I - Tub(Intelligent-Tub)

「Bathtub service platform
that combines bigdata and iot-technology」

Class : 03 class
Subject : CapstonDesign1
TeamName : Ultimate
Professor : Jung Hyunsook
Leader : Lim Daein(20154300)
Member : Jung Haemin(20144748)
Seo Jeonguk(20154199)
Park Jisu(20154280)

CONTENTS

1. Program Introduction

2. Development contents

3. Development Details

4. Else

1. Program Introduction

1-1) Topic

I-Tub(Intelligent Tub)「Bathtub service platform that combines bigdata and iot-technology」

1-2) Development Motivation and Purpose

❑ Development Motivation

Baths are done by all ages, men and women of all ages. At the beginning or the end of the day, it was regrettable that people thought only of it as simply washing. Heat the water, use a towel to apply body wash and rinse with water, and I think it was too simple even though it was the most important activity. Taking that into account, we began to develop the act of bathing in order to emphasize the feeling of relaxation in simple washing and to make bath worthwhile by adding pleasure.

People are reluctant to use bathtubs. It's because annoying and takes too long to prepare, so it becomes work. If you turn on the water and check if it is at the right temperature, see if the water is filled to the right height, and if you put in the bath, check if it melted well and repeat the work. I thought that if these processes were automated, other people's baths were used, and the best and optimal environment was created, people would enjoy bathing and if they did it before going to work, it would encourage them and relieve the stress they had accumulated during the day if they had done it after work. I thought it would be more effective if big data and artificial intelligence were added together.

❑ Development Purpose

These kinds of contents are popular these days, such as PC rooms, salt rooms, and ice rooms that serve as many dishes as buffets, jjimjilbangs with themes, horror and mystery.
But there was no part about bathing. Therefore, such baths were also intended to be made into contents to be used as new business items and to create a platform that can be easily used at a low price so that anyone can receive quality service by selling or renting them.
And each person has a different style of bath. Some people relieve their fatigue by dipping themselves in hot water for several minutes, and others relieve stress by dipping themselves in cool water for a short time. There are so many people who want to data it and provide different services by age, age, occupation, and even predict using artificial intelligence technology to maximize customized services and usability.

1-3) Expectancy effects and Utilization plan

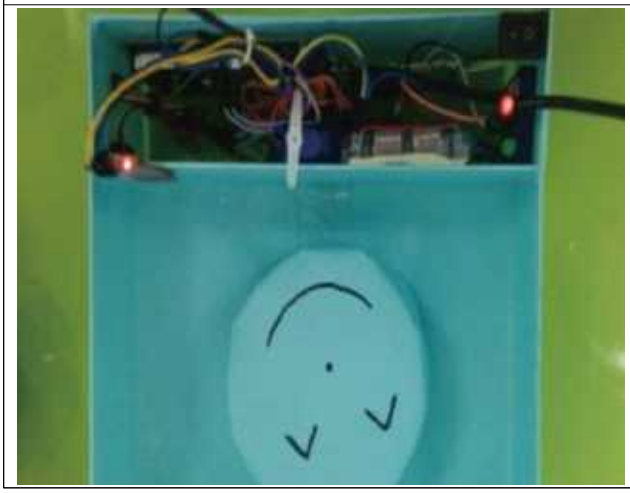
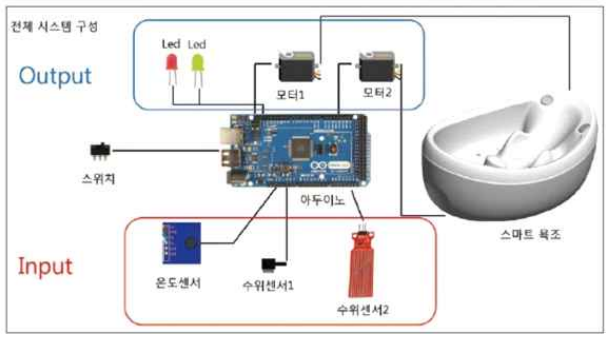
❑ Expectancy effects

- Environment created by a few touch without preparation코드의 핵심 만 적었다
- Provide an easily environment for relieving tired from school, housework, and other activities
- Expected to be satisfy the needs of younger generations who want customized services and convenient use and to be easily used by different age groups such as older people and children
- It is expected to be easily install it if people have a bathtub, it can be used anywhere as a detachable kit in various businesses and at home,

❑ Utilization plan

- People can experience different ways of experiencing different ways unlike jjimjilbang or sauna to make shower and bathing by one content.
- Even if you don't have a bathtub, you can use it if you change the shape of the kit slightly (change the model of parts / No need to change internal Software)

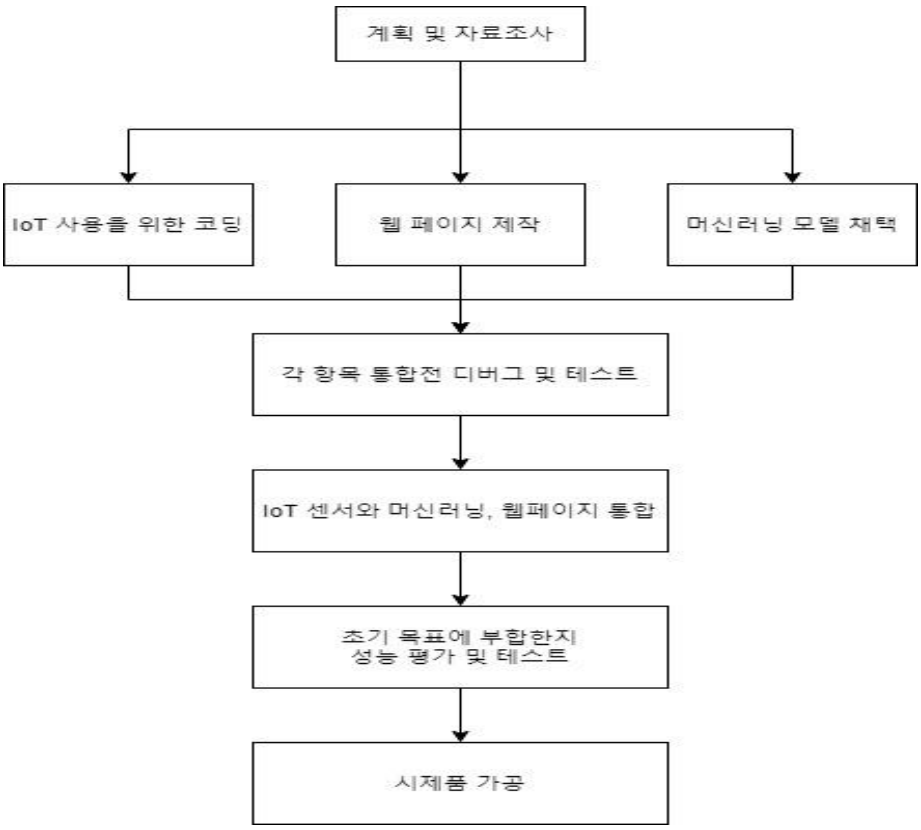
1-4) Existing Similar Research

Participation Prize of Busan National University ICT Convergence Project Contest in 2016	
	

This topic, Busan National University ICT Convergence Project in 2016, found something similar to our project. It is 'The Bathtub With Added Smart Function' which controls the water level, temperature set and maintain of water and automatically drains. The work in question planned to automatically create an environment when simply bathing, but since it includes the function and is a "smart bathtub using big data and machine running with customizing" it is thought that it will be able to provide enhanced user-tailored services by combining additional technology elements with IoT rather than just an automated system.

2. Development contents

2-1) Development plan and schedule



Step	Duration (month) (plan : ■)												note
	April				May				June				
	1	2	3	4	1	2	3	4	1	2	3	4	
project conference and topic selection	■	■											
Selection of development language and design		■	■										
Learning development language and program study			■	■									
Data collecting and preprocessing		■	■	■	■								
Develop				■	■	■	■	■	■				
Error minimization and refactoring									■	■	■		
User demonstration and deployment											■	■	
Project final demo and organization												■	

	week report
1	Background survey for task ideas, Online meetings to idea conference and team introduce
2	Final selection of project topic. Checking required data and parts
3	Determine parts and Modify topic Modeling and Web server&Database learning, design progress
4	Database design, prototype modeling(3Max) completed, Webpage created with Brackets Editor.
5	Collaboration development with GitHub. Web design and develop
6	Web server open and distribution. Database demonstration Data preprocessing and basic data design development
7	Checking Database, Web test and design development
8	Middle check
9	Development for final result
10	Develop completed and demonstration
11	Minimize error occurrence during User demonstration
12	Final user demonstration and prepare presentation
13	Participation in the Industrial-Academic CapstonDesign Competition
14	Individual opinion of CapstonDesign and project organization
15	Final report

2-2) Development Environment :

The development environment between projects is as follows (all people are the same)

- Operation System : Microsoft Windows 10 Education (version: 1903, build: 18362)
- Database : MySQL (version: 8.0)
- Web : HEROKU(Web distribution, version : 7.39.5), Node.js(Web server, version: 12.16.2 LST)
- Source Editor : Visual Studio Code (version: 1.38.1)
- IDE : Python IDLE (version : 3.7)
- Collaboration Tool : Github (Address : https://github.com/Ultimate-ItubProject)

The order of installation of development environment is as follows

1. Operation System : Microsoft Windows 10 Education (version: 1903, build: 18362)
2. Source Editor : VisualStudio Code (version: 1.38.1)
3. Web-related : Node.js(Web server, version: 12.16.2 LST),
   HEROKU(Web distribution, version : 7.39.5)
4. Database : MySQL (version: 8.0)
5. IDE : python IDLE (version : 3.7)

❑ Node.js

It is a platform built based on JavaScript engine and performs the role of server using JavaScript. This can replace the role of the popular PHP and JSP. And the advantage is that I/O processing can be handled asynchronously and event-oriented, so we can receive it every time we make a request, not when we want a value, and we can easily use event algorithms to get the desired event algorithm. It is also a platform suitable for services that handle data in real time, such as streaming data at high speed. So, I decided to use the platform that is suitable for the I-Tub project, which is important to exchange the set value with Raspberry Pi in real time.

❑ MySQL

MySQL uses standard SQL, widely known as relational database management system. It also supports a variety of programming languages, making it highly compatible and easy to modify for the user's use. Most of all, the free open source license is less performance and functionality than paid DB like ORACLE, but it is easily accessible. Also, there are many examples and error solutions in online so even beginners can use them easily. Also, Python and JavaScript were easily linked to the database, so it was used.

❑ Python

Python is a language that can easily incorporate machine learning and can check the results immediately using the interpreter method. It also has the advantage of being able to easily analyze data and even use visualization. Generally used for machine learning, data preprocessing, and visualization. We have come to use here for both machine learning, data pretreatment and visualization.

