

제18회 임베디드SW경진대회 개발완료보고서

[임베디드SW 주니어 메이커]

□ 개발 요약

팀 명	구해줘 일상
	
작품명	V-killer
작품설명 (요약)	<p>V-killer는 공공장소를 소독해 주는 자동 방역기입니다. 화장실이나 도서관에 부착해서 사스, 메르스, 코로나 등 각종 바이러스나 세균들에 노출된 장소를 무인으로 소독할 수 있습니다.</p>
시연동영상	https://youtu.be/vDn8u1mK1To

□ 개발 개요

○ 개발 작품 개요

- 최근 사회적으로 전염병 문제가 심각해지고 있는 상황입니다. 그 때문에 사람들의 활동이 매우 잦아들고 우리나라의 경제와 소비가 많이 뒤처지고 있는 상황입니다. 이런 전염성이 강한 바이러스를 퇴치하기 위해서는 예방과 방역 작업이 무엇보다 중요합니다. 특히 불특정 다수가 사용하는 공공시설의 화장실이나 학교 화장실 같은 곳은 자주 방역을 해야 하지만 소독약을 사람이 없는 상태에서 뿌려야 하기 때문에 자주 소독하기가 어렵습니다. 이런 상황들을 보고 저는 '공공시설에 자동으로 방역을 해주는 기계가 있으면 좋을 것 같다.' 라는 생각이 들었습니다. 그래서 저는 학교 화장실을 자동으로 방역해 주는 AI 자동 방역 시스템 V-killer를 만들고 싶다고 생각했습니다.

또한 "최근 코로나19 바이러스가 사람의 배설물에서도 검출된다는 연구 결과가 나오고 있다. 변기에 묻은 환자의 배설물이 손 등을 통해 눈, 코의 점막과 닿으면 감염될 수도 있다." 라는 보고 공공장소 소독의 중요성을 생각하게 되었습니다.

○ 개발 목표

- V-killer가 완성되어 우리 생활에 실제로 사용이 된다면 사람들이 방역하는 시간과 노동력이 훨씬 줄어들고 그 남은 시간들로 다른 업무들이나 활동을 효율적으로 진행할 수 있습니다. 그래서 우리 생활을 위협하는 바이러스 질병들이 잘 예방되는 것이 이 작품의 개발 목표입니다.

○ 개발 작품의 필요성

- 현재 바이러스 질병을 예방하기 위해서는 공공장소 소독이 정말 중요합니다. 하지만 소독약을 수시로 뿌리려면 사람이 아무도 없을 때를 기다려야하기 때문에 수동으로 관리하기에는 너무 큰 노력이 필요합니다. 그래서 무인으로 관리해주는 기계가 필요하다고 생각하였습니다.
- V-killer는 매 시각 정시에 소독을 진행합니다. 그리고 소독 상황을 관리자 스마트폰에 기록하기 때문에 많은 장소를 효율적으로 관리할 수 있습니다. 학교처럼 한 건물에 화장실이 여러개 있는 곳을 관리하기에 아주 적합합니다.

□ 개발 환경 설명

○ Hardware 구성

- [아두이노 보드]
코드를 넣고 연결된 부품을 조종하는 기능을 합니다.

- [초음파 센서]
거리를 측정하는 센서입니다. 소독 공간에 사람이 있는지 없는지를 파악해줍니다.

- [mg-996r 서보모터]
소독약을 뿌리기 위한 모터입니다. 일반 서보모터보다 더 강력한 힘을 가진 모델입니다.

- [쿨링 팬]
소독 후 약을 공기 중에 골고루 퍼트리기 위한 환기팬입니다.

- [LCD 모듈]
소독이 진행될 때 외부에서 사람이 들어오지 않도록 글자를 보여줍니다.

- [스피커, mp3모듈]
소독이 진행될 때 외부에서 사람이 들어오지 않도록 소리를 들려줍니다.

- [블루투스 모듈]
스마트폰 앱과 연결하기 위한 통신 장치입니다.

