서 런치패드를 사용하는 방법을 보게 되면, DAW에서 런치패드 기능을 하는 소프트웨어의 경우에는 소프트웨어가 샘플러(Sampler)의 역할을 겸하고 있어 파형을 편집하고 이펙터(Effector)를 사용할 수 있게 하는 다양한 노브(Knob)들이 존재하며, 결국 이는 초보자나 입문자분들이 사용하기에는 상당한 어려움을 주는 요소들로 작용할 수 있습니다. (아래 사진 참고)



1. **프로그램 개요**

위에 써 내려간 내용들을 바탕으로 저희는 Keyboard LaunchPad(이하 KL)프로그램을 만들고자 생각하게 되었습니다. KL 이란, 키보드의 버튼들을 런치패드의 버튼처럼 대신 사용하는 것으로써, (이후 프로그램 구조에서 자세히 설명하겠지만 여기서 개략적으로 설명하자면) 키보드의 버튼들을 누를 때마다 해당 버튼에 해당하는 소리가 하나씩 나게 해주는 프로그램입니다.

저희는 이 프로그램을 만들고자 생각하면서 크게 3가지 부분에 집중하였습니다.

첫째로, 음악적 지식이 없는 초보자도 쉽게 이용할 수 있는 편리하고 직관적인 UI

둘째로, 예쁜 외관 디자인과 퍼포먼스 용도로도 사용할 수 있는 화려한 UI

셋째로, 놀이와 오락 퍼포먼스 용도에 적합한 정도의 꼭 필요한 기능들만 넣은 미니멀리즘

위 3가지 부분을 큰 뼈대로 하여 세부적인 내용을 구현하고자 합니다.

1. **관련 프로그램/서비스 조사 결과**

참고한 소프트웨어로는

Native Instruments – Battery 4

Image Line – Drumaxx

Ableton Live – DrumSynth Abstract

* 위 3개의 프로그램은 런치패드를 이용할 때 사용하는 가장 기본적인 프로그램으로 프로그램의 큰 틀에 대한 아이디어를 얻음.

참고한 하드웨어로는

Novation – LaunchPad MK2

Native Instruments – MASCHINE MIKRO

* 위 2개의 하드웨어로는 BANK 기능, PLAY 기능에 대한 구체적인 아이디어를 얻었습니다.

등이 있습니다.

1. **프로그램 구조**

[디자인]

디자인은 크게 ‘레이아웃 디자인’과 ‘버튼 컬러 디자인’으로 나누어져 있습니다.

<레이아웃 디자인>

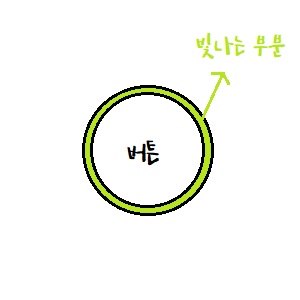
레이아웃 디자인은 UI 디자인을 말하는 것으로 

위 키보드의 레이아웃과 같이 동글동글한 모양의 버튼들을 키보드 모양과 동일하게 배치됩니다.

위 키보드 레이아웃을 보면 행이 6개인데, 이 중에서 제일 위쪽 행은 이후 ‘기능’파트에서 설명할 ‘BANK’ 기능으로 이용할 것이며, 가운데 4개의 행은 ‘PLAY’ 기능으로 이용할 것이며, 제일 아래쪽 행은 ‘MENU’ 기능으로 이용할 것입니다.

<버튼 컬러 디자인>

실제 런치패드를 보게 되면, 런치패드 버튼을 누르게 되면 해당하는 버튼과 그 주변 버튼이 예쁜 모양을 그리며 빛나게 되어 화려한 시각적 효과를 주게 됩니다.



따라서 위 그림과 같이, 버튼을 만들 때 버튼 주변으로 시각적 효과를 줄 수 있는 빛나는 부분을 만들어서 해당 버튼을 누를 때 버튼이 반짝일 수 있도록 만들 예정입니다.

[기능]

위 UI설명을 보시면 알겠지만, UI의 모든 부분은 버튼으로 이루어져 있습니다. 그렇기에 버튼의 종류를 크게 3개로 분할하였고, 각각의 버튼들은 ‘좌클릭, 우클릭, 키보드 타이핑’ 3가지 이벤트에 따라 각각 다르게 혹은 똑같이 행동하게 만들고자 합니다.

**# PLAY 버튼 (좌클릭 O, 우클릭 O, 타이핑 O)**

PLAY 버튼의 가장 큰 역할은 ‘소리의 재생’ 입니다. 자세하게 풀어보겠습니다.

\*좌클릭을 했을 때 = 해당하는 키보드를 타이핑 했을 때 : 해당 이벤트가 발생하면, ‘소리의 재생’ 이 이루어지게 됩니다. 여기서 재생되는 소리는 우리가 불러온 사운드 소스이며, 어떤 소스를 불러오고 사용할지는 우클릭 이벤트에서 고를 수 있게 됩니다.