❑ Heroku

Heroku is a PaaS(platform-as-a-service) cloud service and is a platform that can easily deploy small Web servers. Limited resources can be used free of charge and there are many add-ons, which can easily be linked to web servers through addition. Once a website is created, it is designed exclusively for that site and has an isolated nature, which makes scaling easy and does not affect much even if there is a lot of access. I decided to use it because I thought it was suitable for this project.

❑ Github

The collaboration tool, GitHub, is a distributed version management system supported by Microsoft, that allows all operations to be recorded and tracked. Local storage and remote storage were separated, so it was easy to recover even if the local storage was accidentally erased, and it was possible to recover with a few lines of code even when the wrong operation was performed.

2-3) Developer

- Lim Daein : Web back-end, IoT and machine learing coding,
  : Data analysis and preprocessing, Prototype make.
- Jung Haemin : Web front-end, Data collecting.
- Seo Junguk : Web front-end.
- Park Jisu : Database Design, Prototype make.

2-4) Basic Design

1. Data collecting : Since there is no data on bathing, we collect data by determining what we need directly.

1-1) Collect data using a survey site. (using Now&Survey's Survey square)

1-2) Use GoogleForm to collect data from relatives and outsiders.

1-3) Collects and updates repeatedly until the end of the project.

2. Data preprocessing and visualization, Prediction Model Learning.

2-1) Perform data preprocessing that remove unnecessary elements from collected data such as the end time of the shower and Remove Unintentional Questionnaire.

2-2) Visualize the data based on the final data that has been pre-processed to determine if it is suitable for use or correlated.

2-3) Final data is used to create and execute machine learning models for predicting user services in Python and compare performance to confirm. (Testing models such as RandomForrest, XGBooost, and DecisionTree.)

3. Web Site Implementation

3-1) Create a framework for the website to serve users (using Node.js, Heroku, VScode)

3-2) Implement Web functions such as interlocking with member management and learned prediction models and statistical functions

3-3) Web environments, note local environments, ensure that the actual functions are functioning properly

3-4) Launch web services

4. Prototype making

4-1) Design a model framework that will work with web services

4-2) Checking and purchasing the parts needed to build the model in be linked with the school link business group

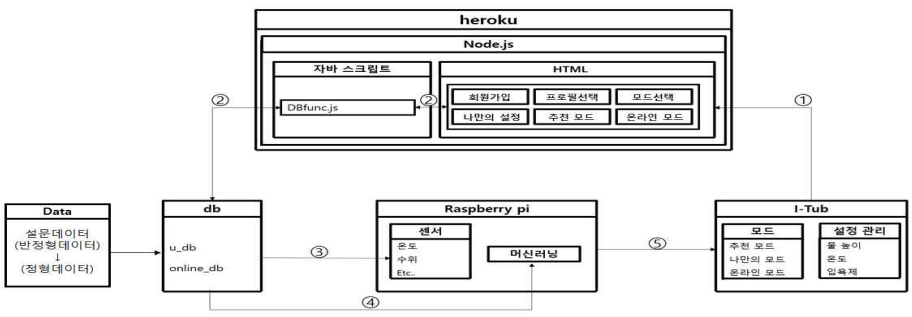
4-3) Coding for communication between sensors and web servers (manufactured using VScode, Python)

4-4) Interlock Test Between Web and Model

4-5) Prototype service

3. Development Details

3-1) I-Tub System Architecture



① The process in which users use I-Tub web services allows them to act such as signing up and selecting modes on each page.

② When a user uses a website, the user's information is stored in the DB as a process that is processed and the settings related to bathing are stored.

③ It is the process of operating various sensors by sending user information and bath setting values stored in the DB to Raspberry pi (at this time, DB has pre-surveyed bath survey data).)

④ Every time a user requests it, Raspberry Pi uses a machine learning model to learn and recommends the right values for the user.

⑤ Raspberry pi is the process of finally providing service to users, and sensors operate to control the height and temperature of the water.

3-2) Data collecting and preprocessing

❑ Data collecting

- Some of the members' relatives have jobs that are familiar with a lot of people, so conduct the survey with help.
- Upload surveys to professional survey sites to collect data

Survey	https://forms.gle/2VM7vNB8BpA2Zspu7 <GoogleForm> https://www.nownsurvey.com/issue/share/link/1471 <survey website>

❑ Collected data

The items collected were shown in the table below, and a total of nine factors were collected by adding factors (such as shower start time, availability of bath products, etc.) to consider when bathing before the survey, and (such as age, occupation, etc.) necessary for forecasting implementation.

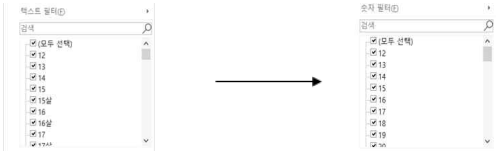
설문한 날짜	귀하의 성별은?	귀하의 나이는?	귀하가 원하는 샤워/목욕 온도는?	귀하의 샤워/목욕 시작 시간대는?	귀하의 샤워/목욕 하는데 걸리는 시간은?	목욕완료시간	샤워/목욕 시 입욕제 사용 여부	귀하의 직업은?	날씨	목욕시간대 온도
2020-04-17	남	12	시원함	16	10	16:10	무	학생	16.30 ℃	2020-04-17 16
2020-04-14	남	12	따뜻함	17	10	17:10	무	학생	20.10 ℃	2020-04-14 17
2020-04-14	남	13	시원함	17	15	17:15	무	학생	20.10 ℃	2020-04-14 17
2020-04-17	남	14	따뜻함	21	10	21:10	무	학생	13.80 ℃	2020-04-17 21
2020-04-04	남	14	시원함	17	20	17:20	무	학생	11.10 ℃	2020-04-04 17
2020-04-14	남	15	미지근함	16	10	16:10	무	학생	20.40 ℃	2020-04-14 16
2020-04-02	남	15	따뜻함	20	10	20:10	무	학생	10.90 ℃	2020-04-02 20
2020-04-17	남	15	시원함	22	10	22:10	무	학생	12.30 ℃	2020-04-17 22
2020-04-17	남	15	따뜻함	17	20	17:20	유	학생	16.40 ℃	2020-04-17 17
2020-04-26	남	15	따뜻함	21	20	21:20	무	학생	14.10 ℃	2020-04-26 21
2020-04-26	남	16	따뜻함	17	7	17:07	무	학생	18.10 ℃	2020-04-26 17
2020-04-14	남	16	시원함	20	10	20:10	무	학생	13.80 ℃	2020-04-14 20
2020-04-24	남	16	따뜻함	19	20	19:20	무	학생	12.90 ℃	2020-04-24 19
2020-04-28	남	16	따뜻함	20	30	20:30	유	학생	14.30 ℃	2020-04-28 20
2020-04-28	남	17	미지근함	20	5	20:05	무	학생	14.30 ℃	2020-04-28 20
2020-04-17	남	17	따뜻함	12	10	12:10	무	학생	14.50 ℃	2020-04-17 12
2020-04-17	남	17	시원함	23	10	23:10	무	학생	11.00 ℃	2020-04-17 23
2020-04-27	남	17	미지근함	7	15	7:15	무	학생	7.60 ℃	2020-04-27 7
2020-04-14	남	17	따뜻함	11	15	11:15	무	학생	15.70 ℃	2020-04-14 11
2020-04-28	남	17	미지근함	21	15	21:15	무	학생	13.60 ℃	2020-04-28 21
2020-04-17	남	17	따뜻함	23	15	23:15	무	학생	11.00 ℃	2020-04-17 23

Column Name	explain
Date	Date surveyed
Gender	Select male or female
Age	Surveyer`s age
Favorite temperature	Survey select favorite temperature for shower (hot, normal, cold)
Start bathing time	Time to start shower
During bathing time	Time to during shower
End bathing time	Time to end shower
bathbombs	Whether or not to use bathbombs for shower
Job	Surveyor`s job
Weather	temperature of each day
Temperature of date	Combining date and weather column

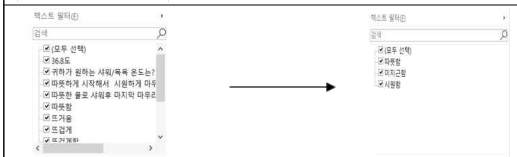
\

❑ Data preprocessing

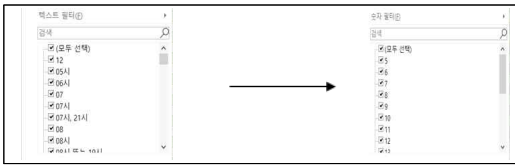
After collecting the data, there were answers that were different from the intention of the survey, so the process of unification was carried out through the data preprocessing process.



- Unify various answers (like ‘만 24세’, ‘24살')
into numbers (such as 24, 25)



- Unify multiple answers (like '뜨뜻하게‘)
into three answers ('따뜻함', '미지근함', '시원함')

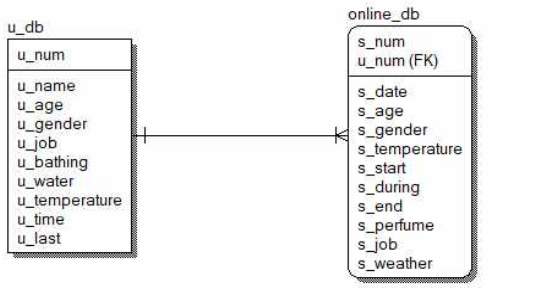


- Unify various answers (like AM, PM, 00:00) into 24 hours(number) (such as '10', '18')

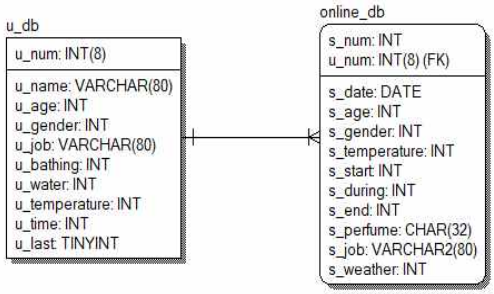
3-3) I-Tub Database

The database consists of tables that store information about members and tables that store values and data obtained from surveys.

