|  |
| --- |
| **스마트 약병 IoT**  2016104102 고원빈  2015104173 박권수  2015100552 윤형선  **요 약**  하루가 다르게 고령화가 진행되고 있는 대한민국. 2017년 기준, 나이가 65세를 넘은 ‘노인'이 전체 인구의 14%를 넘어섰다. 이에 따라 약을 복용하는 인구도 크게 늘어난 것을 알 수 있다. 본 연구는 약 복용을 관리하고 지원하는 플랫폼에 관한 것이다. 본 연구는 약병에 센서를 장착하여, 정기적 통신을 통해 데이터를 송수신하는 하드웨어부와, 여기서 얻은 데이터를 가공하여 사용자에게 각종 정보를 보기 쉽게 GUI로 제공하는 소프트웨어부로 구성된다. 본 연구에 따르면, 사용자는 복용해야 될 약을 잊지 않고 먹을 수 있고, 약에 관한 정보를 얻을 수 있으며, 약병을 안전한 곳에 보관할 수 있어, 약의 효능을 의료진이 제시한 최대한으로 누릴 수 있다. |

**1. 서론**

**1.1. 연구배경**

노령인구가 크게 증가하면서 이들이 병원을 찾는 횟수도 크게 늘었는데, 이들을 관리해 줄 수 있는 의료인력은 크게 늘어나지 않았다. 의사들은 환자에게 약을 처방하고, 복약지도대로의 약 복용과 보관을 온전히 복약자에게 맡길 수 밖에 없는 상황이다. 그러나 복약자가 약 복용 시간을 놓칠때도 있고, 냉장보관이나 직사광선을 피해 보관하지 않는 경우가 있다.

시장에는 이와 같은 문제를 해결하기 위해, 다양한 제품을 내놓고 있는데, 대부분 한가지 기능만 탑재하고 있으며, 주로 약의 복용 시간 준수 확인 및 정량 복용 확인을 중심으로 개발되고 있다.

**1.2. 연구목표**

본 연구는 의료진이 제시한 약의 복용지도를 지켜, 의료진이 의도한 약의 최대 효과를 끌어낼 수 있게 지원한다. 아두이노와 각종 환경 수집 센서를 통해 데이터를 수집하고, 이를 기반으로 복약자(이하 사용자)에게 의료진이 의도한 최적의 복용법을 제공한다.

첫째, 약을 정해진 시간에 정해진 양만큼 복용할 수 있도록 복약자에게 정보를 제공한다. 복약자가 최초 약을 처방 받고, 복용시간과 복용량을 소프트웨어에 등록하면, 정해진 시간에 소프트웨어와 하드웨어에 알림을 발생시켜 잊지 않고 복용을 돕는다.

둘째, 약을 처방 받고, 약의 정보를 소프트웨어에 등록하면, 건강보험심사평가원에서 제공하는 공공데이터를 활용해 해당 약에 대한 상세 효능을 소프트웨어를 통해 제공한다.

셋째, 해당 약을 최적의 장소에 보관할 수 있게끔, 사용자에게 알려준다. 하드웨어의 센서와 아두이노를 통해 현재 약이 위치한 곳의 환경 데이터를 수집하고, 이를 약의 보관기준과 비교하여 적합한지 판단한다. 만약 부적합할 경우, 소프트웨어에서 최적의 예시 장소를 안내하여주고, 이동 시킬 것을 권유한다.

**2. 관련연구**

**2.1. 사물 인터넷**

사물인터넷(IoT)은 Internet of Things의 줄임말로, 사물들 간의 연결을 통해 상호적으로 정보 공유를 하고, 이를 통해 더 나은 서비스를 제공하는 것을 목표로 한다. 우리에게 익숙한 스마트홈(스마트 플러그, 홈CCTV 등)부터 시작해, 자율주행자동차, 스마트 그리드(전력 수요 관리:DR)까지 다양한 제품군에서 사용되고 있는 기술이다. Wi-Fi, BLE, LoRa, LTE-M 등 다양한 네트워크를 통해 M2M(Machine to Machine) 통신을 하며, 센서 들의 정보를 수집하고, 이를 분석 및 처리(예측 등)하여 사용자에게 유용한 정보를 제공한다.