○ Hardware 기능 (제어 방법 등 서술)

- 정각이 되면 아두이노에서 초음파 센서로 사람이 있는지 없는지 감지합니다.
만약 사람이 있다면 사람이 나갈 때까지 기다리고 사람이 없으면
모터를 돌려 소독 액을 뿌리고 팬 쿨러로 소독 액을 골고루 퍼트려 줍니다.
또한 lcd와 mp3로 방역이 시작되었음을 알려줍니다.
방역이 시작되면 아두이노에서 신호를 보내 앱에 시간이 찍히게 됩니다.
또한 앱의 테스트 페이지에서 각 테스트에 대한 버튼을 누르게 된다면
아두이노에서 버튼에 따른 신호를 받아서 서보모터, lcd와 mp3, 초음파센서를
실행시킵니다.

○ Software 구성

- 앱 : 메인 페이지, 방역 기록 페이지, 테스트 페이지 3가지로 구성되어있습니다.



[메인 페이지]

방역 기록 페이지와 테스트 페이지로 이동할 수 있는 버튼이 있습니다. V-killer를 나타내는 로고와 이미지를 볼 수 있습니다.

[방역 기록 페이지]

아두이노에서 방역이 진행되면 날짜와 시간을 자동으로 기록합니다. 기록을 삭제할 수 있는 삭제 버튼과 이전 페이지로 돌아가는 확인 버튼이 있습니다. 기록들은 폰의 DB에 저장됩니다.

[테스트 페이지]

아두이노에 연결된 센서들이 고장나지 않고 잘 작동하는지 하나씩 테스트 할 수 있는 페이지입니다. 서보모터 테스트, 알림 테스트, 초음파 센서 테스트를 각각의 버튼으로 조종해볼 수 있습니다.

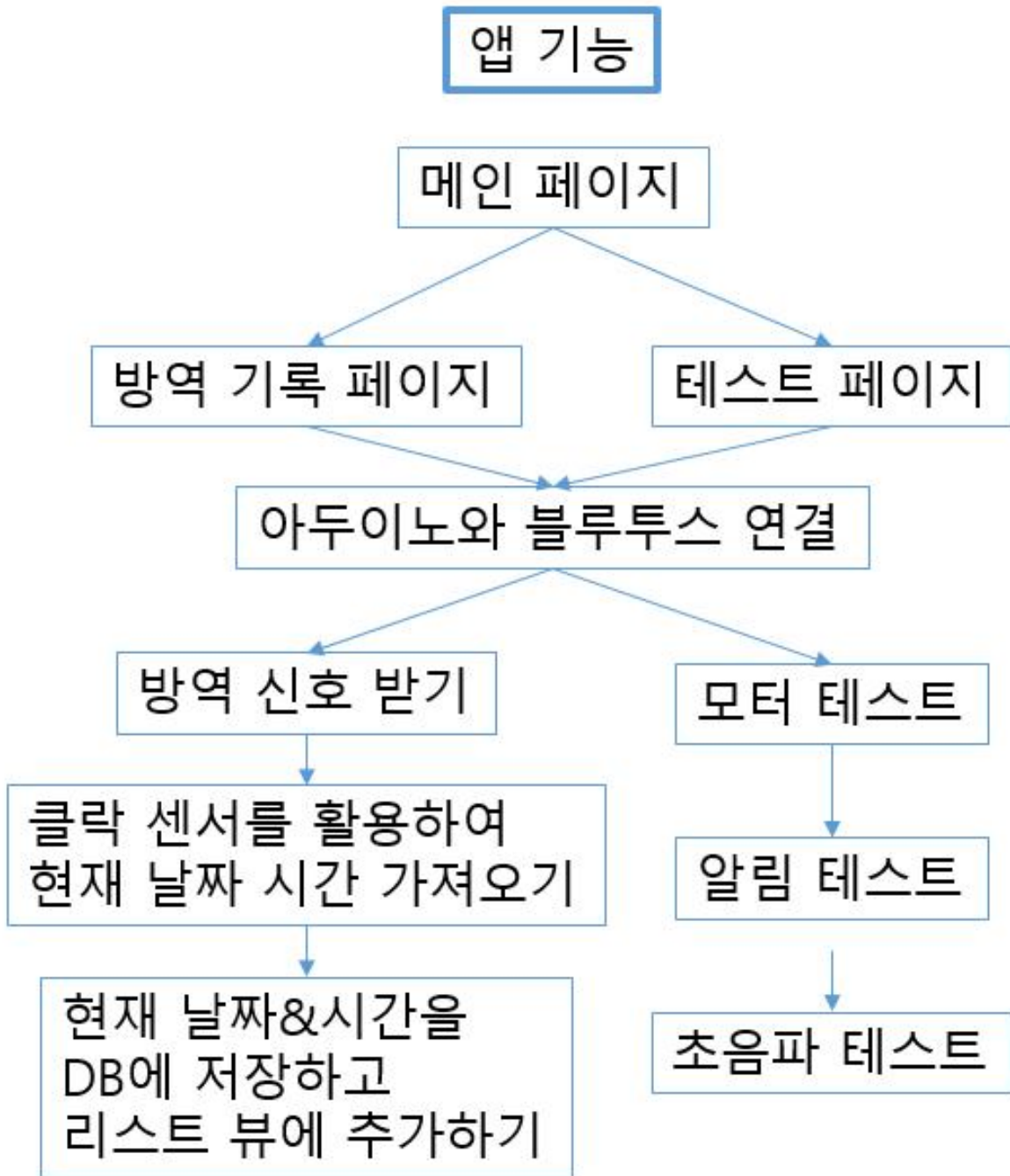
- 아두이노 : 방역기록 코드, 테스트 코드로 구성되어 있습니다.