❑ Design with Erwin



Logical Design



Physical Design

❑ online_db (This table that stores data used by users)

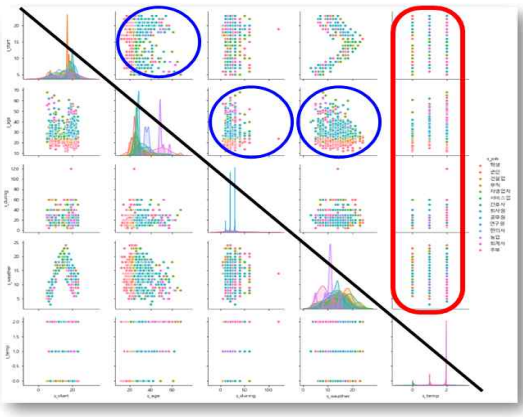
Column Name	Type	Constraint	Note
s_num	INT	NOT NULL AUTO_INCREMENT PRIMARY KEY	auto_increment
s_date	DATE	NOT NULL	date
s_age	INT	NOT NULL	user age
s_gender	INT	NOT NULL	user gender
s_temperature	INT	NOT NULL	temperature
s_start	INT	NOT NULL	start bathtime
s_during	INT	NOT NULL	during bathtime
s_end	INT	NOT NULL	end bathtime
s_perfume	Char(32)	NOT NULL	bathbombs
s_job	VarChar(80)	NOT NULL	user job
s_weather	INT	NOT NULL	weather to date

❑ u_db (This table that stores data informations by users)

Column Name	Type	Constraint	Note
u_num	INT	NOT NULL AUTO_INCREMENT PRIMARY KEY	auto_increment
u_name	VarChar(80)	NOT NULL	user name
u_age	INT	NOT NULL	user age
u_gender	INT	NOT NULL	user gender
u_job	VarChar(80)	NOT NULL	user job
u_bathing	INT	NOT NULL	bathbombs of user setting
u_water	INT	NOT NULL	water level of user setting
u_temperature	INT	NOT NULL	temperature of user setting
u_time	INT	NOT NULL	timer of user setting
u_last	TINYINT	NOT NULL	check user that using

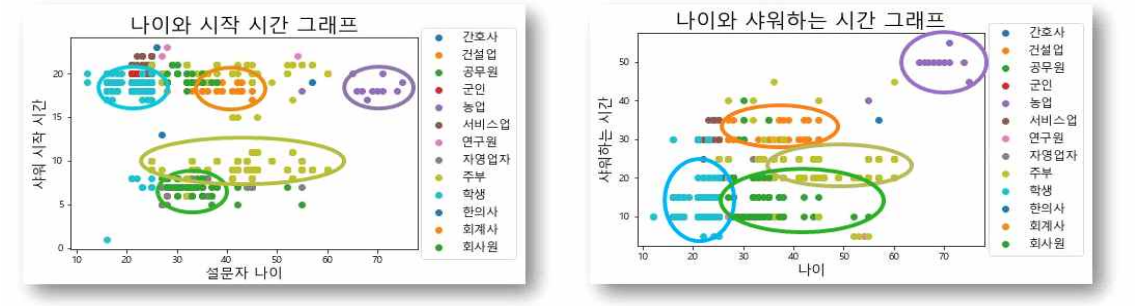
3-4) I-Tub Machine Learning Model Selection

❑ Scatterplot between all elements



Before selecting the model, we drew a scatterplot among all the elements and proceeded with the process of selecting important elements. In the graph, blue circles were necessary, red circles were unnecessary, for example, and shower time and shower temperature were randomly distributed so that they were not correlated, and in the case of user age and shower start time, they were clustered and correlated like blue circles. The analysis was conducted in detail by selecting two of the most highly correlated factors.

❑ Two highest correlation scatterplots



Factors such as age and name, occupation and name could not be correlated, so they were excluded and proceeded. Age and start time are also graphically shown that although not everyone does, it is possible to classify them because of the clustered time of showering according to age and occupation. In addition, it was found that most of the shower times, depending on age and occupation, could be clustered and sorted separately, which allowed the model to determine which columns to use and which to predict when forecasting. Other factors did not have a significant correlation with each other, although they were included in the necessary factors in the prediction, but did not have a significant impact.

	Column Name
Elements Required for Prediction	Age, Job, Gender, Weather
Predictable element	bathing time, bathing start time, temperature of bathing

Through the visualization process, we found highly correlated elements among the elements and found predictable elements and conducted tests with various machine learning algorithms to select models based on factors such as accuracy and precision.

❑ Machine Learning Model Selection

The best model has higher precision and recall than another models,The concept is as follows

- Accuracy : A figure of correct answers matched by model`s predictions
- Precision : A figure of how many include of correct answer by model`s predictions
- Recall : A figure of correct answers predicted by model
- F1-score : Harmonic mean of precision and recall

(A figure obtained by Harmonic mean of precision is overwhelmingly higher than recall or vice versa)

Model	Accuracy	Precision	Recall	F1-score
DecisionTree	65% (train : 76%)	66%	65%	64%
RandomForest	65% (train : 77%)	66%	65%	64%
XGBoost	66% (train : 76%)	66%	66%	66%
KNN (K-nearest-neighbors)	50% (train : 50%)	51%	50%	49%

Comparisons show that the XGBoost model has the best figures, but it takes the longer to predict, so we choose the faster and the next best model, Randomforest

❑ Source code with above model (predict.py)

```
#-*- coding:utf-8 -*-
import  numpy as  np
import  pandas as  pd
import  time
import  sys

# import statement to use random forest model
from  sklearn.ensemble import  RandomForestClassifier

# import statement to share learning and test data and to check various evaluation scores
from  sklearn.metrics import  confusion_matrix, classification_report, roc_curve, auc
from  sklearn.model_selection import  train_test_split, cross_val_score

import  requests, json

col_li = ['s_weather','s_age','s_gender','직업_간호사','직업_건설업','직업_공무원','직업_군인','직업_농업','직업_무직','직업_서비스업','직업_연구원','직업_자영업자','직업_주부','직업_학생','직업_한의사','직업_회계사','직업_회사원']

def  machin (py_age , py_gender , py_job , predict_col ):
    idx = 0
    new_df = []

    # the part where data is edited for inclusion in the predictive model.
    for  i in  range (0 , len (col_li)):
        if  (col_li[i] == ('직업_' + py_job)):
            idx = i
    for  i in  range (0 , 17 ):
        if  (i == idx):
            new_df.append(1 )
        elif  (i == 0 ):
            new_df.append(20 )
        elif  (i == 1 ):
            new_df.append(int (py_age))
        elif  (i == 2 ):
            if  (py_gender == '여성'):
                new_df.append(1 )
            else :
                new_df.append(0 )
        else :
            new_df.append(0 )
    # Tells precision, reproducibility rate, f1 score, support.
    x = pd.concat([shower_data['s_weather'], shower_data['s_age'], shower_data['s_gender'],
                                pd.get_dummies(shower_data['s_job'], prefix ='직업')], axis =1 )
    y = shower_data[predict_col]
    x_train, x_test, y_train, y_test = train_test_split(x, y, test_size =0.3 , random_state =44 )
    model = RandomForestClassifier()
    model.fit(x_train, y_train)
    tmp = pd.DataFrame(data =new_df).T
    tmp.columns = col_li
```

```
    # The part that is edited to pass the value to the web server.
    if  (predict_col == 's_temp'):
        result = str (model.predict(tmp))[2 :-2 ]
    else :
        result = str (model.predict(tmp))[1 :-1 ]
    new_df = []
    return  result

shower_data = pd.read_csv('./public/survey_data.csv', encoding ='utf-8')
shower_data.head()

# Change to 0 and 1 respectively, man and women
shower_data.loc[shower_data['s_gender']=='남', 's_gender'] = 0
shower_data.loc[shower_data['s_gender']=='여', 's_gender'] = 1
#shower_data.head()

# Change the presence of bath bombs to numbers
shower_data.loc[shower_data['s_perfume']=='무', 's_perfume'] = 0
shower_data.loc[shower_data['s_perfume']=='유', 's_perfume'] = 1
#shower_data.head()

shower_data['s_gender'] = pd.to_numeric(shower_data['s_gender'])
shower_data['s_perfume'] = pd.to_numeric(shower_data['s_perfume'])
shower_data.drop(['s_end'], axis =1 , inplace =True )
shower_data.head()

# A code that modifies various information passed over from a web server and puts it into the machine
learning function.
start = machin(sys.argv[1], sys.argv[2], sys.argv[3], 's_start')
during = machin(sys.argv[1], sys.argv[2], sys.argv[3], 's_during')
temp = machin(sys.argv[1], sys.argv[2], sys.argv[3], 's_temp')
print (str (start+'/'+during+'/'+temp))
```

❑ Result of Predictions that apply to the above model

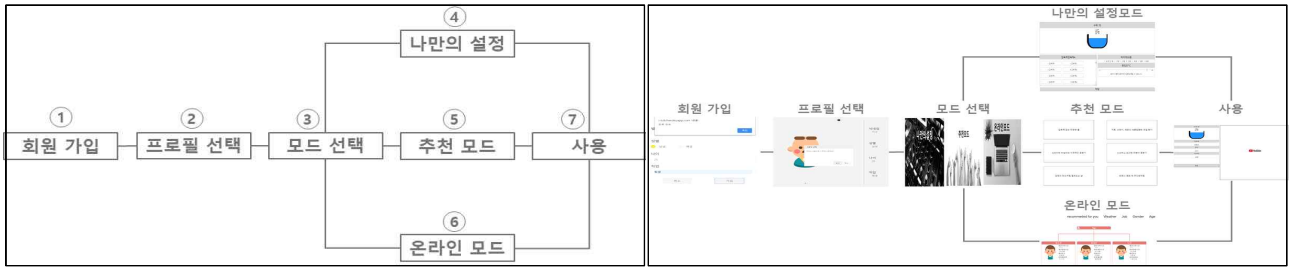
Prediction 1

Prediction 2

It is a model selected, and it seems to have made predictions by substituting actual data. If you look at the part checked by the red box, you can see the predicted appearance using the user's gender, age, occupation, and the weather for that date. According to the selected model, a 14-year-old female YouTuber is expected to take a bath at 5:00 a.m., do it for 30 minutes, and at 25 a.m., male and student will bathe at 12:00 a.m., take a bath for 20 minutes and take a warm bath.

3-4) WebPage

I-Tub Webpage consists of a total of seven pages, with roles and configurations as follows.



❑ HTML

File Name	Description
main_page.html	Mainpage
user_choice.html	own_setting, online_mode, recommend_mode
recommend.html	Recommend page
own_settings.html	own_setting page
profile.html	user register and select page
register.html	new register page
online_mode.html	uses user information to predict and service usage and to statistically visualize and view usage information for others.
item_info.html	you can show user-set usage and watch media

❑ CSS

File Name	Description
main_page_style.css	Mainpage`s stylesheet
user_choice_style.css	each page(own_setting, online_mode, recommend_mode)`s style sheet
recommend_style.css	Recommend page`s stylesheet
own_settings_style.css	own_setting page`s stylesheet
profile_style.css	user register and select page`s stylesheet
register_style.css	register page`s stylesheet
swiper.min.css	Stylesheet for Using Web Slides
jquery-ui.css	Stylesheet for using Jquery
item_info_style.css	Usage Page`s Style Sheet

❑ JS

File Name	Description
DBfunc.js	JavaScript for Using the Database
index.js	JavaScript acts as a web page controller
swiper.min.js	JavaScript to Use Web Slides
jquery-3.5.1.js	JavaScript for Use of Jquery
jquery-ui.js	Javascript for using Jquery's UI
orgchart.js	Javascript for using Orgcharts

❑ DATA

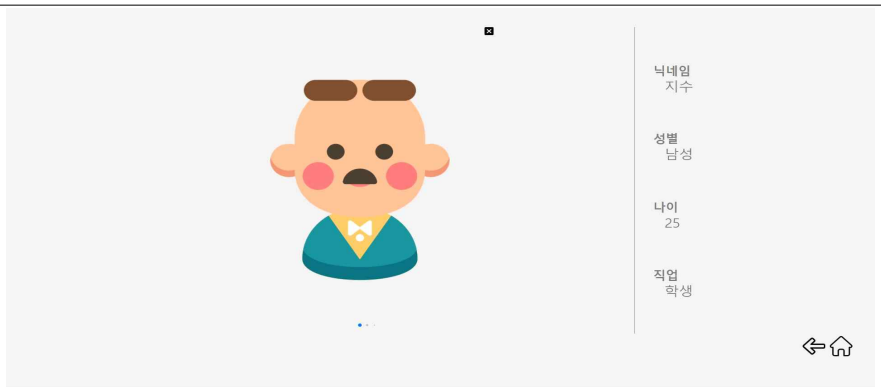
File Name	Description
survey_data.csv	Data collected in a survey and based on this data, the user's usage values are updated in this data.