**2.2. 장치 간 통신**

**2.2.1. 블루투스**

대표적인 근거리 통신기술로, 10M 정도의 짧은 거리에서 주로 사용한다. 사용하는 에너지가 다른 통신기술에 비해 크게 적어, 배터리 소모에 민감한 무선 키보드, 무선 이어폰, IoT장비 들에서 주로 사용된다. 기존 Wi-Fi와 동일한 2.4GHz대역을 이용하며, 1MHz 대역폭의 79개 채널 혹은 2MHz 대역폭의 40개 채널을 이용하여 통신하다. 2021년 현재, 5.2버전이 최신 버전이며, 최신 노트북에서 주로 찾아볼 수 있다. IoT에는 음성 통신+초저가가 아닌 이상 BLE(Bluetooth Low Energy) 기술이 도입된 4.0 이후 버전을 주로 사용한다.

**2.1.2. MQTT(Message Queueing Telemetry Transport)**

MQTT는 2016년 표준화된 Pub-Sub방식의 메시지 송수신 프로토콜이다. HTTP/HTTPS에 비해 메시지 용량이 가벼워 전력 효율이 중요한 IoT의 가벼운 데이터 전송이나, Facebook Messenger과 같이 대규모 데이터 전송에 주로 사용된다.

IoT기준으로 MQTT프로토콜은, 센서에서 브로커에 데이터를 Publish하고, 이를 클라이언트는 Subscribe하는 구조이다. 센서가 특정 Topic(주제)에 대해 데이터를 Publish하면 이 Topic을 구독하고 있는 클라이언트는 이 데이터를 받을 수 있게 된다. 단일 데이터에 대해 여러 장비가 각각 요청하지 않고, 전부 브로커가 관리하기 때문에 다대다, 1대다 와 같은 통신에 주로 사용된다.

**2.3. 아두이노**

대표적인 오픈하드웨어 플랫폼으로, 전자공작에서 단순 반복된 하드웨어 작업을 컨트롤할 때 주로 사용된다. 가장 대표적인 것이 Arduino UNO 보드로 GPIO핀을 통해 각종 센서나 장비들을 조작할 수 있다. 아두이노를 동작시킬 프로그램의 경우, Blink라는 IDE를 통해 C언어로 코드를 작성하여 보드에 업로드 할 수 있다. 하드웨어가 매우 제한된 것이 특징이다. 메모리 용량이 32KB에 16MHz의 CPU가 탑재된 만큼, 평소 짜던 코드와 다르게 매우 효율적으로 작성할 필요가 있다.

설계도가 모두 공개된 오픈하드웨어 플랫폼인 만큼 다양한 호환 제품이 있다. 최근에는 아두이노의 약한 연산능력을 보완하기 위해 x86기반 하드웨어에 아두이노 컨트롤러를 추가한 라떼판다라는 것도 등장하였다.,

메인 보드 뿐만 아니라, 센서 모듈 또한 무궁무진하게 많은 것이 특징이다.

**2.4. 기존 스마트 약병에 대한 연구**

미국의 Vitality사에서 만든 GlowCap이라는 제품은 약 먹을 시간을 하드웨어로 알려준다. 영국의 Elucid Health사에서 만든 Pill Connect라는 제품은 여기서 한단계 진화하여, 약의 개수를 제한하는 기능을 탑재하고, 약을 섭취했는지를 의료기관에서 실시간으로 확인할 수 있게 개량하였다.

국내에서도 이와 같은 연구는 한때 활발하게 진행되었다. IoTeen이라는 회사는 메디알람이라는 제품을 내놓아, 약 먹을 시간을 알려주는 장치를 상용화 하였다.

2015년부터 보건복지부의 복약 지도 강화에 의해, 약을 처방 받게 되면 약봉지에 약에 관한 정보가 인쇄 되어 있는 것을 볼 수 있다. 이를 통해 복약자가 약에 대한 정보를 좀 더 쉽게 알 수 있게 되었다. 그러나, 이 복약 지도는 물리적인 것으로, 현대 사회에서 접근이 떨어지는 것은 사실이다. 이를 공공데이터를 이용해 연동, 통합 한 제품은 아직 없다.