방역기록 코드는 방역기록 페이지와 연결되어 있습니다.

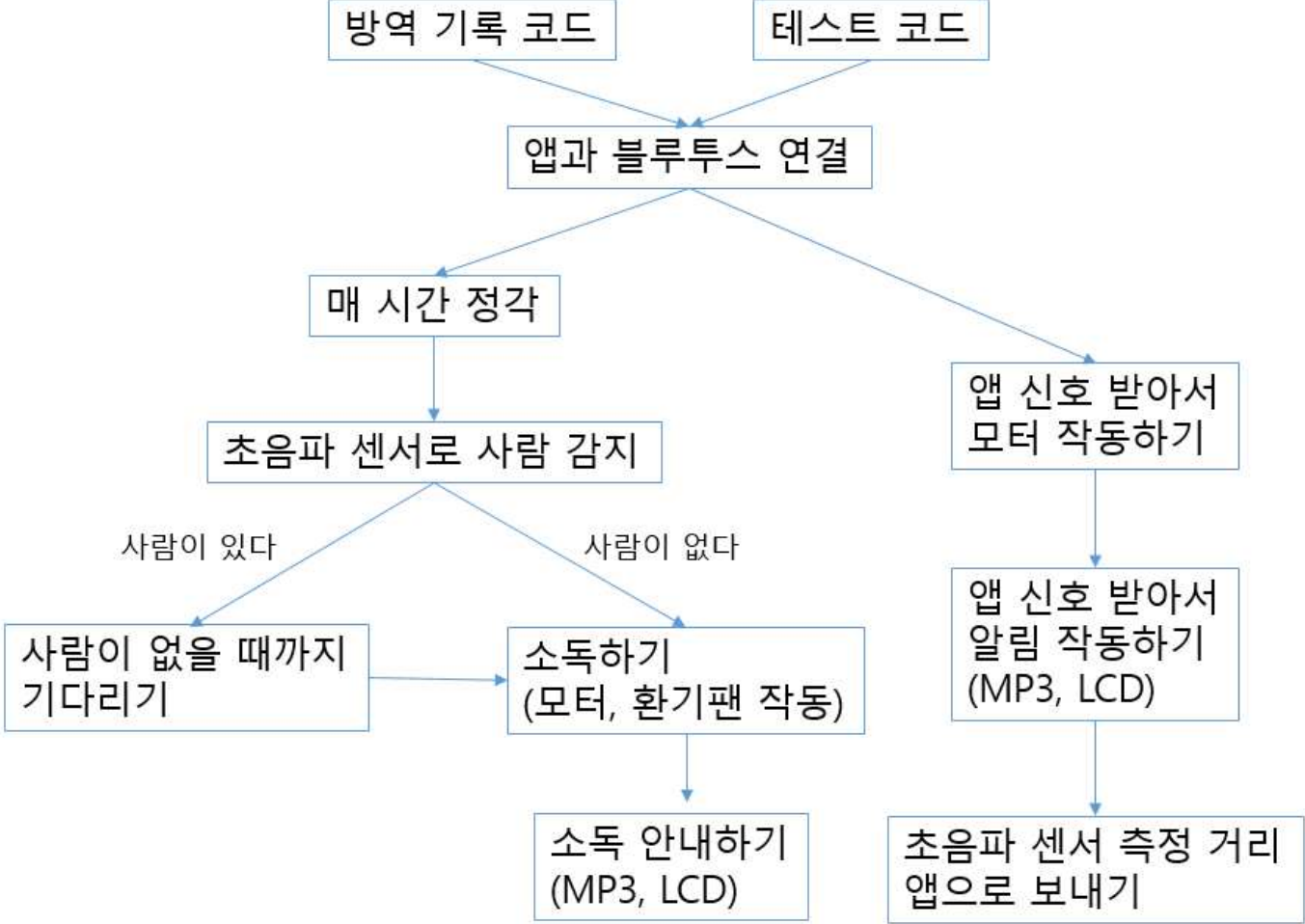
매 시각 정시가 되었을 때, 모터와 초음파센서, LCD 모듈, MP3 모듈, 환기팬을 작동시켜 소독을 진행하고 소독을 한 날짜와 시간을 블루투스 연결을 통해 앱으로 전송합니다.

