

Weekly diary 3주차(2024.04.08 ~ 2024.04.14)

팀 푸바오

팀장 임베디드시스템공학과 201901752 서

정인

팀원 임베디드시스템공학과 201701726 권

오찬

팀원 임베디드시스템공학과 201901747 류

제현

팀원 임베디드시스템공학과 202001697 박

성빈

[작업 기록]

- 2024.04.08 16:30 ~ 20:30 (4시간)
- 2024.04.11 10:00 ~ 12:00 (2시간)

총 회의 및 미팅시간: 6시간

회의 내용

4월 8일 월요일, 과사 방문하여 부품 도착 여부 확인해 보았으나 아직 도착하지 않았다.

주문은 지난주에 완료되었기 때문에 이번주 내로 도착할 것이라 하여 도착하는 시기에 맞춰 미팅 일정 잡고 작업을 시작할 계획이다.

부품이 금요일에 도착하였으나 팀원들 간의 소통 오류가 있었고 일정도 맞지 않아 다음주 월요일 정기 회의 때 물품을 수령하여 본격적으로 개발을 진행할 예정이다.

우선 지난 주 환경 설정 도중 발생한 문제로 인해 진행하지 못했던 UI공부와 미디어파이프 테스트를 진행하였고 압력 감지 코드 자료조사, 도면 설계를 진행하였다.

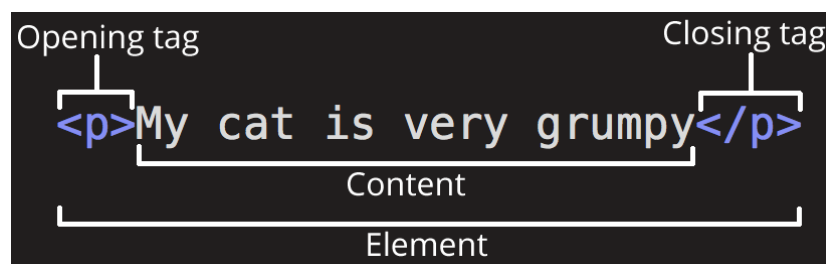
UI 공부

지난주 계획하였다가 실행하지 못했던 UI 제작을 위한 마크업 언어 공부를 실시하였다. 아래는 팀원들이 필요한 내용이 있을 때 바로바로 볼 수 있도록 일부 내용을 요약해둔 것이다.

- HTML

HTML (Hypertext Markup Language,하이퍼텍스트 마크업 언어)은 웹페이지가 어떻게 구조화되어있는지 브라우저로 하여금 알 수 있도록 하는 마크업 언어라고 한다.

마크업 언어는 태그 등을 이용하여 문서나 데이터의 구조를 명시하는 언어의 종류로, 데이터를 기술하는 정도로만 사용되기에 프로그래밍 언어와는 구별된다고 한다.(출처 : 위키백과, 마크업 언어)



HTML 요소(element)의 구조

HTML은 여러 element들이 모여서 이루어진다. 위 사진처럼 My cat is very grumpy라는 문장이 있다면 이 텍스트 문장은Content라고 부르며 태그로 Content를 감싸 element가 된다.

태그는 HTML이 작용하는 범위나 문서의 속성, element의 범위 등을 지정하고 구분하기 위해 사용된다. 여는 태그와 닫는 태그로 구성되며 태그를 여는 부분부터 닫는 부분까지가 그 태그의 범위이다.

```
<HTML>
  <HEAD>
    <TITLE>Top HTML Tags</TITLE>
  </HEAD>
  <BODY>
    HTML is swell.
    Life is good.
    <H3>Here's the big picture</H3>
    <IMG src="yourfile.gif">
      <UL>Make me an unordered list.
        <LI>One programmer</LI>
```

```
        <LI>Ten SDKs</LI>
        <LI>Great Internet Apps</LI>
    </UL>

</BODY>
</HTML>
```

HTML 태그를 사용하여 간단한 웹페이지를 빌드해보았다.