**2.5. 기존 연구의 문제점 및 해결 방안**

**2.5.1. 연구의 문제점**

기존 제품의 경우 복약자의 약 섭취 관리에 솔루션 개발의 중점이 맞춰져 있고, 실제 복약자의 궁금증에 대해서는 해소해주지 못하는 문제를 가지고 있다. 현재 시중에 나와 있는 개별 제품, 서비스를 살펴보면 약 복용 시간 관리, 복약지도, 약 보관 지도를 개별적으로 해주는 제품은 확인되고 있다. 그러나 이를 통합한 제품이 현재 시장에는 없는 상황이다. 이를 Reuse-oriented Integration and configuration모델을 이용하여 문제를 해결하고자 한다.

**2.5.2. 해결 방안**

**2.5.2.1. 자석 센서(리드 센서)**

약 복용시간을 관리하기 위해 자석 센서(리드 센서)를 사용하여 약병 개폐 여부를 확인한다. 리드 센서는 자석이 가까이 있으면 1을 반환하고, 없으면 0을 반환한다. 이 데이터를 이용해 복약자가 정해진 시간에 약병을 열고 닫았는지 기록할 수 있다.

**2.5.2.2. 7Segment LED**

약을 한 알 먹어야 될 수도 있고 아홉 알을 먹어야 될 수도 있다. 약 먹을 시간이 되면, 7Segment LED를 통해 먹어야 될 약의 수를 표시해준다. 이를 통해 복약자는 직관적으로 몇 개의 약을 먹어야 되는지 알 수 있게 된다.

**2.5.2.3. 초음파 센서**

약병에 약이 얼마나 남았는지 보통 무게로 알 수 있다. 그러나 원격으로도 이를 알기 쉽게 초음파 센서를 이용하고자 한다. 초음파센서는 2cm이상 떨어진 물체부터 0.1cm간격으로 거리를 측정할 수 있어 전체 통 길이에서 역산하여 약의 대략적인 잔량을 알 수 있다.

**2.5.2.4. 온습도 센서**

약 마다 적절히 보관 되어야 하는 온습도가 지정 되어 있다. 이를 확인하기 위해 온습도센서를 이용하고자 한다. DHT-22온습도센서는 스마트약병에 적합한 센서로, -40섭씨온도~80섭씨온도간 0.1섭씨온도 간격으로 측정할 수 있으며, 0~100% 사이를 0.1%간격으로 습도를 측정할 수 있다. 센서를 통해 취득한 데이터를 기반으로 적정 약 보관온습도와 비교하고, 최적의 온습도가 아닐 경우 안내할 수 있다.

**2.5.2.5. 의약품성분약효정보서비스 API**

의약품성분약효정보서비스 API는 건강보험심사평가원에서 운영하는 보건의료빅데이터 개방시스템의 오픈API이다. 복약자는 최초 약병 초기화 시, 약 정보를 소프트웨어에 입력하게 되고, 이 정보를 기반으로 API를 호출하여, 복약자에게 약효나 부작용 등 상세 정보를 안내할 수 있게 된다.

**3. 프로젝트 내용**

**3.1. 시나리오**

**3.1.1. Initial setting GUI**

** 텍스트이(가) 표시된 사진

자동 생성된 설명**

**[그림1] 시작 화면 [그림2] 회원 가입 화면**

[그림 1]은 어플리케이션의 시작 화면이다. 시작화면에서는 로그인 기능과 회원 가입 페이지로 넘어 갈 수 있다.

[그림 2]는 화원 가입 화면으로 사용자로부터 id, password, 약병에 사용할 약의 정보를 입력 후 회원 가입을 할 수 있도록 지원한다. 중복확인을 누를 시 현재 중복된 id가 있는지 확인이 가능하며 password 입력란과 password 확인란의 입력 내용이 다른 경우 밑의 빨간 글씨로 경고 문구가 출력 된다.