테스트 코드는 테스트 페이지와 연결되어 있습니다.

앱의 각 버튼을 누르면 블루투스를 통해 신호가 아두이노로 전송되며 해당하는 센서를 작동시키는 코드가 들어있습니다.



아두이노 기능



○ Software 기능 (필요 시 알고리즘 설명 포함)

- 앱 기능

- 1) 아두이노와 통신할 수 있도록 블루투스 연결 기능이 있습니다.
- 2) 서보모터 테스트 : 모터의 각도를 0도와 90도로 조종하는 버튼이 있습니다.
- 3) 알림 테스트 : LCD와 MP3를 동시에 작동하는 버튼이 있습니다.
- 4) 초음파 테스트 : 아두이노에서 초음파 센서로 거리를 측정하여 블루투스를 통해 앱으로 값을 전송합니다. 그 값으로 사람이 있는지 없는지를 이미지로 표현해줍니다.
- 5) 방역 시간 기록 : 앱인벤터의 클릭을 사용하여 소독이 시작될 때 날짜, 시간을 구해 폰의 DB와 리스트 뷰에 저장합니다.
- 6) 방역 시간 삭제 : 필요 없는 정보를 관리자가 직접 삭제할 수 있는 기록 삭제 버튼이 있습니다.

- 아두이노 코드 기능 :

- 1) 앱과 통신할 수 있도록 블루투스 연결 코드가 있습니다.
- 2) 앱에서 모터 테스트 버튼을 누르면 블루투스를 통해 신호를 받습니다.
'a'를 입력 받았을 때 - 모터 90도로 움직이기
'b'를 입력 받았을 때 - 모터 0도로 움직이기
- 3) 앱에서 알림 확인 버튼을 누르면 블루투스를 통해 신호를 받습니다.
'c'를 입력 받았을 때 - mp3와 LCD 작동하기
- 4) 0.1초마다 한번씩 초음파 센서의 거리 값을 블루투스를 통해 앱으로 보냅니다.
- 5) 방역하기
 - > 매 시간 정각이 되면 초음파 센서를 작동시켜 사람이 있는지 없는지 검사합니다.
 - > 사람이 없으면 모터를 작동하여 소독액을 뿌립니다.
 - > 환기팬을 5초간 작동합니다.
 - > LCD 모듈과 MP3 모듈을 작동하여 알림을 시작합니다.
 - > 블루투스로 앱에 신호를 보내 앱에서 현재 날짜오 시간을 기록하게 합니다.

