

# 밥친구 곰돌이 —

부모의 설득 부담을 줄이고  
아이의 자발적 식사를 유도하는  
혁신적 육아보조장난감

대화형 행동 강화로 부모-아이 갈등 완화  
미디어 의존도 감소 목표

발표팀 소니소니손





# 팀 소개



팀 구성 0: 소니소니손  
(36개월 딸 이름 빌려줌, 최소니)



팀 구성 1: 팀장 최수길(바인드소프트 차장)  
프로그래밍 및 기타 작업



팀 구성 2: 팀원 손지영(업사이클링 자)  
프로젝트 매니저



메이커톤 특성상 짧은 개발 기간에 효율적 협업으로  
시제품 완성도 향상



협업 방식: 전문성 기반 역할 분담과 긴밀한 피드백  
루프

GitHub: [https://github.com/freshmea/ict\\_makerton\\_2025.git](https://github.com/freshmea/ict_makerton_2025.git)

# 개발 배경



**프로젝트명:** 밥친구 곰돌이



**개발 배경:** 세 살 딸 소니의 지속적 식사 거부 경험에서 출발



**목적:** 식사 시간의 부정적 경험 감소 및 부모-아이의 긍정적 상호작용 증대



**솔루션:** 대화형 도우미 장난감으로 아이의 흥미 유발, 미디어 의존형 접근과 차별화



**팀:** 소니소니손 — 팀장: 최수길 (프로그래밍), 팀원: 손지영(제작)



**요약:** 실사용 문제(세 살 식사 거부) 해결을 목표로 한 감성적·혁신적 제품



# 육아의 현실

"세 살 딸 소니, 밥 먹이기가 너무 힘들어요 ㅠㅠ"

밥 먹기 거부

식사가 재미없어서 안 먹으려  
함

끝없는 협상

"이거 먹으면 다음 장 읽어줄  
게"

훈육의 어려움

식탁 위 머리싸움의 연속



# [영상자료1] 이럴때 어떻게 ? 비위를 맞춰야 먹는 아이



[영상자료2] 이럴때 어떻게 ? 일상 중에도 밥을 잘 먹는 것이 주저





# 기존 해결방법의 한계

1

유튜브 보여주기

쉽지만 미디어 의존도 증가 우려



2

배고플 때까지 굶기기

정석이지만 부모 마음이 힘들



3

메이커의 해결책

흥미로운 식사 훈육 도구 개발!



# 밥친구 곰돌이란?

화면 의존 없이 아이의 자발적 식사를 유도하고, 부모의 반복 설득 부담을 줄이는 대화형 행동강화 디바이스



즉시 보상

한 입 = 작은 보상 체감



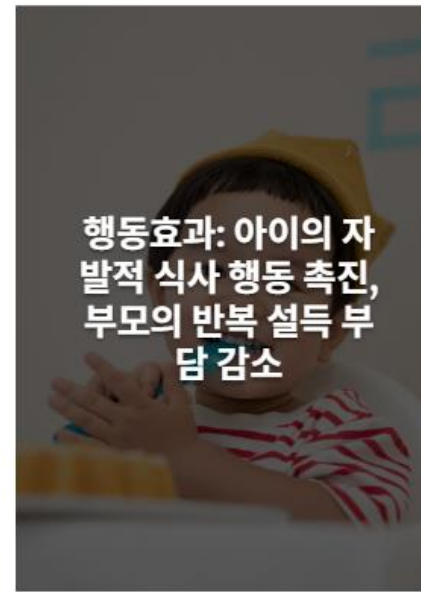
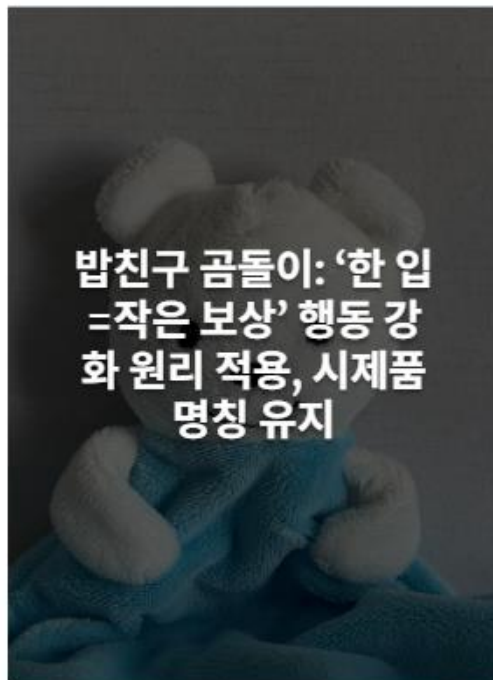
동작 피드백