**3.1.2. Main GUI**

**텍스트, 스크린샷, 전자기기이(가) 표시된 사진

자동 생성된 설명 텍스트, 전자기기, 주차장, 스크린샷이(가) 표시된 사진

자동 생성된 설명**

**[그림 3] Main화면 [그림4] 약병 내부 정보**

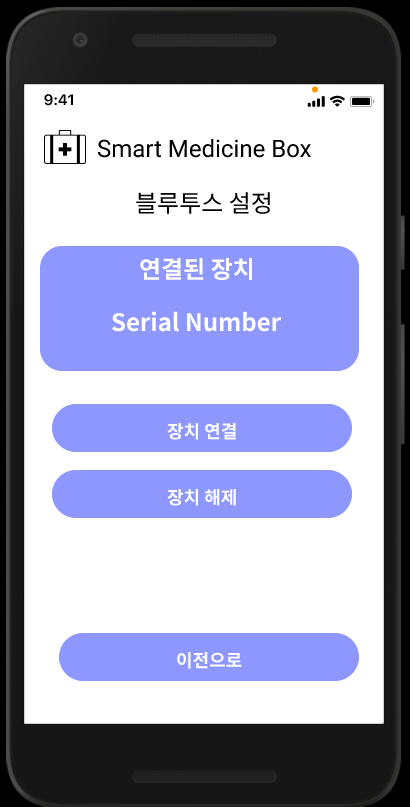
**텍스트, 전자기기, 스크린샷이(가) 표시된 사진

자동 생성된 설명 텍스트, 전자기기, 스크린샷이(가) 표시된 사진

자동 생성된 설명**

**[그림 5] 약병 외부 정보 [그림 6] 설정 화면**

**텍스트이(가) 표시된 사진

자동 생성된 설명 **

**[그림7] 알림 설정 팝업 [그림 8] 블루투스 설정**

**텍스트, 휴대폰, 전자기기, 스크린샷이(가) 표시된 사진

자동 생성된 설명 텍스트, 전자기기, 스크린샷이(가) 표시된 사진

자동 생성된 설명**

**[그림9] 블루투스 연결 [그림 10] 블루투스 검색**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

**[그림11] 약 정보 변경 팝업**

[그림 3]은 Application의 Main 화면이다. 설정 버튼과 약병 내부 정보, 약병 외부 정보 버튼을 통하여 각각 [그림 4], [그림 5], [그림 6]의 화면으로 넘어 갈 수 있다. Main 화면의 중앙에는 로그인한 유저의 정보를 바탕으로 현재 연동되어 있는 약병 안의 약의 정보를 출력해 준다. 약병 내부 정보 버튼을 누를 경우 [그림 4]가 출력이 되며 약병 외부 정보 버튼을 누를 경우 [그림 5]의 약병 외부 정보 화면이 출력이 되고 설정 icon을 누를 경우 [그림 6]의 설정 화면이 출력되게 된다.

[그림 4] 약병 내부 정보 화면에는 웹서버로부터 전달받은 약병 내부 온도, 습도, 약병 내부 잔량, 최근 개폐 시간의 정보를 출력을 해주게 된다. 약병 외부 정보 버튼을 누를 시 [그림 5]의 약병 외부 정보 화면이 출력되며 설정 icon을 누를 시 [그림 6]의 설정 화면이 출력되게 된다.

[그림 5] 약병 외부 정보 화면은 의약품성분약효정보서비스 API를 통해서 약 성분과 약효 정보와, 최근 개폐 시간을 통하여 계산된 다음 약 권장 복용 시간을 출력하게 된다.

약병 내부 정보 버튼을 통하여 [그림 4] 약병 내부 정보 화면으로 이동이 가능하다.

설정 icon을 통하여 [그림 6] 설정 화면으로 이동이 가능하다.

[그림 6]은 설정 화면으로 알림 설정, 블루투스 설정, 약 정보 수정, 개발자 정보 표시 기능을 수행한다. 알림 설정 버튼을 누를 시 [그림 7]의 알림 설정 화면 팝업이 나오면서, 약 복용 시간을 설정하여 알림 기능을 수행할 수 있도록 한다. 블루투스 설정 버튼을 누를 경우, [그림 8]블루투스 설정 화면을 출력하며, 현재 연결되어 있는 약병의 정보와 다른 장치(약병) 연결, 장치 연결 해제의 기능을 제공한다.

