OSS project proposal final report



2021.04.24

**Team 5 (강현준, 김도균, 박상엽, 손현민)**

**1. 프로젝트 주제**

‘운지법’은 악기를 처음 접할 때 가장 먼저 배우는 것이다. 기타의 경우도 마찬가지로 기본적인 코드들의 운지법을 가장 먼저 접하게 될 것이다. 하지만 특이하게도, 기타는 사람마다 운지법이 다르기 때문에 기타를 치는 사람의 입장에서 보면 어느 운지법을 써야 더욱 편하게 기타를 칠 수 있는지 판단이 되지 않는 경우가 많다. 또한 사람마다 기타를 친 경력이나 손의 크기, 손가락의 굵기 등등이 모두 다르기 때문에 사람들이 어느 운지법을 많이 사용한다고 해서 그것이 나에게 꼭 맞는다는 보장이 없다. 이 프로젝트는 특정한 사람에게 최적화된 운지법을 제공할 수 있을까? 라는 생각에서 출발하였다.

기타의 경우 악보의 어떤 음에 대하여 어느 줄의 몇 번째 프렛을 잡아야 하는지 알려주는 특별한 악보인 ‘타브 악보’가 존재한다. 하지만 이 타브 악보에는 몇 번째 손가락으로 줄을 눌러야 하는가에 대한 정보는 나와있지 않고, 이를 사용자가 적절하게 판단하여 눌러야 한다. 이런 과정에서 경험의 차이가 나타나게 되는데, 기타를 오래 쳐서 숙련된 사람들은 (물론 이 사람들은 타브 악보가 필요 없겠지만) 부드럽게 연주할 수 있지만, 그렇지 못한 사람들은 이 ‘판단’을 함에 있어 시간이 걸리기 마련이다.

기본적인 코드들에 대한 운지법은 어느 정도 보편화 되어 있는 것이 사실이지만, 이런 운지법들이 항상 기타를 연주할때 효율적인 것은 아니다. 또한, 코드에 대한 운지법이 보편화 되어있다고 해서 하나의 운지법만을 사용하는 것이 아니라, 곡에 따라 여러 운지법들이 사용된다. 멜로디를 연주하는 경우 특정한 코드를 계속해서 잡고 연주하는 것이 아니기에 이런 운지법의 변화가 더 자주 나타나게 된다. 이 프로젝트는 연주자가 얼마나 편하게 운지 할 수 있는가, 혹은 어떻게 하면 연주가 끊어지지 않고 자연스럽게 이어질 수 있는가 이 두 가지의 목표를 가지고, 개인의 특성을 고려하여 손의 움직임을 최소화하고, 편하게 운지할 수 있는 운지법을 제공하는 목적을 가지고 있다.

**2. 필요성**

이 프로젝트는 기타의 운지법을 판단하는 데 있어 어려움을 겪는 사람들에게 필요한 주제이다. 기타에 숙련된 사람이라면 굳이 운지법을 최적화하지 않아도 이미 자신의 손에 맞는 운지법을 익혔을 것이고, 비효율적인 운지법을 사용했다 할지라도 오랜 기간 같은 운지법을 사용하면서 손이 운지법에 익숙해졌을 것이다. 하지만 기타에 익숙하지 않은 사람들의 입장에서 보면, 그들에게 최적화된 운지법을 제공함으로써 기타를 더욱 잘 치기 위해 들여야 하는 노력의 양을 조금이나마 감소시킬 수 있는 것이 사실이다.

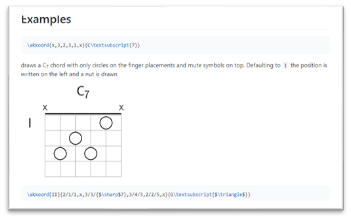
또한, 이를 오픈소스로 공개할 경우 ‘손의 크기, 손가락의 길이나 굵기 같은 다양한 정보를 통해 운지법의 최적화를 이끌어 내는 과정’에 대한 코드를 다른 개발자들과 함께 연구하고 개선할 수 있다. “A라는 특정한 음을 연주할 때 대부분 B라는 운지법을 보편적으로 사용하는데, C라는 특징을 가진 사용자들에게는 B가 아닌 B’ 운지법을 사용하는 것이 더욱 편리하다” 라는 판단을 내린다거나, “바로 다음으로 오는 음이 D일 경우에는 E라는 운지법을 사용했을 때 더욱 부드럽게 연주될 가능성이 높다” 라고 판단하는 것이 그 예시이다. 이런 판단의 과정에 대한 다양한 개발자들의 기여가 활발히 이루어 진다면, 기타 뿐만 아니라 다른 악기의 운지법을 구별하고 판단하는 것에도 영향을 미칠 수 있다.