○ 프로그램 사용법 (Interface)

- 관리자는 스마트폰 앱을 다운로드 받습니다.
- 앱을 열고 아두이노 장치와 블루투스 연결을 합니다.
- 공공장소 화장실 등 소독이 필요한 곳에 V-killer를 설치합니다.
(모터와 환기팬은 안쪽에 MP3와 LCD는 문 입구에 설치합니다.)
- V-killer에 전기를 주면 (건전지나 콘센트 사용) 매 시각 소독이 진행됩니다.
- 앱의 방역 기록 페이지의 기록을 보며 소독이 잘 진행되는지 확인합니다.

○ 개발환경 (언어, Tool, 사용시스템 등)

- 아두이노 (C언어로 코드 작성)
- 앱인벤터 (블록형 언어로 코드 작성)

□ 개발 프로그램 설명

○ 파일 구성

- 앱 코드는 앱인벤터 페이지를 확인할 수 있는 aia 파일과 완성된 앱을 직접 다운로드 할 수 있는 apk 파일이 있습니다.
- 아두이노 코드는 방역 코드와 테스트 코드 2가지로 만들어져 있습니다.

○ 함수별 기능

- 아두이노에는 setup()과 loop() 2가지 함수로 크게 나누어져 있습니다. setup()에는 초기 설정을 위한 코드가 들어있고 loop()에는 작동을 위한 상세 코드가 들어 있습니다.
- 앱에는 블루투스 기능, 화면이 열렸을 때 블록, 클릭 블록, 각 버튼 블록 등으로 나누어져 있습니다. 블루투스 코드는 처음 연결하는 코드와 값을 주고 받는 통신 코드로 이루어져 있습니다. 화면이 열렸을 때에는 DB에 저장되어 있던 값을 불러오는 코드가 있습니다. 클릭에는 날짜와 시간을 계산하는 코드와 0.1초마다 블루투스에서 값을 읽어오는 코드가 있습니다. 각 버튼에는 아두이노와 통신하기 위한 통신 코드와 각자의 역할을 진행하는 코드가 들어 있습니다.

○ 주요 함수의 흐름도

- setup()
 - > 씨리얼 통신 시작 코드 (통신 속도 9600)
 - > 블루투스 통신 시작 코드 (통신 속도 9600)
 - > MP3 모듈 연결 코드 (D3번)
 - > LCD 모듈 연결 코드 (밝기 설정)
 - > 모터 핀 연결 코드와 모터의 각도를 시작 시 0도로 맞추는 코드 (D5번)
 - > 초음파센서 핀 연결 코드 (D6번, D7번)
- loop()
 - > 블루투스와 연결을 유지하고 0.1초마다 값을 주고 받는 코드
 - > 초음파 센서의 거리 값을 측정하여 씨리얼 모니터에 출력하는 코드
 - > if문을 사용하여 앱에서 오는 값에 따라 아두이노 부품을 조종하는 코드

○ 기술적 차별성

- 앱과 아두이노를 연결하여 프로젝트의 완성도를 높였습니다.
- 아두이노에 필요한 라이브러리를 찾아 모두 추가하였습니다.
- 여러 개의 부품을 한꺼번에 작동시켜 전기가 부족한 것 같아 건전지를 2개로 추가하여 전기가 모자라지 않게 테스트를 완료하였습니다.

□ 개발 중 장애요인과 해결방안

- 서보모터가 너무 약해서 어떻게 소독액을 뿌릴지
 - 원래는 일반 서보모터를 사용해서 v-killer를 만들었는데 중간점검을 할 때 소독액이 잘 안나가서 서보모터를 좀더 썬 mg-996r서보모터로 바꾸게 되었습니다.