장치(약병) 연결 버튼을 누를 경우, [그림 9] 블루투스 연결 화면으로 넘어가며 현재 찾은 장치의 목록과 연결 여부를 묻게 된다. 검색 버튼을 누를 경우, [그림 10]의 블루투스 검색 화면으로 넘어 가며 장치를 검색하게 된다. 검색된 장치의 목록 중 하나를 선택하여 확인을 누르면, [그림 9]의 블루투스 연결 화면으로 돌아가며 최종적으로 연결을 누르는 경우 장치의 연결이 진행되게 되며 [그림 6]의 설정 화면으로 돌아간다.

약 정보 수정 버튼을 [그림 6] 설정 화면에서 누르는 경우 [그림 11]약 정보 변경 팝업 화면이 출력되게 되며 현재 사용자가 입력한 약의 정보를 변경 할 수 있는 기능을 수행한다.

**3.2. 요구사항**

**3.2.1. 디지털 리드 센서에 대한 요구사항**

**-** 아두이노 디지털 리드 센서를 이용하여 뚜껑이 언제 열리고 닫혔는지를 확인한다.

- 뚜껑의 개폐 여부에 대한 데이터를 웹서버에 전송하도록 구현한다.

**3.2.2. 온습도 센서에 대한 요구사항**

**-** 아두이노 온습도 센서를 이용하여 약병 내부의 온습도를 파악한다.

- 약병 내부의 온습도를 웹서버에 전송하도록 구현한다.

**3.2.3. 초음파 센서에 대한 요구사항**

**-** 아두이노 초음파 센서를 이용하여 약병 내부의 약의 잔량을 대략적으로 파악한다.

- 약병 내부의 약의 잔량을 웹서버에 전송하도록 구현한다.

**3.2.4. 약병에 대한 요구사항**

**-** 각 센서로부터 받은 데이터들을 웹서버로 전송하고, 가공된 데이터(섭취해야 될 약 개수 등)들을 수신 받는다.

- 수신 받은 데이터들을 바탕으로 user가 약을 복용해야 할 때 LED로 알려주고, 복용해야 될 양을 7Segment LED로 표시하는 것을 구현한다.

**3.2.5. 어플리케이션에 대한 요구사항**

**-** 회원가입 및 로그인/로그아웃을 할 수 있도록 한다.

- 어플리케이션에서 약병을 찾고, 약병을 연결할 수 있도록 한다.

- 로그인 시, user가 복용하고 있는 약이 무엇인지 등록할 수 있게 한다.

- 약을 언제 복용했는지에 대한 timeline을 알 수 있게 한다.

- 복용중인 약에 대한 전반적인 정보를 알 수 있게 한다.

- 허브로부터 얻은 데이터를 웹서버로 전송할 수 있게 한다. 또 웹서버에서 받은 데이터를 허브로 전송할 수 있게 한다.

**3.2.6. 웹 서버에 대한 요구사항**

**-** user의 정보를 저장한다.

- user가 등록한 약에 대한 정보를 의약품성분약효정보서비스 API를 이용해 받아오도록 한다.

- 어플리케이션으로부터 전달 받은 데이터를 가공한다.

- 가공된 데이터를 어플리케이션에 전달한다.

**3.2.7. BLE통신에 대한 요구사항**

**-** 허브를 이용하여 어플리케이션이 실행 중이지 않더라도 BLE 통신이 가능하게 한다.

- 어플리케이션과 약병의 데이터 교환이 가능하게 한다.

**4. 향후 일정 및 역할 분담**

안드로이드 어플리케이션 : 고원빈

웹서버 구현 : 박권수

데이터베이스 : 박권수

통신(허브) : 윤형선

하드웨어 프로그래밍(약병) : 윤형선

[전체 프로젝트 일정]

|  |  |
| --- | --- |
| **Task** | **진행일자** |
| 팀원 구성 | 3.09 ~ 3.15 |
| 주제 선정 | 3.16 ~ 3.22 |
| 기초조사서 작성 | 3.23 ~ 3.29 |
| 제작 비용 책정, | 3.30 ~ 4.05 |
| 설계 | 4.06 ~ 4.12 |
| 설계 | 4.12 ~ 4.19 |
| 중간 보고서 제출 | 4.20 ~ 4.26 |
| 상세 설계 | 4.27 ~ 5.03 |
| 구현 | 5.04 ~ 5.10 |
| 구현 | 5.11 ~ 5.17 |
| 구현 | 5.18 ~ 5.24 |
| 최종 데모 제작 및 테스트 | 5.25 ~ 5.31 |
| 검수 및 미비점 보완 | 6.01 ~ 6.07 |
| 최종 보고서 제출 | 6.08 ~ 6.14 |