**3. 차별성**

이 프로젝트와 관련하여 기존에 존재하는 기술을 조사하였고, 그 결과는 다음과 같다.

**1) ronligt / LaTeX-Akkoord**

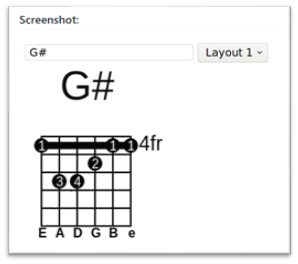
Link : <https://github.com/ronligt/LaTeX-Akkoord> (오픈소스)



기타 코드를 입력받아 눌러야 하는 부분을 그림으로 나타내주는 오픈소스 프로젝트이다.

**2) T-vK / Chord-Draw**

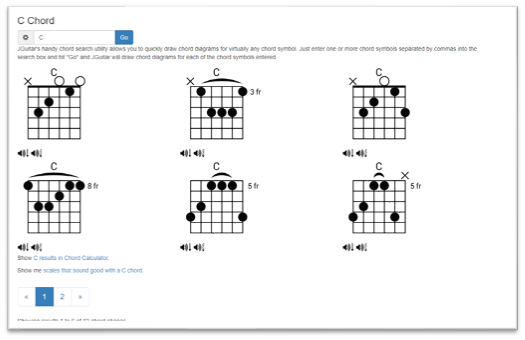
Link : <https://github.com/T-vK/Chord-Draw> (오픈소스)



기타 코드를 입력 받아 눌러야 하는 부분을 그림으로 나타내주는 오픈소스 프로젝트이다. 1)과 달리 손가락의 번호도 나타내고 있다.

**3) Jguitar**

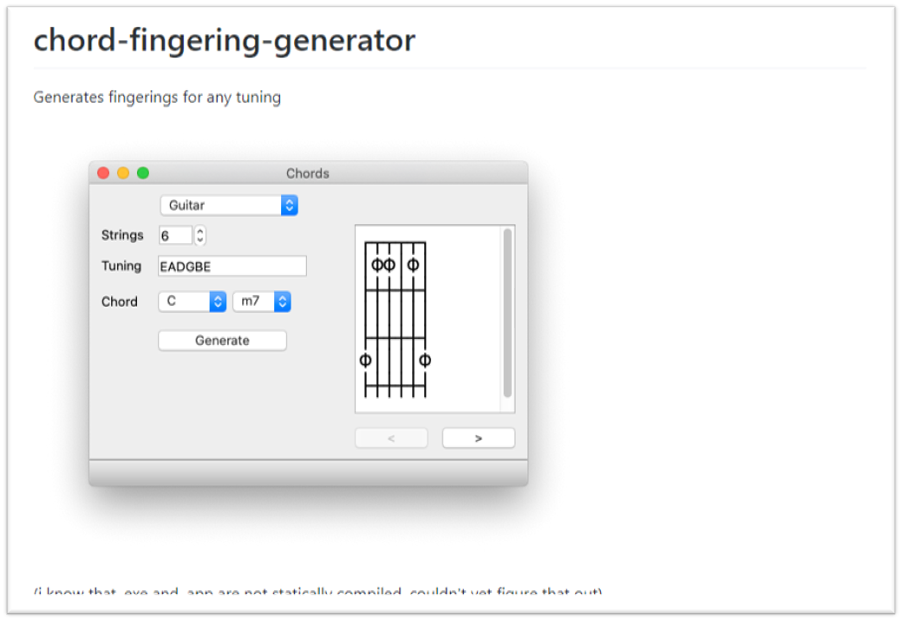
Link : <https://jguitar.com/chordsearch/>



기타 코드에 따른 다양한 운지법을 그림으로 나타내주는 사이트이다. 거의 대부분의 운지법들을 모두 나타내주고 있다.

