**<PROJECT 중간보고서>**

**IoT 기술을 활용한 스마트케어 시각화 웹 구현**

**Team E : 유재민, 이재동, 이현재**

**엄현상 교수님**

**윤지수 Master (SK Planet)**

Table of Contents

1. Abstract 2

2. Introduction 2

3. Background Study 2

A. 관련 접근방법/기술 장단점 분석 2

B. 프로젝트 개발환경 2

4. Goal/Problem & Requirements 2

5. Approach 2

6. Project Architecture 3

A. Architecture Diagram 3

B. Architecture Description 3

7. Implementation Spec 3

A. Input/Output Interface 3

B. Inter Module Communication Interface 3

C. Modules 3

8. Current Status 3

9. Future Work 3

10. Division & Assignment of Work 3

11. Schedule 3

 [Appendix] Detailed Implementation Spec 3

# **Abstract**

본 프로젝트는 스마트기기로부터 수집되는 실시간 health data를 입력 받아, 이를 목적에 맞는 스마트케어 전용 차트로 표현하여 웹 페이지에 시각화 하는 것이 목적이다. 해당 프로젝트의 핵심은 health data를 차트로 만들어주는 라이브러리를 직접 구현하는 것이다. 이미 여러 차트 라이브러리가 상용화 되어 있지만, 스마트케어를 표현하기에 적합한 차트는 찾아보기가 힘들다. MQTT 프로토콜을 이용해 다양한 스마트기기에서 수집되는 data를 입력 받을 것이며, 웹 페이지는 스마트기기의 다양한 디스플레이에서 볼 수 있도록 반응형으로 구현할 것이다.

# **Introduction**

최근들어 스마트케어에 대한 관심이 나날이 높아지고 있다. 현재 대부분의 휴대용 기기에는 신체정보를 감지할 수 있는 센서가 포함되어 있어서 제조사의 어플리케이션을 통해 정보를 전달한다. 그러나 그런 어플리케이션은 사용자의 특정한 요구를 반영하거나 사용자의 취향대로 최적화할 수 없다는 단점이 있다. 그런 어플리케이션은 재미로 사용하기는 쉽지만 실제로 스스로의 건강상태에 대한 실시간 모니터링이 필요한 사람들이 믿고 사용하기는 어렵다. 이에 따라 신체정보를 시각화하여 제공하고 위험상태를 경고해주는 서비스에 대한 수요가 늘고있지만 그런 서비스는 현재 거의 존재하지 않는다. 인터넷에 배포되고 있는 오픈소스 웹차트 라이브러리를 통해 스마트케어 데이터를 시각화할 수는 있지만 그런 라이브러리에도 부족한 점이 많다. 신체정보의 특징에 따라 직관적으로 정보를 전달하거나 위험상태 및 정상상태를 구분할 수 없고 스마트케어 정보라는 특수성에 기인하여 차트를 시각화할 수도 없다. 따라서 본 프로젝트에서는 스마트케어 정보를 위한 웹차트 라이브러리를 만들어 웹페이지에 구현함으로서 사용자의 요구를 정확하게 반영한다. 또한 완성된 라이브러리를 자유롭게 배포함으로써 사람들이 자유롭게 수정하고 개선시킬 수 있도록 만든다.

# **Background Study**

## **관련 접근방법/기술 장단점 분석**

## **프로젝트 개발환경**

# **Goal/Problem & Requirements**

# **Approach**

**[프로젝트 수행 목표 또는 문제 정의 및 접근방법]**

# **Project Architecture**

## **Architecture Diagram**

## **Architecture Description**

# **Implementation Spec**

## **Input/Output Interface**

## **Inter Module Communication Interface**

## **Modules**

# **Current Status**

# **Future Work**

# **Division & Assignment of Work**

|  |  |
| --- | --- |
| 항목 | 담당자 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# **Schedule**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 내용 | 9월 | | | 10월 | | | | 11월 | | | | 12월 | |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |

# **[Appendix] Detailed Implementation Spec**

## **Module Name**

### **Function Name (Function Prototype)**

**[다음 예시를 참고하여 모든 module의 prototype을 작성하고, 그 안에서 동작하는 모든 function의 역할을 기술하시오]**

*int ssd\_compute\_access\_time(ssd\_t \*s, int elem\_num, ssd\_req \*\*reqs, int total)*

***SSD의 copyback 설정에 따라서 다른 일을 하는데, copyback이 able 되어 있다면 ssd\_issue\_overlapped\_ios 함수를 부른다.***