❑ Webpage Implementation Part

Only the core of the code, complete code found at https://github.com/Ultimate-ItubProject/Itub_Web

MainPage

Profile Selection Page




```
<script  src ="./js/jquery-3.5.1.js"></script> //제이쿼리 사용
<script  src ="./js/jquery-ui.js"></script>
<script  src ="./js/swiper.min.js"></script>
<script>
    var  tmp  = '';
    var  swiper  = new  Swiper ('.swiper-container', {
        loop: false ,
        pagination: {
            el: '.swiper-pagination',
            clickable: true ,
            dynamicBullets: true ,
        },
        navigation: {
            nextEl: '.swiper-button-next',
            prevEl: '.swiper-button-prev',
        },
        on: {
            tap : function  () {
                if  (tmp .length  != 0  && swiper .activeIndex  !== tmp .length ) {
                    $(function  () {
                        $('#dialog_text').text ('[' + tmp [swiper .activeIndex].u_name
                                                    + '] 사용자로 시작하시겠어요?');
                        $("#dialog").dialog ({
                            //Set titme
                            title: '사용자 선택',
                            //Width and height of LayerPopup
                            width: 400 ,
                            height: 250 ,
                            //when you want to set background disable,set true
                            modal: true ,
                            //button type
                            buttons: [
```

```
                                {
                                    //button text
                                    text: "확인",
                                    //action when click event occurs
                                    click : function  () {
                                        document .getElementById ('cookie_name').value  =
                                        document .getElementById ('label_name').innerText ;
                                        document .getElementById ('cookie_age').value  =
                                        document .getElementById ('label_age').innerText ;
                                        document .getElementById ('cookie_gender').value  =
                                        document .getElementById ('label_gender').innerText ;
                                        document .getElementById ('cookie_job').value  =
                                        document .getElementById ('label_job').innerText ;
                                        document .getElementById ('cookie_submit').click ();
                                    }
                                },
                                {
                                    text: "취소",
                                    //action when click event occurs
                                    click : function  () {
                                        $(this ).dialog ("close");
                                    }
                                }
                            ]
                        });
                    });
                }
                else  document .getElementById ('register_submit').click ();
            },
            slideChangeTransitionStart : function  () {
                if  (tmp .length  != 0  && swiper .activeIndex  !== tmp .length ) {
                    document .getElementById ('label_name').innerHTML  =
                    tmp [swiper .activeIndex].u_name ;
                    if  (!tmp [swiper .activeIndex].u_gender )
                    document .getElementById ('label_gender').innerHTML  = '남성';
                    else  document .getElementById ('label_gender').innerHTML  = '여성';
                    document .getElementById ('label_age').innerHTML  =
                    tmp [swiper .activeIndex].u_age ;
                    document .getElementById ('label_job').innerHTML  =
                    tmp [swiper .activeIndex].u_job ;
                    $('.btn').show ();
                }
                else  {
                    document .getElementById ('label_name').innerHTML  = '사용자를 추가시켜 주세요';
                    document .getElementById ('label_gender').innerHTML  = '사용자를 추가시켜 주세요';
                    document .getElementById ('label_age').innerHTML  = '사용자를 추가시켜 주세요';
                    document .getElementById ('label_job').innerHTML  = '사용자를 추가시켜 주세요';
                    $('.btn').hide ();
                }
            }
        }
    });
    function  del_div () {
        else  {
            $(function  () {
                $('#dialog_text').text ('[' + tmp [swiper .activeIndex].u_name
                                                + '] 사용자를 삭제하시겠습니까?');
                $("#dialog").dialog ({
                    title: '사용자 선택',//set title
                    //width and height of layerpopup
```

```
                    width: 400 ,
                    height: 200 ,
                    modal: true , //when you want to background disable,set true
                    //button type
                    buttons: [
                        {
                            text: "확인",//button text
                            click : function  () {//action when click event occurs
                                sendDB ('/db/delete', tmp [swiper .activeIndex].u_num ); //사용자 삭제
                                alert (tmp [swiper .activeIndex].u_name  + ' 사용자가 삭제되었습니다.');
                                window .location .reload (true );
                            }
                        },
                        {
                            text: "취소",                //button text
                            click : function  () {       //action when click event occurs
                                $(this ).dialog ("close");
                            }
                        }
                    ]
                });
            });
        }
    }
    function  add_div () {
        console .log (tmp .length );
        if  (tmp .length  == 0 ) {
            swiper .appendSlide ('<div class="swiper-slide"><div class="vertical">
                                  <img src="./images/plusBtn.png" class="imgtag"></div></div>');
            swiper .update ();
            $('.btn').hide ();
        }
        else  {
            for  (var  i  = 0 ; i  < tmp .length ; i ++) {
                if  (tmp [i].u_age  < 15 ) {
                    if  (tmp [i].u_gender  == 0 ) path  = './images/boy.png'
                    else  path  = './images/girl.png'
                } else  if  (tmp [i].u_age  >= 15  && tmp [i].u_age  <= 65 ) {
                    if  (tmp [i].u_gender  == 0 ) path  = './images/fa.png'
                    else  path  = './images/ma.png'
                } else  {
                    if  (tmp [i].u_gender  == 0 ) path  = './images/granfa.png'
                    else  path  = './images/granma.png'
                }
                var  htmlCode  = '<div class="swiper-slide"><div class="vertical">
                                  <img src=' + path  + ' class="imgtag"></div></div>';
                swiper .appendSlide (htmlCode );
                swiper .update ();
            }
            swiper .appendSlide ('<div class="swiper-slide"><div class="vertical">
                                 <img src="./images/plusBtn.png" class="imgtag"></div></div>');
            document .getElementById ('label_name').innerHTML  =
            tmp [swiper .activeIndex].u_name ;
            if  (!tmp [swiper .activeIndex].u_gender )
                document .getElementById ('label_gender').innerHTML  = '남성';
            else  document .getElementById ('label_gender').innerHTML  = '여성';
            document .getElementById ('label_age').innerHTML  = tmp [swiper .activeIndex].u_age ;
            document .getElementById ('label_job').innerHTML  = tmp [swiper .activeIndex].u_job ;
        }
    }
```

```
        function  sendAjax (url , data ){
            var  inputdata  = {'temp':'0'};
            data  = JSON .stringify (inputdata );
            var  xhr  = new  XMLHttpRequest ();
            xhr .open ('POST', url );
            xhr .setRequestHeader ('content-type', 'application/json');
            xhr .send (data );
            xhr .addEventListener ('load', function  () {
                console .log (xhr .responseText );
            });
        };
        function  sendDB (url , data ) {
            var  data  = {
                'data': data
            };
            console .log (data  + '데이터형 변경 전');
            data  = JSON .stringify (data ); // json to make string type
            console .log (data  + '데이터형 변경 후');
            var  xhr  = new  XMLHttpRequest ();
            xhr .open ('POST', url );
            xhr .setRequestHeader ('Content-Type', 'application/json');
            // specify to send json type data when sending to server
            xhr .send (data ); // Must be in string type
            console .log (data  + 'xhr send')
            xhr .addEventListener ('load', function  () {
                var  result  = JSON .parse (xhr .responseText );
                var  resultDiv ;
                console .log ('데이터 넘겨받음' + result .result )
                if  (result .result  == 'ok' && result .type  == 'reference') {
                    tmp  = result .DB ;
                    add_div ();
                    console .log ('조회가 완료되었습니다.');
                } else  if  (result .result  == 'ok' && result .type  == 'delete') {
                    console .log ('삭제가 완료되었습니다.');
                } else  if  (result .result  == 'none' && result .type  == 'reference') {
                    add_div ();
                    console .log ('삭제 실패 하였습니다');
                } else  if  (result .result  == 'none' && result .type  == 'delete') {
                    console .log ('삭제 실패 하였습니다');
                }
            });
        }
        window .onload  = function  () {
            sendDB ('/db/reference', 'ok');
        }; //Run reference from DBfunk when loading pages
    </script>
```