**4) povle / chord-fingering-generator**

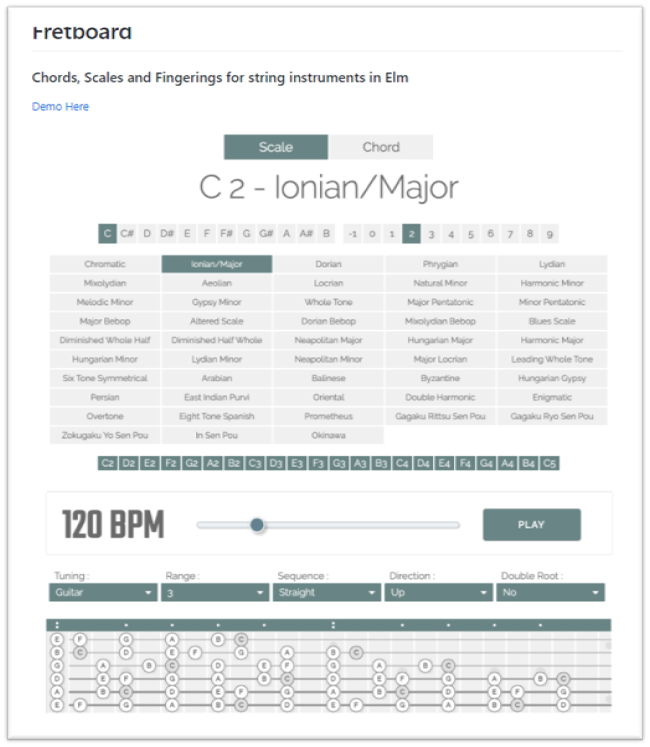
Link : <https://github.com/povle/chord-fingering-generator> (오픈소스)



기타의 코드를 입력받아 눌러야 하는 부분을 그림으로 나타내주는 오픈소스 프로젝트이다.

**5) fredericcormier / Fretboard**

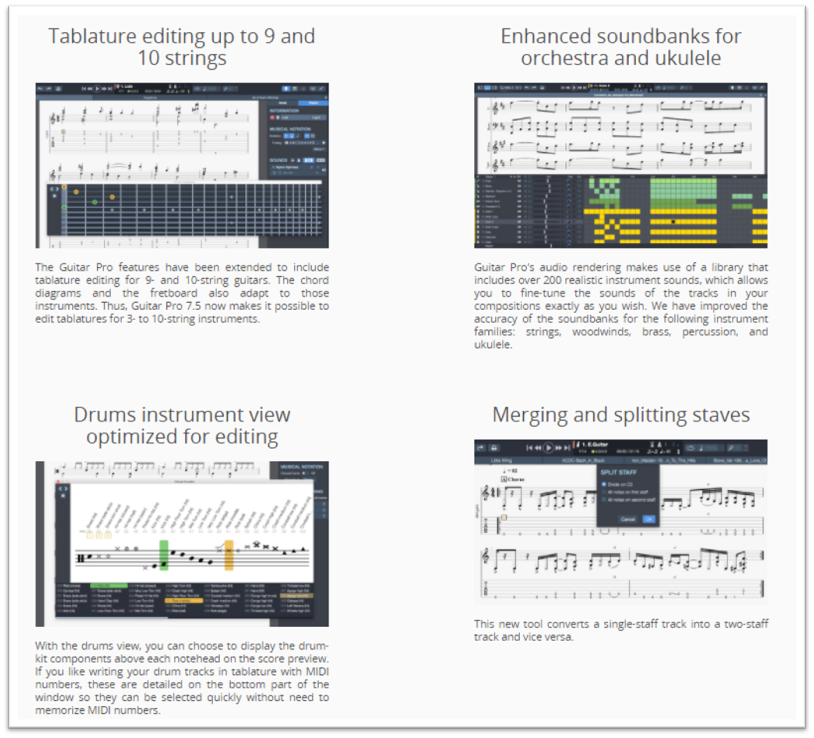
Link : <https://github.com/fredericcormier/Fretboard> (오픈소스)



현악기(기타)의 코드와 스케일, 운지법을 제공하는 오픈소스 프로젝트이다.