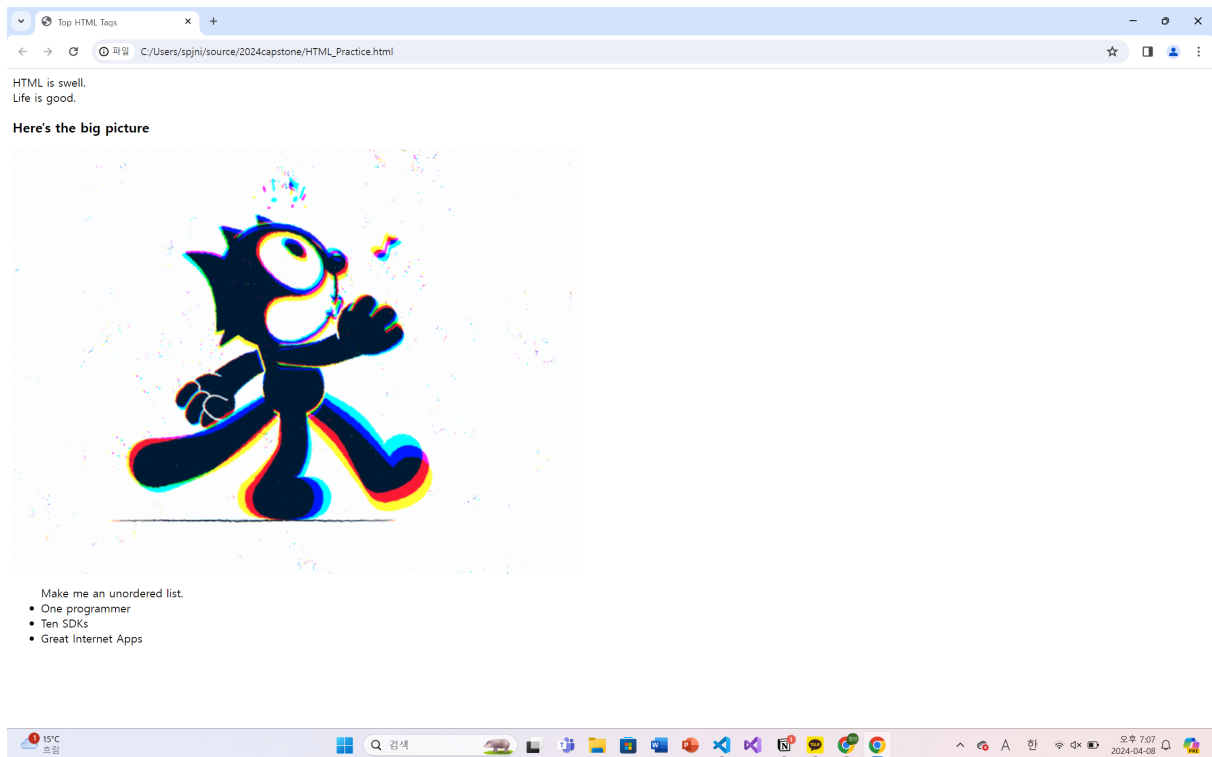
```
<HTML>
  <HEAD>
    <TITLE>Top HTML Tags</TITLE>
  </HEAD>
</HTML>
```

```
<BODY>
  HTML is swell.
  Life is good.
</BODY>
```

```
<H3>Here's the big picture</H3>
```

```
<IMG src="singingcat.gif">
```

```
<UL>Make me an unordered list.
<LI>One programmer</LI>
<LI>Ten SDKs</LI>
<LI>Great Internet Apps</LI>
</UL>
```



전체 코드 실행 화면.

