

# 개발 완료 보고서

(pH 시험 검정기)

## (Development Completion Report)

Project Name	ph시험기 개발			
변경 이력	ph 시험기 소프트웨어 개발 - 최초 작성			
	조직명	(주) 두원글로벌	전화번호	063-214-8556
	작성자	연구개발팀	승인자	박민규/팀장
	DATE	2022-09-02	Version	1.0

## 제정 및 개정 이력

[illegible]

## 용어 및 약어

NO	약어	원어	설명
1	HMI	Human Machine Interface	프로세서 시스템과 운영자 간의 인터페이스로, 기본적으로는 운영자 대시보드입니다.
2	DAQ	Data acquisition	센서와 컴퓨터를 이용해 전기적(전압, 전류), 화학적(온도, 압력, 음성 등) 신호를 측정하는 모든 행위를 의미한다.
3	TBC	To be confirmed	추후 확정
4	TBD	To be determined	추후 협의/추후 결정/미정
5	노드	node	온실통합제어기와 디바이스(센서-구동기) 간 통신을 가능하게 하는 매개체
6	센서	sensor	온실 내외의 환경 정보를 수집하는 장치
7	센서노드	sensor node	센서와 통신 모듈이 결합된 구조로써 측정된 센싱 값을 온실통합제어기에 전달하는 장치
8	구동기	actuator	농업 환경 또는 상태를 변경하기 위해 전기, 유공압, 기계적 신호 등에 따라 구동하는 장치
9	구동기 노드	actuator node	구동기와 통신 모듈이 결합된 구조로서, 온실 통합 제어기로부터 전달받은 메시지에 근거하여 구동기를 제어하는 기능이 있음.
10	복합노드	integrated node	센서, 구동기, 양액기, 디스플레이 등 이종 디바이스가 1개 이상 부착되고 통신 모듈이 결합된 구조로, 온실 통합 제어기로부터 전달받은 메시지에 근거하여 동작함
11	디바이스	device	온실 내외의 환경 정보를 수집하거나 생육 환경을 제어하는 장치로서, 그 역할에 따라 센서 혹은 구동기로 분류됨
12	노드규격	node specification	노드의 특성 정보(모델명, 제조사명, 타입 등) 및 통신을 위해 필요한 정보들이 기술되어 있는 규격으로 내부에 디바이스 규격들을 포함할 수도 있다.
13	디바이스 규격	device specification	센서, 구동기, 양액기, 디스플레이, 트리거 등과 같이 노드에 부착되어 제어기로부터 명령을 수행하거나 상태정보를 취득하는 등의 작업을 수행하기 위해 필요한 정보와 디바이스의 특성정보(모델명, 제조사명, 타입 등)를 기술하는 규격
14	PnP	Plug and Play	플러그 앤 플레이

## ■ References

- 아키텍처 패턴 :

<https://towardsdatascience.com/10-common-software-architectural-patterns-in-a-nutshell-a0b47a1e9013>

- KS X 3266:<sup>2022</sup> 스마트온실을 위한 센서 인터페이스

[https://www.rra.go.kr/ko/reference/kcsList\\_view.do?nb\\_seq=2283&cpage=4&nb\\_type=6&searchCon=&searchTxt=&sortOrder=](https://www.rra.go.kr/ko/reference/kcsList_view.do?nb_seq=2283&cpage=4&nb_type=6&searchCon=&searchTxt=&sortOrder=)

- KS X 3267:<sup>2022</sup> 기반 스마트온실 디폴트 센서노드 및 구동기 노드 개발

[https://www.rra.go.kr/ko/reference/kcsList\\_view.do?nb\\_seq=2284&cpage=4&nb\\_type=6&searchCon=&searchTxt=&sortOrder=](https://www.rra.go.kr/ko/reference/kcsList_view.do?nb_seq=2284&cpage=4&nb_type=6&searchCon=&searchTxt=&sortOrder=)

- KS X 3268:<sup>2022</sup> 스마트 온실 구동기 메타데이터

[https://www.rra.go.kr/ko/reference/kcsList\\_view.do?nb\\_seq=2285&cpage=4&nb\\_type=6&searchCon=&searchTxt=&sortOrder=](https://www.rra.go.kr/ko/reference/kcsList_view.do?nb_seq=2285&cpage=4&nb_type=6&searchCon=&searchTxt=&sortOrder=)