\*우클릭을 했을 때 : 우클릭은 메뉴 버튼으로, 해당하는 버튼 관리를 할 수 있게 됩니다.

기능 1. Load(사운드 개별 소스 불러오기)

* 이 기능의 경우에는 ‘Drag & Drop’ 기능으로도 구현할 예정으로, 사용하고자 하는 사운드 소스가 있을 때, 해당 소스를 ‘Drag & Drop’ 한 후 원하는 버튼으로 끌어오면 해당 소스가 별도의 Load 버튼 없이도 Load 됩니다.

기능 2. Save(사운드 개별 소스 저장하기)

기능 3. Check(사운드 소스 파일명 확인하기)

* 이 기능의 경우에는 ‘Hovering’ 기능으로도 구현할 예정으로, 해당 버튼위에 마우스를 갖다 댔을 때 어떤 사운드소스를 이용하고 있나 바로 확인할 수 있도록 할 예정입니다.

기능 4. Solo(해당 버튼 사운드만 확인 가능, 따라서 다른 모든 버튼 Mute상태)

* 별 의미가 없어 구현 안하기로 함.

기능 5. Mute(해당 버튼 사운드만 음소거, 따라서 다른 모든 버튼은 Unmute 상태)

* 별 의미가 없어 구현 안하기로 함.

기능 6. Volume(사운드 소스 볼륨 조절)

기능 7. Panning(사운드 소스 패닝 조절) (패닝 : 소리의 좌우 균형)

기능 8. 버튼 색상 변경 기능

**# BANK 버튼 (좌클릭 O, 우클릭 O, 타이핑 O)**

BANK 버튼은 ‘소리 묶음의 저장’이 역할로, 자세히 풀어보겠습니다.

만약, Q라는 키보드 버튼에 A라는 소리가 저장되어 있으면 Q버튼을 누르게 되면 항상 A라는 소리만 재생되게 됩니다. 하지만, BANK를 이용하게 된다면 이는 상황에 따라 달라지게 됩니다. 1번뱅크의 Q라는 키보드 버튼에 A라는 소리를 저장해 두고, 2번뱅크의 Q라는 키보드 버튼에 B라는 소리를 저장해 두고 BANK만 바꿔준다면 A와 B라는 소리를 모두 Q라는 버튼을 통해 이용할 수 있게 됩니다.

BANK버튼에는 빛나는 부분은 없으며, 해당 버튼이 눌려있다는 표시만 하게 할 것입니다.

\*좌클릭을 했을 때 = 해당하는 키보드를 타이핑 했을 때 : BANK의 변경 (뱅크는 1번부터 8번까지 만들 예정입니다.)

-> BANK 12번까지 만듬.

\*우클릭을 했을 때 : Menu버튼으로

-> 우클릭 기능 구현 안함. 왜냐하면, 어차피 폴더에 BANK단위로 저장이 되기 때문.

기능 1. Bank 단위 Save

* 구현 X

기능 2. Bank 단위 Load

* 구현 X

**# MENU 버튼 (좌클릭 O, 우클릭 X, 타이핑 O)**

MENU 버튼은 ‘STOP’ 버튼과 ‘SETTINGS’ 버튼 2가지가 있습니다.

\*STOP : 재생중인 모든 소리의 정지

\*SETTINGS

기능 1. BANK 단위 사운드 저장(BANK SAVE)

* + - 기능 구현 X (BANK 파트에서 설명)

기능 2. BANK 단위 사운드 불러오기(BANK LOAD)

* + - 기능 구현 X (BANK 파트에서 설명)

기능 3. 모든 사운드 저장 (SAVE ALL)

* + - 기능 구현 X (BANK 파트에서 설명)

기능 4. 모든 사운드 불러오기 (LOAD ALL)

* + - 기능 구현 X (BANK 파트에서 설명)

기능 5. 버튼 색상 패턴 만들기

* + - 기능 구현 X (이건 퍼포먼스 용도로 만들라고 했는데 의미 없음.)

기능 6. 레이아웃 디자인 변경

# API 기능

1. SoundCloud 연동을 통하여 내가 만든 음원을 조금 더 쉽게 공유하고, 런치패드를 사용할때 올라온 음원들을 더 쉽게 참고할 수 있게 한다.
2. 음원탐색 사이트인 BEATPORT를 통하여 EDM 계열 음악을 더 쉽게 탐색할 수 있게 한다.
3. **개발 일정**

5월 2주차 (5/7 ~ 5/12) : 디자인 프로그래밍(XAML)

5월 3주차 (5/13 ~ 5/19) : 구조 프로그래밍(C#)

5월 4주차 (5/20 ~ 5/26) : 프로그램 버그 수정 및 완성도 점검

5월 5주차 (5/27~6/2) : 발표준비