**6) Guitar pro**

Link : <https://www.guitar-pro.com/>



기타 관련 프로그램이다. 관련된 기술로는 악보에 마디별 음들을 모두 기타 지판위에 그림으로 나타내 주는 것이 있다.

**기존 기술들의 한계점**

1. 기존의 프로젝트들은 해당코드를 입력하면 코드의 운지법을 알려준다. 하지만 해당 방식은 하나의 코드를 독립적으로 생각하고 운지법을 알려주기 때문에 이전 또는 다음 코드 운지법과의 상호관계를 생각하지 않는다. 따라서 수많은 코드들의 선택지가 있어도 ‘초보자’들에게는 이전 코드에서 해당 코드를 칠 때, 또 해당 코드에서 다음 코드를 칠 때, 비효율적인 운지법을 선택할 수 있다.
2. 해당 프로그램은 직접 코드를 입력해야 운지법을 알려주기 때문에 하나하나 검색하려면 시간적 비용이 꽤 많이 든다.

**제안하는 기술의 차별성**

1. 우리는 바뀌는 코드간의 관계를 생각해 효율적인 운지법이 가능하게 한다. 이는 초보자들이 상황에 맞게 하나의 코드를 유동적으로 최적의 운지법으로 연주할 수 있게끔 해준다.
2. 우리의 프로그램은 맨 처음 악보 데이터를 입력하기만 하면, 알고리즘을 통해 효율성을 고려한 운지법들을 알려줄 것이다.

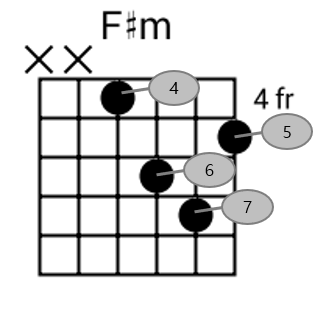
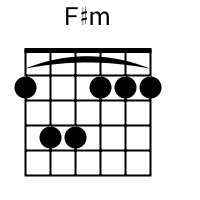
**4. 개발 방법 및 계획**

**1) 코드에 따른 운지법**

기타를 치는 방법은 스트로크와 핑거스타일, 크게 두가지로 나뉜다. 스트로크는 왼손으로 코드를 잡고 오른손으로 기타줄을 튕겨 소리를 내는 방식이다. 주로 반주를 해주는 기타가 오른손을 크게 흔들고 있다면 이 주법을 사용하고 있는 것이다. 기타의 기본 주법으로써, 왼손이 보통 노래의 한 마디 정도를 한 가지 코드만 잡고 있는 것이 일반적이다. 핑거스타일이란 비교적 왼손이 정적인 스트로크 주법과 달리 왼손이 바쁘게 움직이는 주법이다. 주로 왼손 손가락 하나하나가 한음을 잡고 오른손 손가락으로 줄을 튕겨 소리를 낸다. 기타 악보중 스트로크 주법이 사용되는 악보는 주로 타브악보보다는 간단히 코드만 써놓은 악보들이 많다. 하지만 핑거스타일 악보에서 우리는 이전에 말한 타브 악보를 많이 본다. 핑거스타일의 타브악보가 코드 악보에 비해서 왼손 손가락을 표현할 수 있는 범위가 많다지만 결국 한마디 코드 안에서 음들이 결정된다. 따라서 해당 마디의 코드를 안다면 왼손으로 그 코드를 잡고 몇가지 손가락의 변화만 주면 연주할 수 있는 경우가 많다. 우리는 이 방법을 통해 왼손의 상황에 맞는 코드가 있다면 그에 따른 왼손 운지법을 알아내는 알고리즘을 개발 할 수 있을 것이다. 예를 들어 다음 곡의 마디를 보자. 이 곡은 Sting의 Shape of my heart이다.



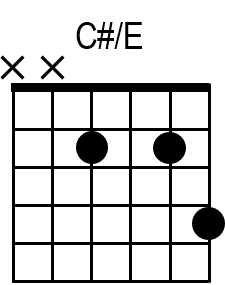
마디의 첫 절반 멜로디를 보면 하나하나 떨어진 음을 치는것 같지만, 사실 한 F#m 코드를 잡고 친 부분이다. 그렇다면 F#m 코드를 보자.