[개별 프로젝트 일정] – App

|  |  |
| --- | --- |
| **Task** | **진행일자** |
| App개발환경 Setting, GUI 설계 | 4.06 ~ 4.12 |
| App 인터페이스 설계 및 구현 | 4.12 ~ 4.19 |
| App 인터페이스 설계 및 구현 | 4.20 ~ 4.26 |
| App과 Server 연동 상세 설계 | 4.27 ~ 5.03 |
| App과 Server 연동 | 5.04 ~ 5.10 |
| App과 Server 연동 | 5.11 ~ 5.17 |
| 장치와 연결 Test | 5.18 ~ 5.24 |
| 최종 데모 제작 및 테스트 | 5.25 ~ 5.31 |
| 검수 및 미비점 보완 | 6.01 ~ 6.07 |
| 최종 보고서 제출 | 6.08 ~ 6.14 |

[개별 프로젝트 일정] – Backend(Server, DB)

|  |  |
| --- | --- |
| **Task** | **진행일자** |
| Server 및 DBMS 선정 | 4.06 ~ 4.12 |
| Server 및 알고리즘, DB설계 및 구현 | 4.12 ~ 4.19 |
| Server 및 알고리즘, DB설계 및 구현 | 4.20 ~ 4.26 |
| Server 및 알고리즘, DB설계 및 구현 | 4.27 ~ 5.03 |
| Server Test 및 Cloud 배포 | 5.04 ~ 5.10 |
| App과 연결 | 5.11 ~ 5.17 |
| 장치와 연결 Test | 5.18 ~ 5.24 |
| 최종 데모 제작 및 테스트 | 5.25 ~ 5.31 |
| 검수 및 미비점 보완 | 6.01 ~ 6.07 |
| 최종 보고서 제출 | 6.08 ~ 6.14 |

[개별 프로젝트 일정] – Hardware

|  |  |
| --- | --- |
| **Task** | **진행일자** |
| 센서 필요량 정리 및 발주 | 4.06 ~ 4.12 |
| 회로 설계 및 초안 제작 | 4.12 ~ 4.19 |
| 회로 설계 및 초안 제작 | 4.20 ~ 4.26 |
| 약통과 허브 간 통신 테스트 | 4.27 ~ 5.03 |
| 약통과 허브 간 통신 테스트 | 5.04 ~ 5.10 |
| 약병 외부 제작 | 5.11 ~ 5.17 |
| 하드웨어 부분 최종 통합 및 테스트 | 5.18 ~ 5.24 |
| 최종 데모 제작 및 테스트 | 5.25 ~ 5.31 |
| 검수 및 미비점 보완 | 6.01 ~ 6.07 |
| 최종 보고서 제출 | 6.08 ~ 6.14 |

**5. 결론 및 기대효과**

노령화가 급속도로 진행되고 있는 사회에 코로나19사태가 겹쳐지면서 건강에 대한 관심이 높아진 요즘, 건강관련식품에 대한 수요도 높아지고 있다. 복용중인 약 이외의 종합 비타민 등도 찾는 사람이 많아졌는데, 스마트 약병을 통해 더 올바른 약 복용 습관을 만들 수 있다. 복용중인 약의 남용, 오용을 막고 제 때 섭취하여 건강에 긍정적인 영향을 줄 수 있다. 또한 건강보험심사평가원에서 제공하는 오픈API를 통해 본인이 복용중인 약에 대한 정보를 보다 정확하고 신속하고 편리하게 알 수 있을 것이다.

**6. 참고문헌**

[1] Arduino : <https://www.arduino.cc/reference/>

[2] MQTT : <https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.pdf>

[3] Android Reference : <http://developer.android.com/>

[4] 의약품성분약효정보서비스API : <https://opendata.hira.or.kr/op/opc/selectOpnsApi.do?divCd=&searchCnd=&searchWrd=&sno=1501&pageIndex=1>

[5] Bluetooth : <https://www.bluetooth.com/learn-about-bluetooth/radio-versions/>