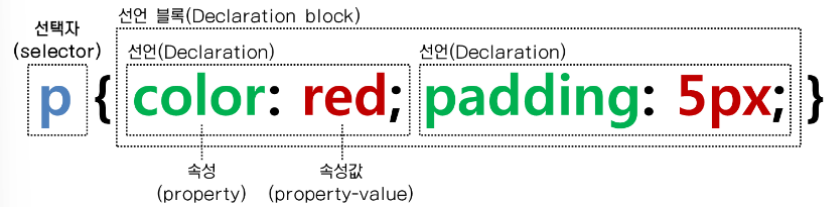
- CSS

CSS(Cascading Style Sheets)는 콘텐츠의 글꼴, 색상, 크기 및 간격을 변경하거나, 여러 열로 분할하거나, 애니메이션 및 기타 장식 기능을 추가하는 등 웹 페이지의 스타일을 지정하고 레이아웃하는 데 사용된다.

CSS를 사용하여 브라우저에서 HTML 요소가 표시되는 방식을 정확하게 제어하여 원하는 디자인을 사용하여 마크업을 표시할 수 있다.

CSS는 규칙 기반 언어로, 웹 페이지의 특정 요소 또는 요소 그룹에 적용해야 하는 스타일 그룹을 지정하여 규칙을 정의한다.

Rule Set (Rule)은 HTML 페이지 안의 특정 요소들을 어떻게 렌더링할 것인지 브라우저에게 알려주는 CSS 문장이다. 스타일 규칙이라고도 불리는 이 문장은 스타일에 관한 규칙들을 집합처럼 나타낸다.



출처: [CSS: 선택자\(Selector\) 이해 \(nexttree.co.kr\)](#).

왼쪽 중괄호가 나오기 전의 모든 부분은 선택자이며, 선택자는 Rule Set의 영향을 받는 HTML 페이지 안의 특정 요소들을 선택해서 선언 블록의 내용을 적용시켜준다.

중괄호 안에는 하나 이상의 선언이 있으며, 속성 및 값 쌍의 형태를 취한다. 콜론 앞에 속성을 지정하고 콜론 뒤에 속성 값을 지정한다.

HTML에 링크하려면 <head>안쪽에 다음 줄을 추가한다.

```
<link rel="stylesheet" href="styles.css" />
```

<link> 태그의 rel 속성은 현재 문서와 외부 리소스 사이의 연관 관계를 명시한다.

rel 속성은 <link> 요소에 반드시 명시되어야한다.

<link> 태그의 href 속성은 링크된 외부 리소스의 URL를 명시한다.

css파일에 다음을 추가한다.

```
h1 {
  color: red;
}
```

모든 단락과 모든 목록 항목을 녹색으로 만들려면 규칙은 다음과 같다

p → 모든 단락, li → 모든 목록 항목

```
p,
li {
  color: green;
}
```

다른 요소를 변경하지 않고 요소의 하위 집합을 선택하려면 클래스를 대상으로 지정 가능

```
.special {  
    color: orange;  
    font-weight: bold;  
}
```

MediaPipe 테스트

```
import mediapipe as mp  
import cv2  
  
mp_pose = mp.solutions.pose  
pose = mp_pose.Pose()
```

```
mp_drawing = mp.solutions.drawing_utils  
cap = cv2.VideoCapture(0)  
while cap.isOpened():  
    ret, frame = cap.read()  
    rgb_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)  
    result = pose.process(rgb_frame)  
    annotated_frame = frame.copy()  
    mp_drawing.draw_landmarks(annotated_frame, result.pose_landmarks)  
    cv2.imshow('Pose Estimation', annotated_frame)  
  
    if cv2.waitKey(5) & 0xFF == 27:  
        break  
cap.release()  
cv2.destroyAllWindows()
```

노트북에는 보통 기본적으로 내장 카메라가 존재하며 내장 카메라의 인덱스는 일반적으로 0을 배정받는다. `cap = cv2.VideoCapture(0)`을 통해 노트북의 내장 카메라와 연결하여 내장 카메라가 촬영하는 장면에 랜드마크 정보를 입혀 화면에 표시해줘야 한다.