왼쪽은 Em 코드를 이용한 F#m의 일반적인 코드이다. 하지만 이 곡에서는 오른쪽 Dm 코드를 이용한 코드를 사용하길 원한다. 4fr은 기타의 4번째 프렛이라는 뜻으로 여기에 따르면 오른쪽 코드의 타브 악보는 4, 6, 7, 5가 되어 훨씬 편하게 연주를 할 수 있다.

**2) 손가락 부담 알고리즘**

코드에서 손가락 운지법을 알아내는 알고리즘을 개발하는것에 성공했다면 다음은 손가락에 가해지는 부담을 알아내는 알고리즘을 만들 차례다. 위의 F#m 코드의 경우 사실 4개의 점을 다 짚어야되는것이 아니다. 3번째 손가락의 경우 짚지 않는것이 더 편하다.



또한 다음 절반 멜로디를 치기 위한 코드는 C#/E 라는 코드인데 이를 짚기 위한 운지법은 손가락을 길게 뻗어 잡는 바레 코드를 잡거나, 조금더 손에 편하게 하나씩 잡는 방법이 있다. 평소의 경우에는 손가락 하나를 잡는게 손가락에 무리가 덜 가겠지만, 이 곡 같은 경우에는 약지손가락의 위치때문에 바레 코드가 더 쉬운 축에 속한다.

이렇게 곡의 전후 코드에 맞게 코드를 잡는 법을 바꿀 수 있다면, 이것을 타브 악보 하나하나의 음에 적용한다면 곡의 흐름에 맞는 운지법 뿐만 아니라 작은 마디에서도 좋은 운지법을 찾을 수 있을 것이다.

**3) 알고리즘과 머신러닝**

위 내용들 구현하기 위한 알고리즘을 개발할 수도 있겠지만, 머신러닝을 통한 개발도 가능할것으로 보인다. 변수를 악보에서 나온 다양한 변수들, 예를 들어 각 프렛간의 거리, 동일 줄에서의 빈도, 앞 뒤 코드등을 사용하면 될것으로 보인다. 만약 이러한 개발을 진행하게 된다면 유튜브 영상이나 촬영한 기타 연주 영상들을 자료로 사용하여 학습시킬 것이다.

**4) 손모양 인식**

위의 머신러닝의 자료, 운지법의 자료를 얻기 위해서 연주 영상을 사용한다면 손 모양을 인식하여 각 손가락의 위치를 알아낼 수 있을 것이다. 또한 연주자의 손 모양이나 특징 또한 변수에 포함한다면 더 다양한 학습을 할 수 있을 것으로 보인다.

**5) 사용자의 손특징**

우리가 위의 4가지 방법을 이용하여 표준 운지법을 만든다고 하더라도 사람마다 서로 다른 손 모양, 크기, 특징들이 있을 것이다. 그렇다면 어떤 사람은 태생적인 한계로 인해 특정 운지법을 사용하지 못하는 상황이 발생 할 것이다. 이런 상황을 위해서 사용자의 손 특징들을 고려를 해야 한다.

이를 위해 프로그램 실행 전에 사용자의 손 특징을 입력하는 부분을 만들것이다. 사용자의 손의 크기, 손가락의 길이 같은 여러 특징들을 확인하는 방식(사용자가 본인의 경우에 맞게 옵션을 선택)을 사용하거나 튜토리얼을 만들어 다양한 운지법들을 제공하고 이를 소화할 수 있는지 테스트하여 손의 크기를 체크하는 방식으로 구현하는 것도 가능하다. 더 나아가 프로젝트 개발과 개선이 활발히 이루어 진다면, 제공된 여러 운지법들 중에 사용자가 선택한 운지법을 분석하여 손 모양이나 크기같은 여러 정보들을 유추하는 알고리즘을 개발할 수도 있을 것이다.

추가로, 알고리즘이 제안한 운지법의 코드들 말고도 숙련자들이 더 나은 운지법을 제안하는 경우 이를 반영할 수 있도록 하는 기능도 개발될 수 있을 것이다.