Mode Selection Page

Own Setting Mode page


```
<script src ="./js/jquery-3.5.1.js"></script>
    <script src ="./js/orgchart.js"></script>
    <script src ="https://cdnjs.cloudflare.com/ajax/libs/jquery-csv/0.71/jquery.csv-0.71.min.js"></script>
    <script>
        var  water  = 70 ;
        var  bathing  = '없음';
        var  temperature  = 36 ;
        var  time  = '설정 안함';
        //storage of data cookies for user usage information (level, bath, temperature, time)
        saveBtn  = document .querySelector ('#save-btn')
        saveBtn .addEventListener ('click', function (e ) {
                var  form  = document .createElement ('form');
                form .setAttribute ('method', 'POST');
                form .setAttribute ('action', '/item_cookie' );

                var  waterFiled  = document .createElement ('input')
                waterFiled .setAttribute ('type', 'hidden');
                waterFiled .setAttribute ('name', 'water');
                waterFiled .setAttribute ('value', water ); // 수위 쿠키 저장
                form .appendChild (waterFiled );
                var  bathingFiled  = document .createElement ('input');
                bathingFiled .setAttribute ('type', 'hidden');
                bathingFiled .setAttribute ('name', 'bathing');
                bathingFiled .setAttribute ('value', bathing ); //입욕제 쿠키 저장
                form .appendChild (bathingFiled );
                var  temperatrueFiled  = document .createElement ('input');
                temperatrueFiled .setAttribute ('type', 'hidden');
                temperatrueFiled .setAttribute ('name', 'temperature');
                temperatrueFiled .setAttribute ('value', temperature ); //온도 쿠키 저장
                form .appendChild (temperatrueFiled );
                var  timeFiled  = document .createElement ('input');
                timeFiled .setAttribute ('type', 'hidden');
                timeFiled .setAttribute ('name', 'time');
                timeFiled .setAttribute ('value', time ); //시간 쿠키 저장
                form .appendChild (timeFiled );
                document .body .appendChild (form );
                form .submit ();
        })
        // The part where bath items are generated.
        var  parent  = document .querySelector (".list");
        for  (var  i  = 0 ; i  < 9 ; i ++) {
            var  bathbomb  = new  Array ('없음','큘', '바닐라', '라벤더', '레몬', '자스민',
                                                    '로즈제라늄', '로즈마리', '블루마린');
            var  div  = document .createElement ('div');
            div .className  = 'bathing item'
```

```
            var  label  = document .createElement ('label');
            var  input  = document .createElement ("input");
            input .type  = 'radio';
            input .name  = 'bathing_agent';
            input .value  = bathbomb [i];
            if (i ==0 ) input .checked ='checked';
            label .appendChild (input );
            var  span  = document .createElement ("span");
            span .className  = 'bathing_item_text';
            span .appendChild (document .createTextNode (bathbomb [i]));
            label .appendChild (span );
            div .appendChild (label );
            parent .appendChild (div );
        }

        //The part that allows the selected bath to be printed on the top title when an item is clicked in
the bath list.
        var  bathingItemList  = document .querySelectorAll (".bathing_item label");
        for  (var  i  = 0 ; i  < bathingItemList .length ; i ++) {
            bathingItemList [i].addEventListener ("click", (e ) => {
                document .querySelector ("#bathing_text").innerHTML  = ' ' + e .target .value ;
                bathing  = e .target .value ;
            })
        }
        // The part that allows the selected time to be printed on the timer title when the timer is set.
        var  timeList  = document .querySelectorAll ("#timeSection input");
        for  (var  i  = 0 ; i  < timeList .length ; i ++) {
            timeList [i].addEventListener ("click", (e ) => {
                var  text  = e .target .value ;
                //
                time  = e .target .value ;
                console .log (e .target .value );
                if  (text  !== '설정 안함') {
                    text  += "분"
                }
                document .querySelector ("#timer_text").innerHTML  = ' ' + text ;
            })
        }
        var  temprange  = document .querySelector ("#range_slider"); // range slider of water temperature
        var  tempvalue  = document .querySelector ("#range_text"); // indication text of water temperature
of water setting
        temprange .addEventListener ("input", (e ) => {
            tempvalue .innerHTML  = e .target .value ; // Read the current value of the range and save it
in the temperature display text
            //
            temperature  = e .target .value ;
        }, false )
        var  waterrange  = document .querySelector (".water-slider");
        var  watervalue  = document .querySelector ("#water-title");
        waterrange .addEventListener ("input", (e ) => {
            watervalue .innerHTML  = e .target .value ;
            //
            water  = e .target .value
        }, false )
        var  getCookie  = function (name ) { // Import cookies based on key values
            var  value  = document .cookie .match ('(^|;) ?' + name  + '=([^;]*)(;|$)');
            return  value ? value [2] : null ;
        };

        // If horizontal bar moves, vertical bar moves as well.
        var  h_to_v  = function () {
            var  h  = $('#water-slider-alfl');
            h .on ('input',function () {
                var  v  = $('.water-slider');
                v [0].value  = h [0].value ;
                watervalue .innerHTML =v [0].value ;
            })
        }
        // if vertical bar moves, horizontal bar as well
        var  v_to_h  = function () {
            var  v  = $('.water-slider');
            v .on ('input',function () {
                var  h  = $('#water-slider-alfl');
                h [0].value  = v [0].value ;
            })
        }
        // There must be two at the bottom to change the value in real time by dragging.
        h_to_v ();
        v_to_h ();
    </script>
```

Recommend Mode Page


```
<script>
        //titlename, watlerlevel, using time of bathing, temperature, bathbombs type
        var  select_mode  = [
            ['입욕제 없는 따뜻한 물', [80 , 5 , 43 , '없음'], './images/hotwater.png'],
            ['자몽, 오렌지, 레몬의 새콤달콤한 과일 향기', [80 , 30 , 38 , '귤'], './images/orange.png'],
            ['크리미한 바닐라와 이국적인 꽃향기', [80 , 30 , 38 , '바닐라'], './images/exotic.png'],
            ['신선하고 포근한 라벤더 꽃향기', [80 , 20 , 38 , '라벤더'], './images/lavender.png'],
            ['감정이 파도처럼 밀려오는 날', [80 , 20 , 38 , '레몬'], './images/wave.png'],
            ['로맨스 영화 속 주인공처럼', [80 , 30 , 38 , '자스민'], './images/romance.png'],
            ['새벽녘, 그 감성속으로', [80 , 30 , 38 , '로즈제라늄'], './images/sunset.png'],
            ['감성이 필요한 당신을 위한 선물', [80 , 30 , 30 , '로즈마리'], './images/present.png'],
            ['꾸리꾸리 비오는 날엔', [80 , 25 , 38 , '블루마린'], './images/rainnyday.png'],
            ['생각이 많은 요즘', [80 , 30 , 36 , '허브'], './images/herb.png']
        ]
        var  parent  = document .querySelector (".list"); // Create each button in advance
        for  (let  i  = 0 ; i  < select_mode .length ; i ++) {
            var  button  = document .createElement ('button');
            button .className  = 'btn';
            button .onclick  = function () {
                var  form  = document .createElement ('form');
                form .setAttribute ('method', 'POST');
                form .setAttribute ('action', '/item_cookie');
                var  a_Field  = document .createElement ('input'); // save cookie of waterlevel
                a_Field .setAttribute ('type', 'hidden');
                a_Field .setAttribute ('name', 'water');
                a_Field .setAttribute ('value', select_mode [i][1][0]);
                form .appendChild (a_Field );
                var  b_Field  = document .createElement ('input'); // save cookie of temperature
                b_Field .setAttribute ('type', 'hidden');
                b_Field .setAttribute ('name', 'temperature');
                b_Field .setAttribute ('value', select_mode [i][1][1]);
                form .appendChild (b_Field );
                var  c_Field  = document .createElement ('input'); // save cookie of time
                c_Field .setAttribute ('type', 'hidden');
                c_Field .setAttribute ('name', 'time');
                c_Field .setAttribute ('value', select_mode [i][1][2]);
                form .appendChild (c_Field );
                var  d_Field  = document .createElement ('input'); // save cookie of bathboms
                d_Field .setAttribute ('type', 'hidden');
                d_Field .setAttribute ('name', 'bathing');
                d_Field .setAttribute ('value', select_mode [i][1][3]);
                form .appendChild (d_Field );
                document .body .appendChild (form );
                form .submit ();
            }
            var  span  = document .createElement ('span');
            var  p  = document .createElement ('p');
            p .appendChild (document .createTextNode (select_mode [i][0]));
            span .appendChild (p );
            button .appendChild (span );
            button .style .backgroundRepeat  = "no-repeat"
            button .style .backgroundImage  = "url(" + select_mode [i][2] + ")";
            button .style .backgroundSize  = "cover";
            parent .appendChild (button ); //Set up CSS for each button
        }
    </script>
```