팔 흔들기, 표정 변화



시각적 진행률

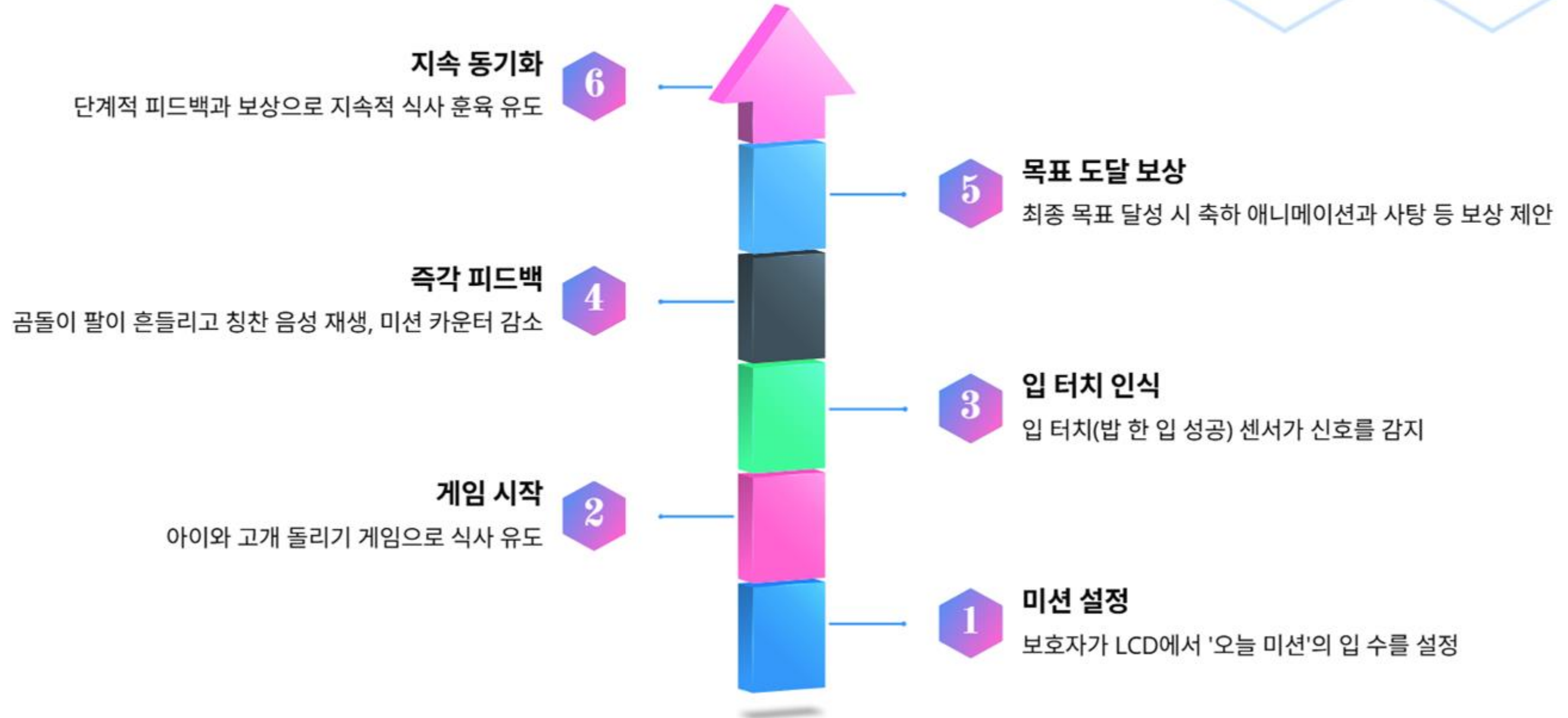
LCD 진행바와 칭찬 문구

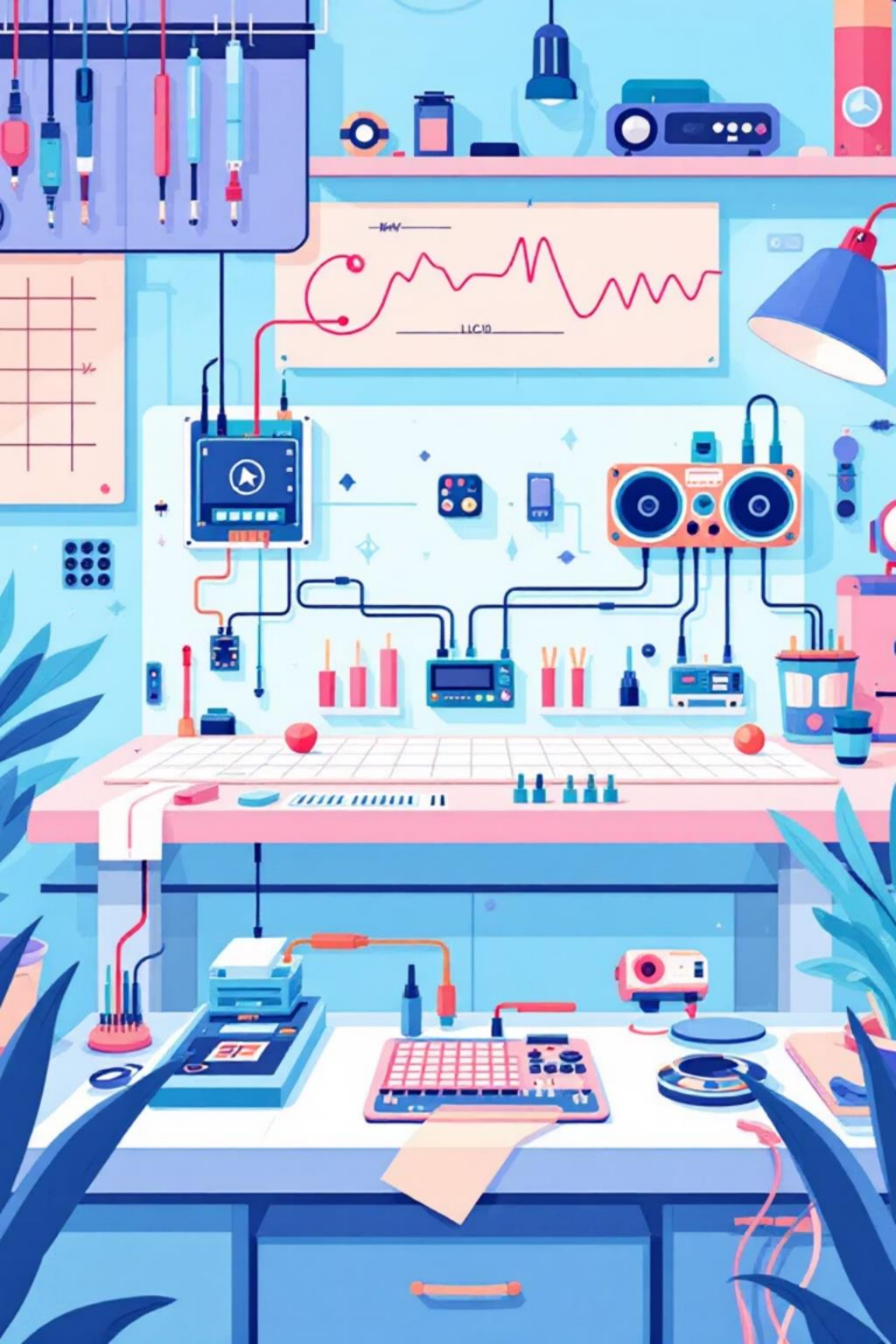




# 시제품 작동 시나리오

보호자가 '오늘 미션' 