하지만 코드를 실행하면 다음 에러가 뜬다.

```
[ WARN:0@2788.406] global cap_v4l.cpp:997 open
VIDEOIO(V4L2:/dev/video0): can't open camera by index
```

```
[ERROR:0@2788.406] global obsensor_uvc_stream_channel.cpp:159
getStreamChannelGroup Camera index out of range
```

노트북에 있는 기본 카메라를 인덱스 0번으로 연결할 수 없어 발생하는 오류인데, 기본 카메라의 인덱스를 찾아보려 했지만 실패했다.

조사 결과 WSL 가상환경에서는 기본적으로 노트북 내장 카메라나 USB 연결 장비를 사용할 수 없으며, 사용을 위해서는 프로그램 설치와 추가 설정이 필요하다고 한다.

usbipd 등 필요한 프로그램들을 설치하고 커널 설정 등 필요한 설정의 세팅을 시도해 노트북 웹캠을 wsl 환경에 연결하였지만 여전히 카메라를 읽어오지 못했다. 제품은 라즈베리파이 보드에 올라가는 별개의 코드와 별도의 깊이 카메라를 통해 작동하기 때문에 wsl환경과 노트북 웹캠의 호환 문제가 큰 이슈는 아니라고 생각하지만 추가적인 조사와 지속적인 설정 시도로 문제를 해결해보도록 하겠다.

압력 감지 자료 조사

<https://github.com/bogde/HX711/tree/master/src>

<https://bota.tistory.com/1629>

<https://blog.danggun.net/7562>

[http://lhdangerous.godohosting.com/wiki/index.php/로드셀_HX711_\(Load_cell_HX711\)](http://lhdangerous.godohosting.com/wiki/index.php/로드셀_HX711_(Load_cell_HX711))

```

#include "HX711.h"

const int LOADCELL_DOUT_PIN = 2;
const int LOADCELL_SCK_PIN = 3;

HX711 scale;

void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  scale.begin(LOADCELL_DOUT_PIN, LOADCELL_SCK_PIN);

  scale.set_scale(-7000);
  scale.tare();

  Serial.println("5초안에 2kg짜리 물건을 올려주세요!");
  delay(5000);
  Serial.println(scale.get_units()*0.453592, 1);

  float factor = -7000;

  while(true){
    scale.set_scale(factor);
    float weight = scale.get_units()*-0.453592;

    Serial.println(weight);
    //측정한 값이 내가 원하는 값에 도달하면 캘리브레이션이 완료!
    if(weight <= 2.0){
      break;
    }else{
      factor -= 100;
    }
  }
  Serial.print("보정값=");
  Serial.println(factor);
}

void loop() {
  // put your main code here, to run repeatedly:

```

캘리브레이션(영점 조절) 코드

Serial.begin(115200) → 데이터의 전송 속도를 장치끼리 일치시키는 것 115200bps를 의미

scale.begin() → 무게 측정할 핀 지정하는 거 같음

scale.set_scale (숫자) → 캘리브레이션을 통해 얻은 숫자를 괄호 안에 넣는다.

scale.tare() → 영점 잡는 것, 현재 측정값을 0으로 둔다.

serial.println (text) → text를 쓴 후 끝에 줄바꿈 하여 프린트 해주는 효과

0.453592를 곱하는 이유: 무게 단위가 달라서

2kg을 0으로 바꾸면 아무것도 없는 상태에서도 가능

우리 팀은 로드셀을 8개를 사용하므로 각각 로드셀의 성능 차이에 따라 캘리브레이션에 사용되는 값이 다를 수도 있으므로 각자 값이 혼동되지 않도록 변수 이름을 다음과 같이 미리 정해두었다.