Online Mode Page


```
    <script  type ="text/javascript" src ="./js/jquery-3.5.1.js"></script>
    <script  type ="text/javascript" src ="./js/orgchart.js"></script>
    <script  type ="text/javascript"
            src ="https://cdnjs.cloudflare.com/ajax/libs/jquery-csv/0.71/jquery.csv-0.71.min.js"></script>
    <script>
        var  bathbomb  = new  Array ('사용안함', '큘', '바닐라', '라벤더', '레몬', '자스민',
                                     '로즈제라늄', '로즈마리', '블루마린', '일랑일랑');
        var  getCookie  = function (name ) {
            var  value  = document .cookie .match ('(^|;) ?' + name  + '=([^;]*)(;|$)');
            return  value  ? value [2] : null ;
        }; //get function of cookie
        var  age  = decodeURI (getCookie ('cookie_age'));
        var  gender  = decodeURI (getCookie ('cookie_gender'));
        document.querySelector('#name-content').innerHTML  = decodeURI (getCookie ('cookie_name'));
        document.querySelector('#age-content').innerHTML  = age ;
        document.querySelector('#gender-content').innerHTML  = gender ;
        document.querySelector('#job-content').innerHTML  = decodeURI (getCookie ('cookie_job'));
        document.querySelector('#predict-start-content').innerHTML=
        decodeURI(getCookie ('cookie_predict_start'));
        document .querySelector ('#predict-during-content').innerHTML  =
        decodeURI (getCookie ('cookie_predict_during'));
        document .querySelector ('#predict-temp-content').innerHTML  =
        decodeURI (getCookie ('cookie_predict_temp'));
        var  img  = document .querySelector ('#user-img'); // Change photos based on user age
        if (gender =='남성') { // change photos base on man fit man`s age
            if (age <19 ) img .style .backgroundImage ="url('./images/boy.png')";
            else  if (age <65 ) img .style .backgroundImage ="url('./images/fa.png')";
            else  img .style .backgroundImage ="url('./images/granfa.png')";
        }
        else  { // change photos base on women fit women`s age
            if (age <19 ) img .style .backgroundImage ="url('./images/girl.png')";
            else  if (age <65 ) img .style .backgroundImage ="url('./images/ma.png')";
            else  img .style .backgroundImage ="url('./images/granma.png')";
        }
        img .addEventListener ('click',function () {// Save the selected value cookie when you click on the
picture
            var  form  = document .createElement ('form');
            form .setAttribute ('method', 'POST');
            form .setAttribute ('action', '/item_cookie' );

            var  water ;
            if (age <16 ) water  = Math .floor (Math .random ()*10 +50 );
            else  water  = Math .floor (Math .random ()*20 +70 );
            var  water_filed  = document .createElement ('input') //save cookie of waterlevel
            water_filed .setAttribute ('type', 'hidden');
            water_filed .setAttribute ('name', 'water');
            water_filed .setAttribute ('value', water );
            form .appendChild (water_filed );
            var  bathing  = bathbomb [Math .floor (Math .random (9 ))];
            var  bathing_filed  = document .createElement ('input'); //save cookie of bathboms
            bathing_filed .setAttribute ('type', 'hidden');
            bathing_filed .setAttribute ('name', 'bathing');
            bathing_filed .setAttribute ('value', bathing );
            form .appendChild (bathing_filed );
            var  temp  = decodeURI (getCookie ('cookie_predict_temp'));
            var  temp_filed  = document .createElement ('input'); //save cookie of temperature
            temp_filed .setAttribute ('type', 'hidden');
```

```
            temp_filed .setAttribute ('name', 'temperature');
            temp_filed .setAttribute ('value', temp );
            form .appendChild (temp_filed );
            var  time  = decodeURI (getCookie ('cookie_predict_start'));//save cookie of timer
            var  time_filed  = document .createElement ('input');
            time_filed .setAttribute ('type', 'hidden');
            time_filed .setAttribute ('name', 'time');
            time_filed .setAttribute ('value', time );
            form .appendChild (time_filed );
            document .body .appendChild (form );
            form .submit ();
        })
    </script>
    <script  type ="text/javascript">
        $(function () {
            // Each title Name
            var  title  = [
                ['청소년', '중장년', '노년'],
                ['남자', '여자'],
                [],
                ['5~10도', '10~15도', '15~20도', '20~25도', '25~30도', '30~35도', '35~40도']
            ];
            var  bathbomb  = new  Array ('사용안함', '귤', '바닐라', '라벤더', '레몬', '자스민',
                                         '로즈제라늄', '로즈마리', '블루마린', '일랑일랑');
            function  csvreading (idx ) { // Calling up a csv file to generate user statistics
                $.get ("survey_data.csv", function (csvString ) { // Runs after page loading
                    var  csvdata  = $.csv .toArrays (csvString , {
                        onParseValue: $.csv .hooks .castToScalar
                    }); //Save the csv file as array
                    var  container  = document .getElementById ("chart-container");
                    var  rfy  = document .getElementById ("rfy");
                    container .style .display  = "block";
                    rfy .style .display  = "none";
                    var  arraylength  = csvdata .length ; // Data length for stored csv files
                    for  (var  i  = 1 ; i  < arraylength ; i ++) {
                        title [2].push (csvdata [i][8]);
                    }
                    // Remove duplicate job names
                    title [2] = title [2].filter ((item , index ) => title [2].indexOf (item ) == index );
                    var  str_size  = title [idx].length ; // titlesize
                    // array to store content for each title
                    // two-dimensional array (i is the title, j is the text output)
                    // 0 start time : as of 24 o'clock
                    // Time taken: 60 minutes
                    // Temperature: 1 coolness 2 lukewarm 3 warm
                    // Bath control status: 2 Not used 1
                    // 4 Title Size : Each Data Size
                    var  text_data  = new  Array (str_size );
                    for  (var  i  = 0 ; i  < str_size ; i ++) {
                        text_data [i] = new  Array (5 );
                        for  (var  j  = 0 ; j  < 5 ; j ++)
                            text_data [i][j] = 0 ;
                    }
                    //Calculate the entire data by rotating it all around
                    for  (var  i  = 1 ; i  < arraylength ; i ++) {
                        var  title_idx  = 0 ;
                        if  (idx  == 0 ) { // Obtain statistical data by age
                            if  (csvdata [i][2] < 20 ) title_idx  = 0 ;
                            else  if  (csvdata [i][2] < 65 ) title_idx  = 1 ;
                            else  title_idx  = 2 ;
                            text_data [title_idx][4] += 1 ;
                            text_data [title_idx][0] += csvdata [i][4];
                            text_data [title_idx][1] += csvdata [i][5];
                            if  (csvdata [i][3] == "cold") text_data [title_idx][2] += 1 ;
                            else  if  (csvdata [i][3] == "nomal") text_data [title_idx][2] += 2 ;
                            else  if  (csvdata [i][3] == "hot") text_data [title_idx][2] += 3 ;
                            if  (csvdata [i][7] == "유") text_data [title_idx][3] += 2 ;
                            else  if  (csvdata [i][7] == "무") text_data [title_idx][3] += 1 ;
                        } else  if  (idx  == 1 ) { //  Obtain statistical data by gender
                            if  (csvdata [i][1] == "남") title_idx  = 0 ;
                            else  if  (csvdata [i][1] == "여") title_idx  = 1 ;
                            else  console .log (i );
                            text_data [title_idx][4] += 1 ;
                            text_data [title_idx][0] += csvdata [i][4];
                            text_data [title_idx][1] += csvdata [i][5];
```

```
                        if  (csvdata [i][3] == "cold") text_data [title_idx][2] += 1 ;
                        else  if  (csvdata [i][3] == "nomal") text_data [title_idx][2] += 2 ;
                        else  if  (csvdata [i][3] == "hot") text_data [title_idx][2] += 3 ;
                        if  (csvdata [i][7] == "유") text_data [title_idx][3] += 2 ;
                        else  if  (csvdata [i][7] == "무") text_data [title_idx][3] += 1 ;
                    } else  if  (idx  == 2 ) { //Obtain statistical data by job
                        for  (var  j  = 0 ; j  < str_size ; j ++) {
                            if  (csvdata [i][8] == title [2][j]) {
                                title_idx  = j ;
                                break ;
                            }
                        }
                        text_data [title_idx][4] += 1 ;
                        text_data [title_idx][0] += csvdata [i][4];
                        text_data [title_idx][1] += csvdata [i][5];
                        if  (csvdata [i][3] == "cold") text_data [title_idx][2] += 1 ;
                        else  if  (csvdata [i][3] == "nomal") text_data [title_idx][2] += 2 ;
                        else  if  (csvdata [i][3] == "hot") text_data [title_idx][2] += 3 ;
                        if  (csvdata [i][7] == "유") text_data [title_idx][3] += 2 ;
                        else  if  (csvdata [i][7] == "무") text_data [title_idx][3] += 1 ;
                    } else  if  (idx  == 3 ) { // Obtain statistical data by weather
                        var  weather_data  = csvdata [i][9];
                        if  (weather_data  <= 10.00 ) title_idx  = 0 ;
                        else  if  (weather_data  <= 15.00 ) title_idx  = 1 ;
                        else  if  (weather_data  <= 20.00 ) title_idx  = 2 ;
                        else  if  (weather_data  <= 25.00 ) title_idx  = 3 ;
                        else  if  (weather_data  <= 30.00 ) title_idx  = 4 ;
                        else  if  (weather_data  <= 35.00 ) title_idx  = 5 ;
                        else  if  (weather_data  <= 40.00 ) title_idx  = 6 ;
                        text_data [title_idx][4] += 1 ;
                        text_data [title_idx][0] += csvdata [i][4];
                        text_data [title_idx][1] += csvdata [i][5];
                        if  (csvdata [i][3] == "cold") text_data [title_idx][2] += 1 ;
                        else  if  (csvdata [i][3] == "nomal") text_data [title_idx][2] += 2 ;
                        else  if  (csvdata [i][3] == "hot") text_data [title_idx][2] += 3 ;
                        if  (csvdata [i][7] == "유") text_data [title_idx][3] += 2 ;
                        else  if  (csvdata [i][7] == "무") text_data [title_idx][3] += 1 ;
                    }
                }
                // orgchart modification
                var  $this  = $(".node");
                $('.node').data ('node', $this );
                // Repeat for each title in the selected menu
                var  test ;
                for  (var  i  = 0 ; i  < str_size ; i ++) {
                    if (Math .floor (i /7 )==(i /7 )) test  = (i /7 );
                    else  test  += (1 +Math .floor (i /7 ));
                    var  nodeVals  = [];
                    var  real_idx  = i  < 2  ? 1  - i  : i ; //Because the child in front of you is 1-0
                    if  (i  >= 7 ) $('.node').data ('node',$($('.node')[test]));
                    else  $('.node').data ('.node',$this );
                    nodeVals .push (title [idx][real_idx]);
                    var  avg_text  = "";
                    if  (text_data [real_idx][2] == 0 ) avg_text  = '데이터없음';
                    else  {
                        text_data [real_idx][2]=(text_data [real_idx][2]/
                                                text_data[real_idx][4]).toFixed(2);
                        if  (text_data [real_idx][2] < 1.50 ) avg_text  = '시원함';
                        else  if  (text_data [real_idx][2] < 2.50 ) avg_text  = '미지근함';
                        else  if  (text_data [real_idx][2] >= 2.50 ) avg_text  = '따뜻함';
                    }
                    var  imagepic  = ""; // Image Output Class
                    if  (i  * 10  <= 20 ) imagepic  = 'boypic';
                    else  if  (i  * 10  <= 60 ) imagepic  = 'fapic';
                    else  imagepic  = 'granfapic';
                    var  code  = '<table class="t" id="t">\  //  Create  to  place  pictures  on  the
generated nodes
                    <colgroup>\
                        <col style="width:50%">\
                        <col style="width:50%">\
                    </colgroup>\
                    <tr>\
                        <td rowspan="10" class="imagine ' + imagepic  + '">\
                        </td>\
                    </tr>\
```

```
                    <tr>\
                        <td><b>평균시작시간</b></td>\
                    </tr>\
                    <tr>\
                        <td class="lb">' + (text_data [real_idx][0] == 0  ? '데이터없음 ' :
                                    (text_data [real_idx][0] / text_data [real_idx][4]).toFixed (0))
                                     + (text_data [real_idx][0] == 0  ? ' ' : '시') + '</td>\
                    </tr>\
                    <tr>\
                        <td><b>평균걸린시간</td>\
                    </tr>\
                    <tr>\
                        <td class="lb">' + (text_data [real_idx][1] == 0  ? '데이터없음' : (text_data
                                      [real_idx][1] / text_data [real_idx][4]).toFixed (2))
                                    + (text_data [real_idx][1] == 0  ? ' ' : '분') + '</td>\
                    </tr>\
                    <tr>\
1                       <td><b>평균온도</td>\
                    </tr>\
                    <tr>\
                        <td class="lb">' + avg_text  + '</td>\
                    </tr>\
                    <tr>\
                        <td><b>입욕제여부</td>\
                    </tr>\
                    <tr>\
                        <td class="lb">' + (text_data [real_idx][3] == 0  ? '데이터없음' :
                                      ((text_data [real_idx][3] / text_data [real_idx][4]) == 1
                                      ? bathbomb [0] : bathbomb [Math .floor (
                                       Math .random () * 9  + 1 )])) + '</td>\
                    </tr>\
                </table>';
                    var  $node  = $('.node').data ('node');
                    var  hasChild  = $node .parent ().attr ('colspan') > 0  ? true  : false ;
                    if  (!hasChild ) { // create child if child node does not exist in org chart
                        var  rel  = nodeVals .length  > 1  ? '110' : '100';
                        oc .addChildren ($node , nodeVals .map (function (item ) {
                            return  {
                                'name': item ,
                                'relationship': '110',
                                'id': getId (),
                                'title': code
                            };
                        }));
                    } else  { //create a child node in the same line as the child node in the org chart
                        oc .addSiblings ($node .closest ('tr').siblings ('.nodes').find
                                        ('.node:first'), nodeVals .map (function (item )  {
                            return  {
                                'name': item ,
                                'relationship': '110',
                                'id': getId (),
                                'title': code
                            };
                        }));
                    }
                }
            })
        };
        var  getId  = function () { //get node ID
            return  (new  Date ().getTime ()) * 1000  + Math .floor (Math .random () * 1001 );
        };
        // 초기화면 데이터
        var  datasource  = {
            'name': 'Age',
            'title': ''
        };
        var  oc  = $('#chart-container').orgchart ({
            'data': datasource ,
            'nodeContent': 'title',
            'chartClass': 'edit-state',
            'exportFilename': 'SportsChart',
            'createNode': function ($node , data ) {
                var  secondMenuIcon  = $('<i>', {
                    'class': 'oci oci-info-circle second-menu-icon',
                    click : function () {
```

```
                        $(this ).siblings ('.second-menu').toggle ();
                    }
                });
                var  secondMenu  = '<div class="second-menu">
                                    <img class="avatar" src="./images/' + 'boy' + '.png"></div>';
                $node .append (secondMenuIcon ).append (secondMenu );
            }
        });
        //reset org chart by clicking each button
        var  box  = document .getElementsByClassName ("content");
        box [0].style .height  = 0 ;
        $('#btn-chart1').on ('click', function (argument ) { //초기화
            oc .init ({
                'data': datasource
            });
            var  box  = document .getElementsByClassName ("content");
            box [0].style .height  = 0 ;
            csvreading (0 );
        });
        $('#btn-chart2').on ('click', function (argument ) { // Show statistics by gender
            var  data  = { 'name': 'Gender'};
            oc .init ({ 'data': data });
            var  box  = document .getElementsByClassName ("content");
            box [0].style .height  = 0 ;
            csvreading (1 );
        });
        $('#btn-chart3').on ('click', function (argument ) { //  Show statistics by job
            var  data  = { 'name': 'Job' };
            oc .init ({ 'data': data });
            var  box  = document .getElementsByClassName ("content");
            box [0].style .height  = 0 ;
            csvreading (2 );
        });
        $('#btn-chart4').on ('click', function (argument ) { //  Show statistics by weather
            var  data  = { 'name': 'Weather' };
            oc .init ({ 'data': data });
            var  box  = document .getElementsByClassName ("content");
            box [0].style .height  = 0 ;
            csvreading (3 );
        });
        $('#btn-chart5').on ('click', function (argument ) { //recommend for user through window
            var  container  = document .getElementById ("chart-container");
            var  rfy  = document .getElementById ("rfy");
            container .style .display  = "none";
            rfy .style .display  = "block";
        });
    });
</script>
```