설정 → 아이의 입 터치 인식 → 보상으로 지속적 식사 동기 부여





# 핵심 구성품

포인트!

## 우리 아이 미래를 위한 업사이클링 실천 버전

종이 박스 활용한 곰돌이 하우스

### 하드웨어

- 서보모터 2개 (양팔 동작)
- 입 터치센서 (한입 확인)
- LCD/OLED (진행바 표시)
- 네오픽셀 링 (조명 효과)
- 부저/스피커 (칭찬 보이스)

### 소프트웨어

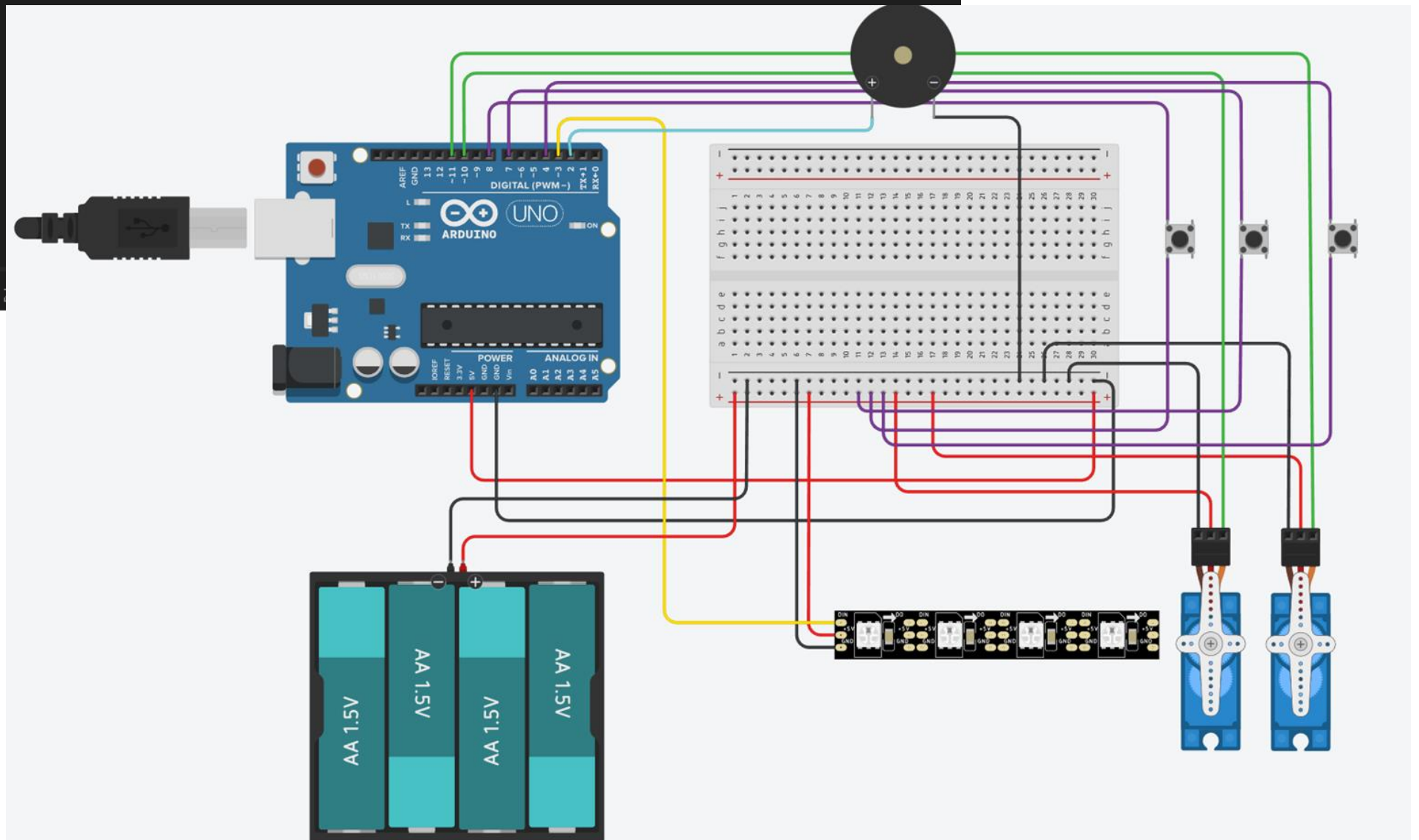
- 아두이노 MCU
- 블루투스 모듈 (앱 연동)
- 5V 전원 어댑터
- 안전 퓨즈/절연 처리