**5. Feedback 반영 내용**

**1) 왜 최적화를 해야하는가에 대한 고민이 필요하다. 운지법을 쉽게 배우기 위해서? 아니면 연주할때 hand movement를 최소화해서 쉽게 연주할 수 있도록 하기 위해서?**

둘 모두. 기타에 익숙하지 않은 사람들에게 최적화된 운지법을 제공함으로써 들여야 하는 노력의 방향성을 개선시켜준다. 병렬적으로 음악에 따른 hand movement를 최소화시키는 것도 목표이다.

**2) 사용자 customizing(손가락이 짧은 경우 혹은 주요 사용 손가락을 설정하여 운지법 에 대해서 알려주는 등)에 대해서 좀 더 생각해보면 좋겠다. 사람마다 손크기 같은 것이 다른데, 이것에 대한 표준이 없는 것 같다.**

사용자의 손 크기, 손가락의 길이, 굵기와 같은 정보들을 입력으로 받고 그에 따라 최적화된 운지법들을 제공하는 쪽으로 프로젝트의 방향을 잡았다. 만약 이 오픈소스 프로젝트가 다양한 개발자들에 의해 개선될 경우, 프로그램 시작 시 같은 코드에 대한 여러 운지법들을 제안하고, 사용자가 선택한 운지법들을 토대로 손 모양을 유추, 분석하여 운지법을 최적화 시키는 방향도 고려하고있다.

**3) 초보자들이 숙련이 되었을 때는 어떤 기능을 제공해 줄 수 있는지도 생각해봐야한다.**

위에서 언급한 숙련자들이 더 나은 운지법을 제안하고, 그에 따라 운지법을 최적화하는 기능도 개발할 수 있을 것이다. 새로운 곡을 배울 때 숙련자들 스스로가 생각한 운지법과 알고리즘이 제안한 운지법을 비교한다면 더욱 정교하고, 효과적인 알고리즘을 만드는 데 도움이 될 것이다.

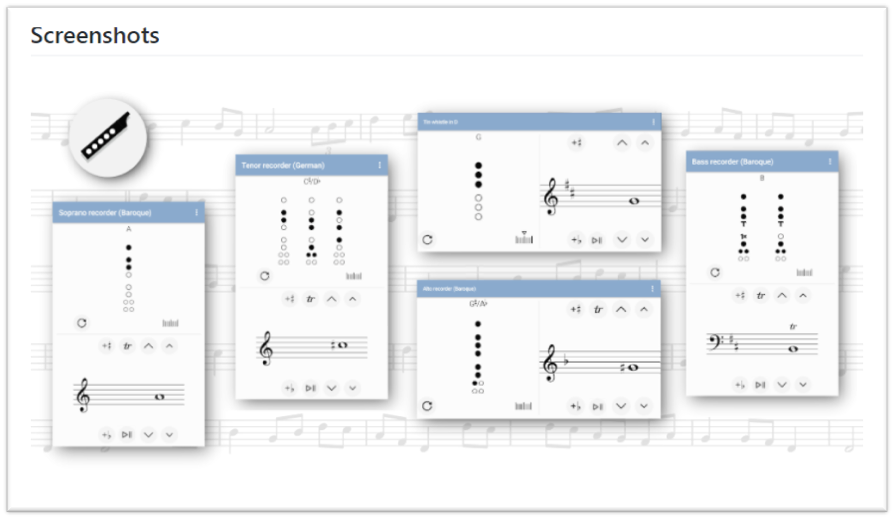
**4) 이미지로 된 악보를 인식하고 운지법으로 만드는 기능이 추가되면 좋을 것 같다. 오픈소스화 하여 다른 악기도 만들어지면 좋을 것 같다.**

오픈소스의 특성상 충분히 개발될 여지가 있는 부분이지만, MVP(Minimum Viable Product)에는 포함을 시키지는 않을 것이다.