Using Page

```
<script>
        var  slider  = document .querySelector ('.water-slider');
        slider .disabled  = true ;
        var  getCookie  = function (name ) {
            var  value  = document .cookie .match ('(^|;) ?' + name  + '=([^;]*)(;|$)');
            return  value ? value [2] : null ;
        }; // get cookie values
       // In the final run window, retrieve the data from the cookie for the output of your current set
value and output it to the screen
        var  water  = decodeURI (getCookie ('cookie_water')); //waterlevel
```

```
        var  bathing  = decodeURI (getCookie ('cookie_bathing')); //bathboms
        var  temperature  = decodeURI (getCookie ('cookie_temperature')); //temperature
        var  time  = decodeURI (getCookie ('cookie_time')); //using of time
        var  waterSlider  = document .querySelector ('.water-slider');
        waterSlider .value  = water ;
        var  waterFiled  = document .querySelector ('#water-value');
        waterFiled .innerText  = water ;
        var  bathingFiled  = document .querySelector ('#bathing-value');
        bathingFiled .innerText  = bathing ;
        var  temperatureFiled  = document .querySelector ('#temperature-value')
        temperatureFiled .innerHTML  = temperature  + '<span>&degC</span>';
        var  timeFiled  = document .querySelector ('#time-value');
        if (time  != '설정 안함') {
            time  += '분'
        }
        timeFiled .innerHTML  = time  ;
    </script>
```

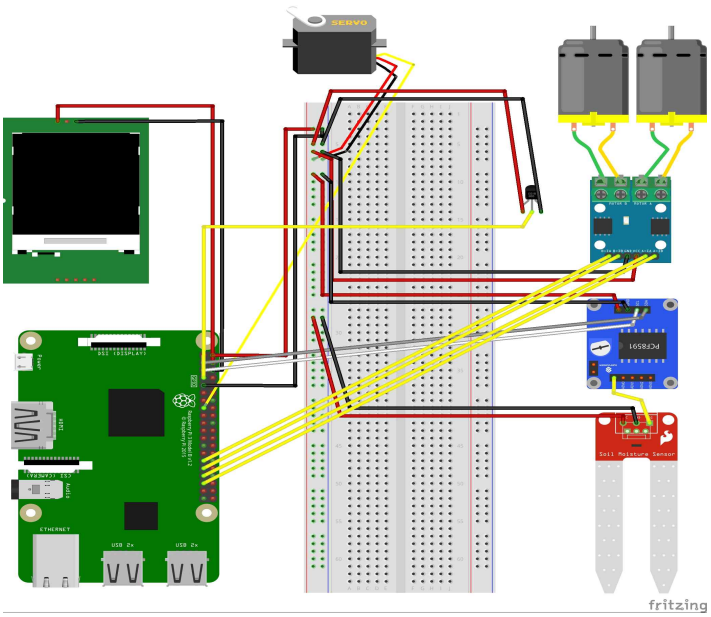
3-6) Parts for Prototype Implementation

In the early planning stage, it was planned to produce prototypes using a 3D printer, but it was decided that using materials such as acrylic directly had an advantage in terms of quality or ease of production, so it was produced by itself. As for the purchase of parts, the purchase was made with the support fund of the Link Business Group of Chosun University, and the details of the purchased parts are as follows.

	Cost	
	parts	estimate
List	RaspberryPi 4 Starter Kit * 2	210,000
	RaspberryPi4`s LCD touch screen 7inch	105,000
	socket jumper capel 40P * 3	2,550
	Soil moisture sensor module	1,500
	Motor driver module	1,800
	DC motor (5V)	1,200
	Step motor (5V)	1,800
	Waterproof type temperature sensor	3,900
	(2T, 285x425mm) * 3	12,300
	Transparent acrylic, shrinktube * 4	1,000
	포맥스 (2T, 300x450mm) * 3	6,300
Sum		347,350 (won)

(About details of Purchase List,please References Address : https://lincplus.chosun.ac.kr/main/)

❑ I-Tub Schematic



This is a picture of a schematic of a Raspberry pi prototype. The two water pumps are connected using a motor driver (L9110) to hold water in the bathtub model, and the soil humidity sensor is connected to measure the water level, but the ADC (PCF8591) is connected because the Raspberry Pi sends them to an analog signal value that it does not receive. Servomotor is a component for dropping bath bombs. And use a waterproof temperature sensor (DS18B20) to measure water temperature. All of these will be connected to the I-Tub service, which will be available using the touchscreen in the upper left corner. When the web page signals, the Raspberry Pi sensors start working at each value and stop when the setting is complete.

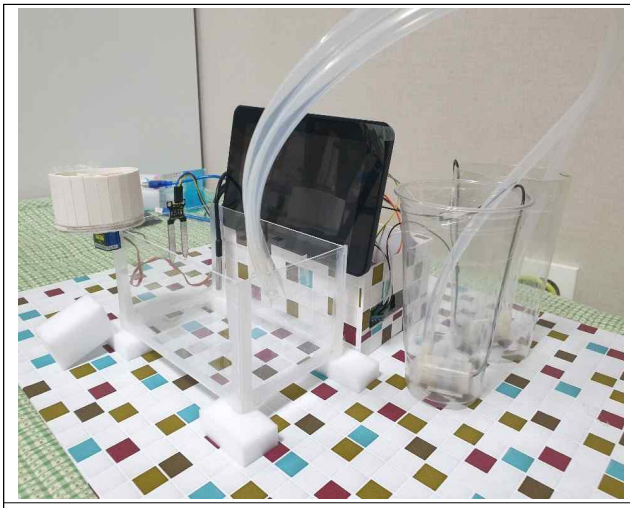
❑ I-Tub System Sensor Operation Code (I-tub_raspberryPi_code.py)

```
import  RPi.GPIO as  GPIO
import  time
import  spidev
import  smbus
import  os
import  glob
import  requests, json
import  pymysql
import  time

GPIO.setwarnings(False )
GPIO.setmode(GPIO.BCM)

# Save information for DB connections
db = pymysql.connect (
    user ='bf1138ba34c820',
    passwd  = '22ac05b88712768',
    host ='us-cdbr-iron-east-01.cleardb.net',
    db  = 'heroku_6295f565c172990',
    charset ='utf8'
)
# Settings for using the water level sensor
bus = smbus.SMBus(1)
add = 0x 48
idx = 0
switch = 17
GPIO.setup(switch, GPIO.IN)
os.system('modprobe w1-gpio')
os.system('modprobe w1-therm')

#Settings for using waterproof temperature sensors
base_dir = '/sys/bus/w1/devices/'
device_folder = glob.glob(base_dir + '28*')[0 ]
device_file = device_folder + '/w1_slave’

# Set GPIO pin of waterpump
IN1 = 19   #37 pin
IN2 = 13   #35 pin
IN3 = 6    #31 pin
IN4 = 5    #29 pin

GPIO.setup(IN1, GPIO.OUT)
GPIO.setup(IN2, GPIO.OUT)
GPIO.setup(IN3, GPIO.OUT)
GPIO.setup(IN4, GPIO.OUT)

# Set GPIO pin of servo motor
servo = 22

GPIO.setup(servo, GPIO.OUT)
p = GPIO.PWM(servo, 50 )
p.start(0 )

# servo motor angulation function
def  setServoPos (degree ):
    if  (degree > 180 ):
```