# 하드웨어 구성 및 결선

```
public:
    // 생성자
    SoneeBot(int s1Pin = 10, int s2Pin = 11, int neoPin = 3, int neoCount = 8,
            int t1Pin = 8, int t2Pin = 7, int t3Pin = 4, int buzPin = 2);

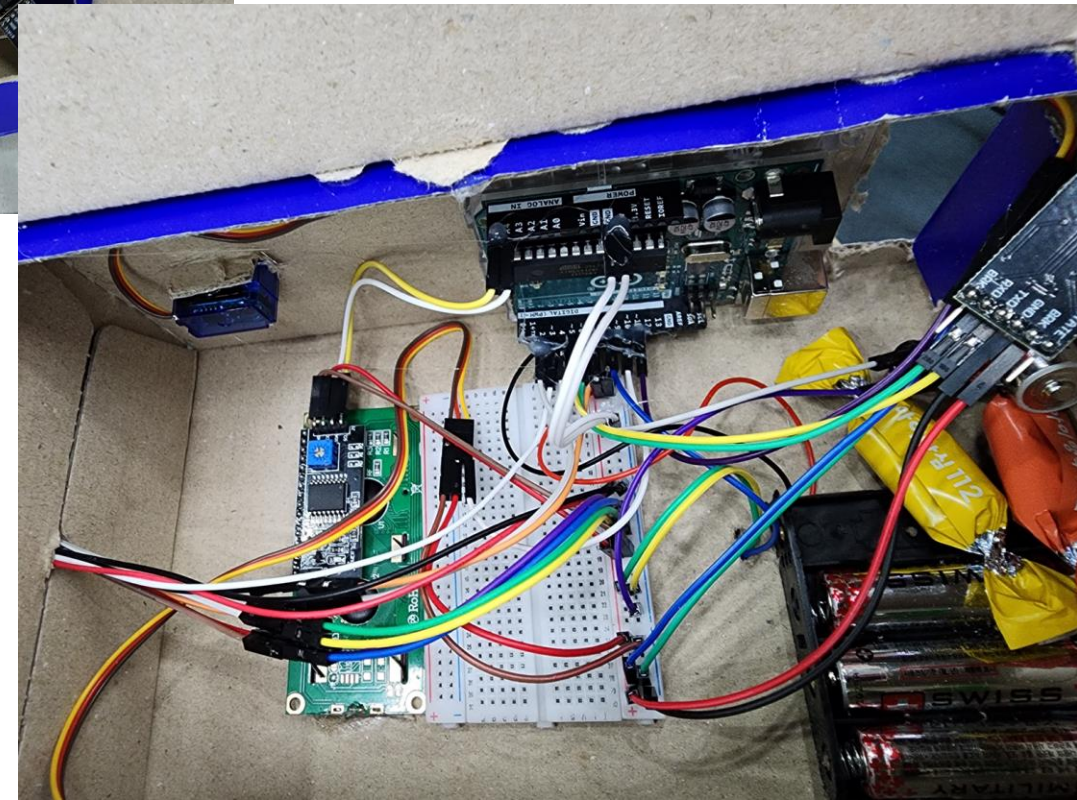
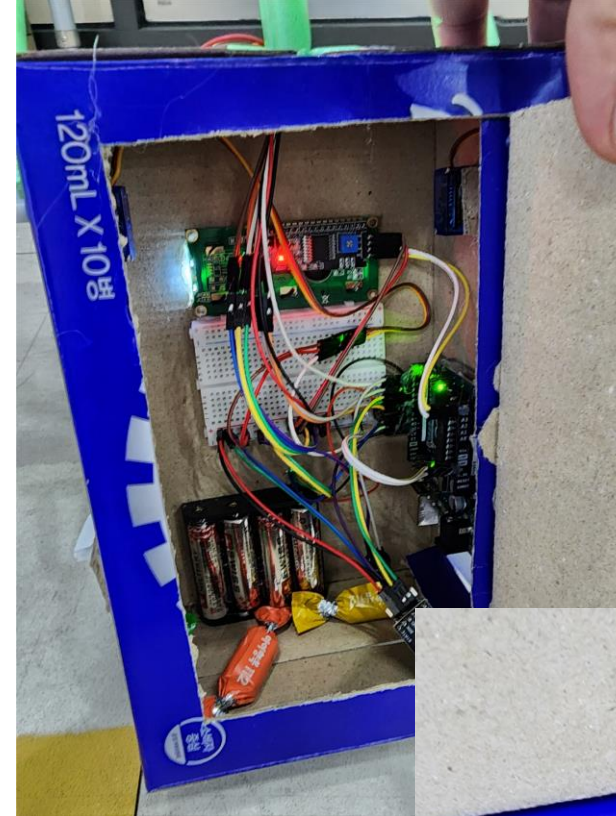
    // 소멸자
    ~SoneeBot();

    // 초기화 함수
    void init();
    freshmea [6 hours ago] • 7/
```