**6. 참고 프로젝트**

**1) thalman / PlayRecorder**

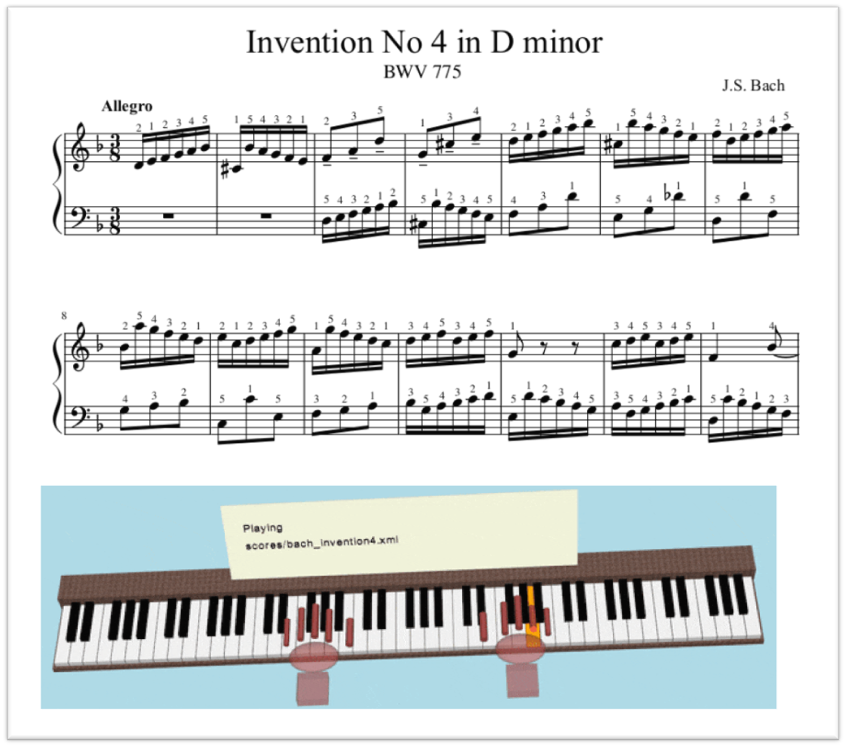
Link : <https://github.com/thalman/PlayRecorder> (오픈소스)



여러 음들에 대해 리코더의 운지법을 이미지로 나타내주는 오픈소스 프로젝트이다.

**2) marcomusy / pianoplayer**

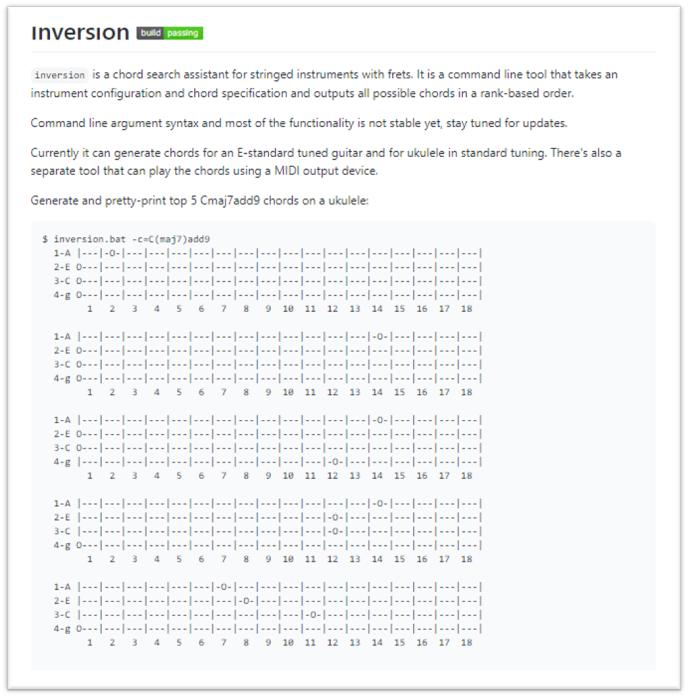
Link : <https://github.com/marcomusy/pianoplayer> (오픈소스)



피아노에서 악보를 통해 운지법을 추천해주는 오픈소스 프로젝트이다. 손의 불필요한 움직임을 피하는 노력을 최소화하는데 중점을 두고 알고리즘을 구현한 것이다.

**3) blacktaxi / inversion**

Link : <https://github.com/blacktaxi/inversion> (오픈소스)



우크렐레에서 몇개의 일부 코드들에 대한 운지법을 나타내주는 프로젝트이다.