leftfoot_front_cali → 왼발 전족, 중족의 로드셀 연결한 hx711 캘리브레이션에 사용하는 값

leftfoot_back_cali → 왼발 후족 2개의 로드셀 연결한 hx711 캘리브레이션에 사용하는 값

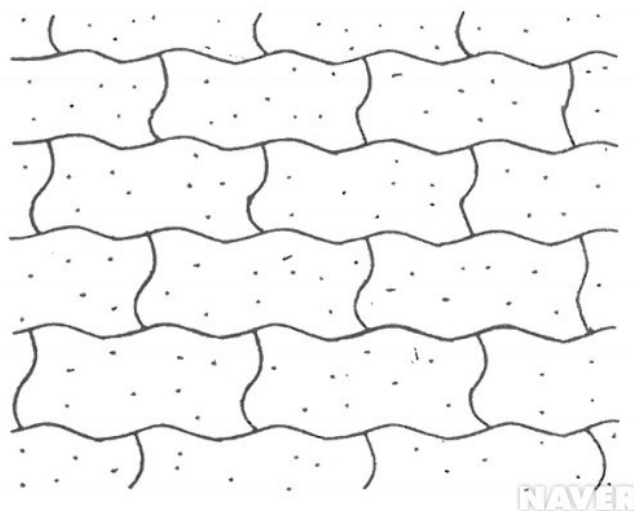
rightfoot_front_cali → 오른발 전족, 중족의 로드셀 연결한 hx711 캘리브레이션에 사용하는 값

rightfoot_back_cali → 오른발 후족 2개의 로드셀 연결한 hx711 캘리브레이션에 사용하는 값

도면 설계

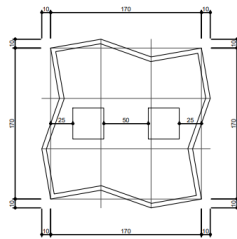
라이노(Rhino)를 이용하여 3d 모델링을 해서 도면을 작성했다.

큰 판에 작은 발판을 넣어서 사용자가 작은 발판을 밟아 압력을 감지하도록 설계했으며, 보도블록의 인터로킹 구조로 발판을 만들어 맞물리는 면적을 더 키워 마찰력을 높였다.

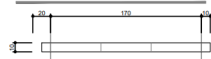


보드와 전선이 통과할 수 있도록 밑부분을 수정하고 stl 파일로 변환하여 메이커스페이스를 이용해 외관을 제작할 계획이다.

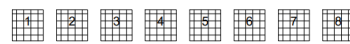
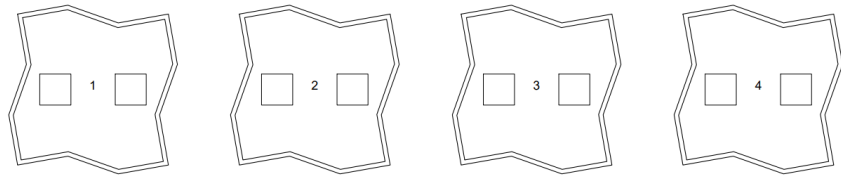
평면도-작은발판