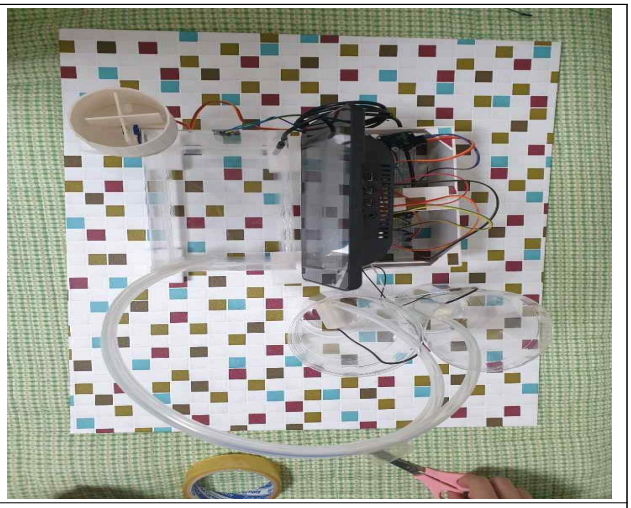
```
        degree = 180
    duty = 3  + (degree*(12 -3 )/180.0 )
    p.ChangeDutyCycle(duty)

# waterpump control function
def  setMotorContorl (INA , INB , stat ):
    #forward
    if  stat == 'FORWARD':
        GPIO.output(INA, GPIO.HIGH)
        GPIO.output(INB, GPIO.LOW)
    #behind
    elif  stat == 'BACKWORD':
        GPIO.output(INA, GPIO.LOW)
        GPIO.output(INB, GPIO.HIGH)
    #stop
    elif  stat == 'STOP':
        GPIO.output(INA, GPIO.LOW)
        GPIO.output(INB, GPIO.LOW)

# A function that prepares to output by reading values from a temperature sensor.
def  read_temp_raw ():
    f = open (device_file, 'r')
    lines = f.readlines()
    f.close()
    return  lines

# A function that takes the values ready to be printed and calculates them to the temperature.
def  read_temp ():
    lines = read_temp_raw()
    while  lines[0 ].strip()[-3 :] != 'YES':
        time.sleep(0.2 )
        lines = read_temp_raw()
    equals_pos = lines[1 ].find('t=')
    if  equals_pos != -1 :
        temp_string = lines[1 ][equals_pos+2 :]
        temp_c = float (temp_string) / 1000.0
        temp_f = temp_c * 9.0  / 5.0  + 32.0
        return  temp_c

# The part that processes the analog signals by reading them.
def  read_ADC ():
    bus.read_byte(add)
    analog = bus.read_byte(add)
    return  analog

# Various variables required for operation
servo_cnt = 0
start = 0
water = 0
temperature = 0

# SQL statement to connect to DB and verify that the user is ready for useDB
cursor = db.cursor(pymysql.cursors.DictCursor)
sql = "SELECT * FROM u_db WHERE u_last = 1"

# Main action statement and repeated through iteration
while  True :
    try :
        if  (start == 0 ):
            sql = "SELECT * FROM u_db WHERE u_last = 1"
            len  = cursor.execute(sql)
            for  val in  cursor:
                print (val)
                water = val['u_water']
                temperature = val['u_temperature']

            if  (len  > 0 ):
                setServoPos(90 )
                time.sleep(1 )
                setServoPos(0 )
                start = 1

        if  (len  > 0 ):
```

```
            # Activates temperature, water level and water pump sensor when user is ready to use
            bus.write_byte(add, 0x 00 )
            ain0 = read_ADC()
            setMotorContorl(19 , 13 , 'FORWARD')
            setMotorContorl(6 , 5 , 'FORWARD')
            print ("temperature value = %3d " % read_temp())
            print ("moisture value = %3d " % (ain0))

            # All sensors stop when the value is set by the user
            if  (ain0 >= water + 50 ):
                servo_con(0 )
                setMotorContorl(19 , 13 , 'STOP')
                setMotorContorl(6 , 5 , 'STOP')
                sql = 'UPDATE u_db set u_last=0 where u_last=1'
                cursor.execute(sql)
                start = 0
                db.commit()

        else :
            print ('sensor close')

        time.sleep(1 )
        db.commit()
    except  KeyboardInterrupt :
        GPIO.cleanup()
    except  Exception  as  e:
        GPIO.cleanup()
        print (e)
```

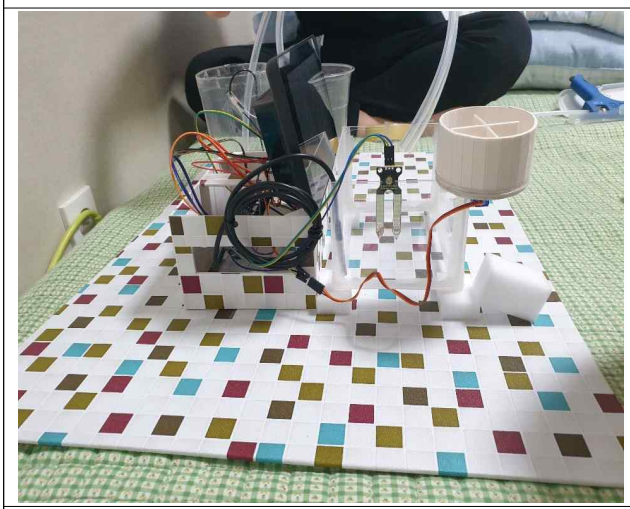
❑ Prototype View



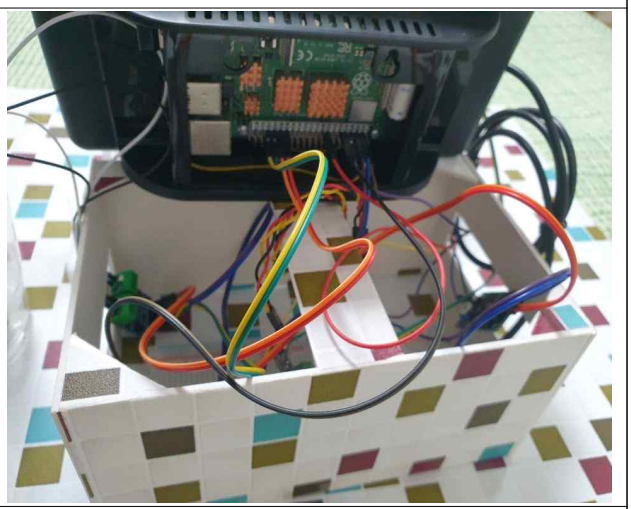
Whole View



View from above



View from side



View from behind

3-7) I-Tub Result and Supplement Points

(Reference to video from https://github.com/Ultimate-ItubProject/weekly_Report/tree/master/Final_report)

Demonstrations using manufactured web pages and I-Tub prototypes took place normally and worked normally as designed. Due to the small size of the bath bomb, the change in aroma and color does not appear much, so we will use a larger bath bomb to solve the problem later. And there was a limit to the complete control of the servo motor, which is operated by analog signals, from the raspberry pi, causing tremors. The problem will be solved by attaching a precise condenser, which will be supplemented later. Finally, in case of water pumps, there is a problem of severe heat generation and breakdown when using hot water, and part of the water pump will be replaced by water pumps that withstand high temperature, water through investigation.

Part of the data also had some complements, but internally, as users accumulate data each time they use it and predict it as a new learning model, without adding additional data, the more accurate and better the operating environment will be. However, the lack of reliability of the data using the data obtained from the survey is thought to be a complement. This part can be solved by asking more questions on professional survey sites and securing more samples.

4. Else

4-1) Felt During this Project

Lim Daein - I never dreamed of developing the web, but I decided to develop it after selecting and summarizing the project topic. I only knew the elements such as HTML table and body and had no knowledge at all, but I think it was a meaningful time to study and develop with my body. I knew only the terms web front-end, back-end, and didn't know what role it played and what it was representative of, but I felt like I was actually in charge of the company's project by designing a homepage and designing a css file and deciding on a web server to open a port so that everyone could come in and do the functions that they wanted. I felt what I lacked and filled it with the help of a team member who did not have enough time or understand it, and I think it was a more realistic class than any other class for four years. If there was anything to be desired, if the web server was a platform called DJango from the beginning, it would have naturally connected to the IoT sensor coding or machine learning, but it would have been very difficult to link Java with Python by choosing a platform called Node.js. However, through this experience, I felt that it was very important to analyze the program's characteristics, current trends, and what I wanted to do before the project began. Another regret was that the team member gave up halfway. Since he has relatively little knowledge of his major, he is more simple labor than coding, and I only assigned him to work and edit documents, which lowered his confidence and made me feel very sorry for making him think he made me think that this is not right for him. It's easy to give you this kind of role, so you won't complain, right? That's what I thought, but I think it was my own idea. But I think this experience will be a great help in dealing with the unexpected situation by getting a job. Thank you for your hard work during the semester!

Jeong Haemin - I worked with Seo Jeonguk in the front-end development part to create a web page that is directly visible to users. I didn't know much about the web until I started this Capstone project. Especially during the project, I had a lot of difficulties with the design and layout of web page elements using css. However, I learned necessary skills through lectures on life-coding websites, YouTube, and Googlling, and I overcame them through communication with team members. It was a very meaningful time for me to meet all the team members who are good and hard-working. At the beginning of the semester, after the non-face-to-face classes and face-to-face classes conducted by Corona, I felt sorry and regretful that I mainly communicated online due to the unavoidable circumstances of commuting to and from another school, and that I couldn't do more for the team.

Seo Jeonguk - I mainly took charge of the front end of the web in this project. I thought a lot about what I should look like when I was creating a web page that I could see for my users, and I also thought a lot about what additional features I could use. However, because of the lack of visual ability, we asked for help from our team members and asked them what to do at the back end, and we worked on the project together to fill each other's gaps. I felt a lot about this project. First of all, I had a hard time choosing a topic from the beginning of the project. When I tried to come up with an idea, I wondered if it was an item that already existed or if it was useful. And even before we started the project, we knew why the planning phase was important. At first, I didn't plan anything, and I didn't know what to do, and I didn't make any progress on the project. But as we were all planning and discussing what to do, I knew what to do, and as the project went on smoothly, I realized that pre-work was really important, not just jumping into the project. And in addition, I've talked to someone who's currently working at the back end a few times, and I thought I knew a little bit about the Web, but from the perspective of the worker, I thought, "You really don't know anything." It's good to start once, but he said he knows the whole concept. It's good to know a lot, but I didn't learn it in depth, only knowing vaguely. So this vacation, instead of learning new languages or skills, I want to set goals and study more deeply about what I want to do in the future.

Park Jisu - I am in charge of database design and prototype production. Even though I recently learned about the database, I had a lot of interest in it, so it was not perfect, but it was new for me to find myself available database. We're going to continue learning until we can implement a more complete database. It was also an auxiliary part of prototyping, but it was fun to make something new. I felt that even if it wasn't my main role in the project, the details of other roles had a significant impact on the performance of the role. I felt that developing something here requires everything from the basics to the application that I have learned, and that I can communicate with my team members and carry out easy team projects when this is supported. While working on this team project, I felt that even if I started from the same starting line as others, everyone had different abilities, efforts, goals, and mindset, but I lacked more in all aspects than other team members. During the three months of the team project, there were many things I didn't help and didn't work hard for the team members, so I want to show you a change in the team project that will come later. After this project is over, I want to re-organize and apply my basic skills again, and I want to have goals and resolutions in every aspect I do.

4-2) References

https://opentutorials.org/course/1 (to using javascript)
https://opentutorials.org/course/3780/18031 (html and css guide)
https://github.com/dabeng/OrgChart
(orgchart open source of to show statistical on online mode)
https://dabeng.github.io/OrgChart/ (orgchart example)
https://win.adrirobot.it/sensori/moisture_sensor/moisture_sensor.html
(soil mosture sensor example)
https://blog.naver.com/elepartsblog/221726825667 (rasbian install to use raspberry pi)
https://www.raspberrypi.org/ (rasbian install reference library to use raspberry pi)
https://www.circuitbasics.com/raspberry-pi-ds18b20-temperature-sensor-tutorial/
(emperature sensor example)
https://blog.naver.com/rhrkdfus/221373635978 rasbian wireless mouse solving an obstacle)
https://www.kocoafab.cc/fboard/view/1089 (understanding the principles of mini-breadboard)
https://www.inflearn.com/course/node-js-%EC%9B%B9%EA%B0%9C%EB%B0%9C#
(understanding node.js for webserver to handle user requests)
https://swiperjs.com/api/ (swiper API to use profile select page)