# 시제품 사진



# 시연 영상



자 아이가 한입을 먹습  
니다. (사실 게임 버튼  
입니다.)



# 핵심 코드 및 파일 구성

프로젝트 깃허브 주소(공개) : [https://github.com/freshmea/ict\\_makerton\\_2025.git](https://github.com/freshmea/ict_makerton_2025.git)

```
You, 7시간 전 | 1 author (You)
1  #include "SoneeBot.hpp"
2
3  // SoneeBot 객체 생성 (기본 핀 설정 사용)
4  SoneeBot robot;
5
6  void setup()
7  {
8      // SoneeBot 초기화 - 모든 장치 초기화 포함
9      robot.init();
10
11     // 초기화 완료 후 테스트 실행
12     // robot.testAllDevices();
13 }
14
15 void loop()
16 {
17     // 터치 상태 업데이트 (매 루프마다 호출 필요)
18     robot.updateTouchStates();
19
20     // 상태 업데이트 (터치 시간 포함)
21     // robot.updateStatusDisplay();
22
23     robot.updateMessage();
24 }
25 [EOF]
```

```
// 초기화 함수
void init();

// 서보 제어
void moveServo1(int angle);
void moveServo2(int angle);
void moveBothServos(int angle1, int angle2);
void moveServoSmooth(int servoNum, int angle, int delayMs = 15);

// 네오픽셀 제어
void setPixelColor(int pixel, uint8_t r, uint8_t g, uint8_t b);
void fillColor(uint8_t r, uint8_t g, uint8_t b);
void clearPixels();
void rainbowEffect();
void updateMissionPixels(int missionCount); // 미션 카운트에 따른 네오픽셀 업데이트 추가
void missionCompleteEffect();             // 미션 완료 효과 추가

// LCD 제어
void lcdPrint(int col, int row, String text);
void lcdClear();
void lcdBacklightOn();
void lcdBacklightOff();
void updateStatusDisplay();
void updateMessage(); // 새로운 함수 추가

// 센서 읽기 (기존 함수들 유지)
bool readTouch1();
bool readTouch2();
bool readTouch3();
```