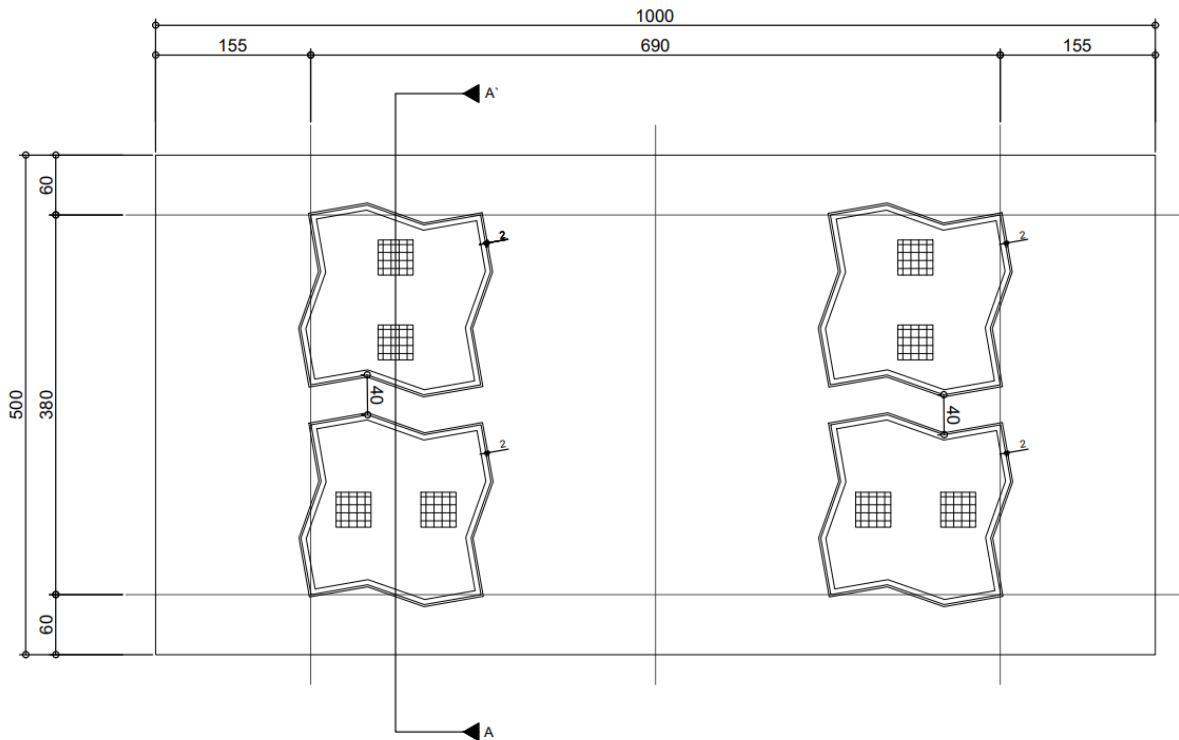
단면도-작은발판



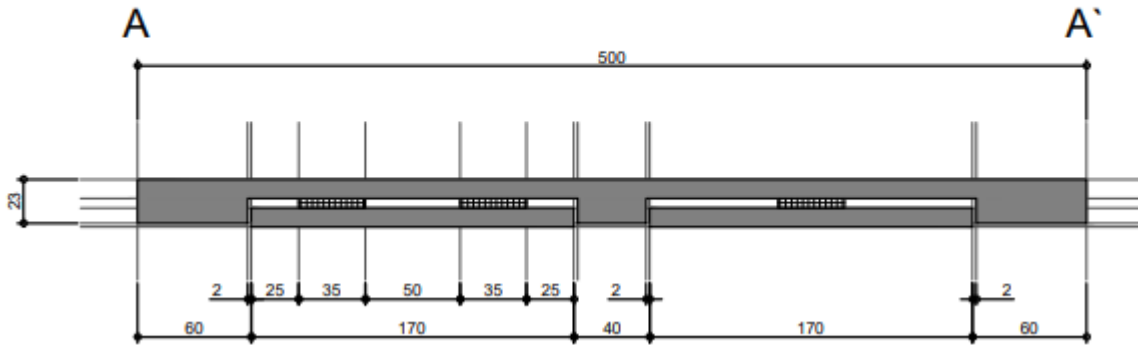
로드셀



평면도-큰판



단면도-큰판



향후 계획

부품 도착에 맞춰서 일정을 잡은 뒤 모여서 회로 구성 등의 작업을 하려 했지만 수요일까지도 부품이 도착하지 않았다. 금요일에 부품이 도착하였으나 가족 행사, 감기 등의 이유로 대부분의 팀원이 금~일요일에는 미팅에 참여할 수 없을 거 같아 목요일에 만나 활동을 진행하였으며 각자 개발할 부분에 대하여 시간 날 때 마다 공부한 후 다음주 월요일 정기 모임 때 물품을 수령하여 본격적으로 작업을 시작할 예정이다.

추가적으로 RTX 4060 노트북 부팅이 정상적으로 되지 않고 계속해서 재부팅이 반복되는 문제가 발생하여 A/S를 요청해야 할 것 같다. 다음주 평일에 A/S를 요청하여 수리를 받는다 하더라도 다음주 작업에는 사용하지 못할 것 같다.