- KS X 3269:<sup>2022</sup> 스마트 온실 센서 메타데이터

[https://www.rra.go.kr/ko/reference/kcsList\\_view.do?nb\\_seq=2286&cpage=4&nb\\_type=6&searchCon=&searchTxt=&sortOrder=](https://www.rra.go.kr/ko/reference/kcsList_view.do?nb_seq=2286&cpage=4&nb_type=6&searchCon=&searchTxt=&sortOrder=)

# 목 차

1.개요 .....	1
1.1 범위 .....	1
1.2 목적 .....	1
1.3 시스템 개요 .....	1
1.4 개발범위 .....	2
1.4 관련 계획 및 표준 .....	3
2. 네트워크 모니터링 프로그램 .....	4
2.1 메인화면 기능 .....	5
2.2 메인화면 핵심 메소드 .....	6
3. 측정 프로그램 .....	7
3.1 대시보드 .....	8
3.2 레시피 관리 .....	10
3.3 레시피 등록 .....	11
3.4 레시피 수정 .....	12
3.5 측정관리 .....	13
3.6 의뢰자 관리 .....	17
3.7 의뢰자 수정 .....	18
3.8 의뢰자 등록 .....	19
3.9 표준시편 관리 .....	20
3.10 표준시편 수정 .....	21
3.11 표준시편 등록 .....	22
4. 분석 프로그램 .....	8
5. 고객지원 .....	9

## **1. 개요**

### **1.1 범위**

본 개발 완료 보고서는 pH검정기에 대한 명세를 기록한다. 요구사항은 센서노드의 요구사항이나 통신을 이용한 요구사항을 상세화한다.

### **1.2 목적**

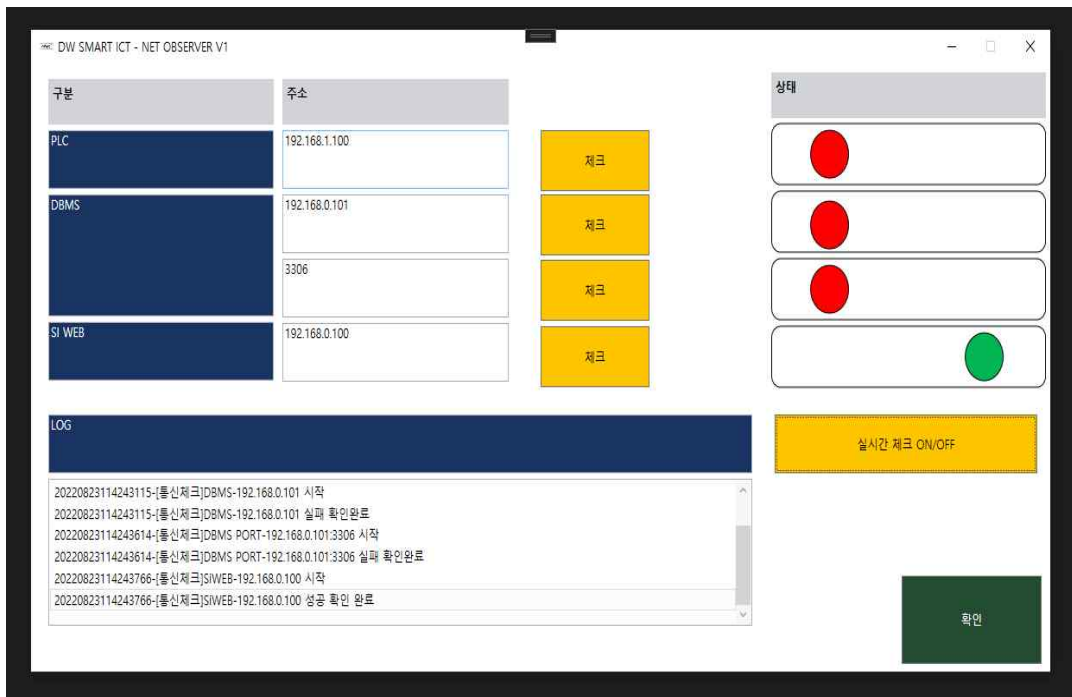
본 명세서의 목적은 센서(온도, 습도, CO2)의 아날로그 신호를 센서노드에서 획득하여, 디스플레이가 가능한 모니터링 시스템이며, 취득한 데이터를 DAQ에 KSX3267:2022을 표준으로 통신하여 보내는 기능을 가지고 있다.