# 코드 구성 실수 회고

```
// 새로운 터치 상태 함수들 추가
void updateTouchStates();
bool isTouchPressed(int touchNum);
bool isTouchReleased(int touchNum);
bool isTouchHeld(int touchNum);
unsigned long getTouchDuration(int touchNum);
bool isTouchLongPress(int touchNum, unsigned long threshold = 1000);

// 부저 제어
void beep(int duration = 200);
void beepPattern(int count = 1, int duration = 200, int interval = 200);

// 상태 정보
int getServo1Angle();
int getServo2Angle();
void printPinInfo();

// 테스트 함수들
void testAllDevices();
void testServos();
void testNeoPixels();
void testLCD();
void testSensors();
```

1. 서보가 360R과 180 이 있었는데, 처음에 360R 을 써서 왜 안 되는지 고민함.

1. 화면 갱신 주기가 너무 빨라서 화면이 흐리게 보임 -> 변수 변경을 확인한 다음에만 업데이트

1. 터치 센서가 high 가 디폴트이고 low 가 되었을 때가 반응 한 것을 거꾸로 생각해서 디버깅이 오래걸림

1. 양쪽 서보가 동시에 일정하게 움직이게 만드는게 어려웠음.  
-> 중간 경로를 계속 양쪽으로 업데이트 하는 코드로 수정

# 주요 기능



## 목표 설정

5~12입 범위에서 목표 입 수 설정 가능

## 게임 기능

게임을 진행하듯 밥을 먹일 수 있음

## 생동감 있는 조명

아이 눈에 즐거운 원색의 네오픽셀 브레스 효과



입 터치 센서가 **아이의 한 입 행동**을 감지



센서 신호로 곰돌이의 **팔 흔들림** 즉시 발생



LCD에 **미션 진행 상황**과 **칭찬 문구** 표시



효과음과 **음성 피드백**으로 즉각적 보상 제공



다감각 피드백은 **긍정적 강화** 기반, 식사 지속 유도



## 제품의 사회적 가치와 기대효과

가정 내 식사 갈등 완화와 건강한 성장 환경 조성의 실생활 혁신

밥친구 곱돌이로 아이  
와 부모 간 식사 갈등  
완화

미디어 과다 의존 억제  
및 자연스러운 관심 전  
환

부모의 반복 설득 부담  
경감 — 정서적 스트레  
스 감소

아동의 자발적 행동 유  
도로 긍정적 상호작용  
증진

건강한 성장 환경 조성  
— 일상형 솔루션의 실  
생활 적용성

실생활 혁신 기대 — 시  
간·정서적 비용 절감과  
행동 변화





## 향후 계획: 기능 확장 및 AI 맞춤형 적용

무선 연결 및 인식 기반 맞춤형 음성 제공을 통한 사용자 경험 고도화



- 1 Wi-Fi 모듈 탑재: 기존 보드를 Wi-Fi 지원 보드로 교체
- 2 카메라 추가: ESP32-CAM 도입으로 영상 캡처 가능
- 3 스피커 및 I2S 모듈: 고품질 음성 출력 구현
- 4 n8n 서버 연동: 워크플로우 기반 AI 기능 통합
- 5 상황 인식 기반 콘텐츠: 카메라센서로 현재 상황 판단 후 텍스트/음성 제공

★스피커 추가 시 엄마아빠 목소리, 유명 캐릭터 목소리 등 활용해 보상 멘트 가능  
★업사이클링 버전 뿐 아니라 아이의 애착인형에 부착시키는 키트 버전도 검토