- lcd중 텍스트lcd와 이미지lcd중 무엇을 쓸지
 - 텍스트 lcd가 이미지 lcd보다 더 간편하게 설치가 가능하고 재생속도가 더 빨라서 텍스트 lcd를 고르게 되었습니다.

- 앱에 더 넣을 기능이 있을지
 - 원래는 앱에 메인 페이지와 방역기록 페이지 뿐이었는데 사용자의 입장을 생각해보니 '각 아두이노의 부품들을 테스트해보는 기능이 있으면 좋을 것 같다.' 라는 생각이 들어서 테스트 페이지를 더 추가하였습니다.

- 아두이노를 실행시키면 lcd와 mp3가 잘 안됐었는데 어떻게 할지
 - 항상 아두이노에 코드를 넣고 실행시키면 lcd는 빛이 새지 않고 mp3는 소리가 거의 안났었는데 배터리를 추가하여 더 많은 전기를 공급해서 더 진행을 할 수 있었습니다.

- 우드록으로 외형을 만들 때 서보모터의 날이 너무 짧아 어떻게 할지
 - 원래는 서보모터의 날만 밖으로 빼려 했는데 실제로 해보니 날이 너무 짧았습니다. 그래서 구멍을 좀 더 크게 뚫어서 서보모터의 앞부분 전체가 나오도록 우드록을 수정하였습니다.

- 분사기와 서보모터의 위치가 잘 안 맞아서 어떻게 해야 할지
 - 처음에는 서보모터 중간에 분사기의 분사부분이 있었는데 조금 수정해서 서보모터 날의 끝부분에 분사기가 있도록 했습니다. 또한 서보모터와 분사기를 테이프로 붙여서 분사기가 미끌어지는 것을 방지했습니다.

□ 개발결과물의 차별성

○ 자동 방역 기능

- 매 시간 자동으로 방역을 시작합니다.
- 초음파 센서가 사람이 있는지 없는지 검사합니다.
- 관리자 앱으로 방역 날짜와 시간을 자동 저장합니다.

○ 실제 상황 테스트

- 일반 서보모터보다 힘이 더 강력한 모터를 사용하여 실제로 소독 버튼이 눌러지는지 확인하였습니다.
- MP3 모듈과 LCD 모듈을 사용하여 소리와 시각 2가지 방법으로 사람들에게 소독중임을 알립니다.

○ 앱과 아두이노를 함께 사용하여 더 효율적으로 작동합니다.

- 앱의 테스트 페이지에서 아두이노의 부품 중 고장난 부분이 있는지 편하게 확인할 수 있습니다.
- 소독을 진행한 시간이 자동으로 앱에 저장되고 관리자가 수동으로도 기록을 관리할 수 있습니다.

□ 개발 일정

※ 실제 프로젝트 개발 일정 작성 / 1page 이내로 작성

No	내용	2020年											
		6月			7月			8月			9月		
1	아두이노 1차개발												
1-1	재료 준비												
1-2	부품 연결후 테스트												
1-3	알맞은 부품 추가												
2	앱 개발												
2-1	UI와 기능 구상												
2-2	앱+아두이노 연결코드 만들기												
2-3	UI수정												
2-4	테스트 페이지 코딩												
2-5	방역기록 페이지 코딩												
3	아두이노 2차개발												
3-1	아두이노 코딩하기												
3-2	아두이노 부품 테이핑												
3-3	외형 구상하기												
3-4	외형 만들기												
4	최종 완성하기												
4-1	실제 테스트 해보기												
4-2	불편한점 보완하기												

□ 팀 업무 분장

※ 프로젝트 개발 관련 팀원의 업무 분장 위주로 작성 / 1page 이내로 작성

No	구분	성명	참여인원의 업무 분장
1	팀장	김태관	<ul style="list-style-type: none">○ 아두이노 하드웨어 만들기○ 아두이노 코딩○ 앱인벤터 UI만들기○ 앱인벤터 코딩○ 앱, 하드웨어 디버깅○ 실제 테스트