### **1.3 시스템 개요**

pH 측정기 사진 업로드 예정(8/24)

## 1.4 개발범위

### 1.4.1 NETMON 모니터링 프로그램




### 1.4.2 pH 측정 프로그램



### 1.4.3 pH 분석 프로그램

DW SMART ICT - PH 분석 V1



2022-08-23 17:05:15

로그인

상태

로그아웃

2022-08-23 일간 측정이력 현황

NO	측정시작시간	측정완료시간	레시피명	의뢰자	평가	측정 AVG	측정 MIN	측정 MAX

2022-08-23 일간 알람 이력 현황

NO	상태	사용자	시간	알람 제목	알람 내용



## 1.5 관련 계획 및 표준

[illegible]

## 2. 네트워크 모니터링 프로그램

### 2.1 메인화면 기능



번호	기능 설명	비고
1	타이틀 - 프로그램명 & 버전 표시	
2	PLC 네트워크 상태 모니터링	
3	PLC 주소입력 - 자동으로 주소 읽어오기	값 변경 가능
4	PLC 네트워크 체크 - PING Test	
5	PLC 네트워크 상태 값	빨강:실패 / 주황:오류 / 초록:성공
6	DBMS 네트워크 상태 모니터링	
7	DBMS 주소입력 - 자동으로 주소 읽어 오기	값 변경 가능
8	DBMS 네트워크 체크 - PING Test	
9	DBMS 네트워크 상태 값	빨강:실패 / 주황:오류 / 초록:성공
10	DBMS 포트 입력 - 자동으로 읽어 오기	값 변경 가능

번호	기능 설명	비고
11	DBMS 네트워크 포트 체크 – TCPING Test	
12	DBMS TCPING 네트워크 상태 값	빨강:실패 / 주황:오류 / 초록:성공
13	통합시스템 네트워크 상태 모니터링	
14	통합시스템 주소 입력 – 자동으로 주소 읽어 오기	값 변경 가능
15	통합시스템 네트워크 체크 – PING Test	
16	통합시스템 네트워크 상태 값	빨강:실패 / 주황:오류 / 초록:성공
17	NET MON LOG	
18	실시간 네트워크 체크 ON/OFF – 프로그램 시작 시 자동 실행	
19	확인 – 트레이 상태로 최소화	

## 2.2 메인화면 핵심 메소드

### 1) PING(Paket Internet Groper) Test

```

public int SendToPing(String strIp)
{
    int iResult = 0;
    try
    {
        Ping pp = new Ping();
        PingOptions po = new PingOptions();

        po.DontFragment = true;

        byte[] bufData = Encoding.ASCII.GetBytes(""); //"pingTest"

        PingReply reply = pp.Send(IPAddress.Parse(strIp), 10, bufData, po);

        if (reply.Status == IPStatus.Success)
        {
            iResult = 1;
        }
        else
        {
            iResult = 0;
        }
        return iResult;
    }
    catch
    {
        throw;
    }
}

```

- 대상 컴퓨터를 향해 일정 크기의 패킷(네트워크 최소 전송 단위)을 보낸 후 응답하는 메시지를 수신하여 대상 컴퓨터 네트워크 확인 및 표시.

## 2) TCPING Test

```
참조 1개
public int SendToTcping(string strIp, string strPort)
{
    int iResult = 0;

    Socket socket = null;
    try
    {
        socket = new Socket(
            AddressFamily.InterNetwork,
            SocketType.Stream,
            ProtocolType.Tcp
        );

        socket.SetSocketOption(
            SocketOptionLevel.Socket,
            SocketOptionName.DontLinger,
            false
        );

        string input = strPort;
        //try
        //{
        int iPort = Int32.Parse(input);
        Console.WriteLine(iPort);

        IAsyncResult ret = socket.BeginConnect(strIp, iPort, null, null);
        iResult = ret.AsyncWaitHandle.WaitOne(100, true) ? 1 : 0;
        //}
        //catch (FormatException)
        //{
        //    iResult = 0;
        //}
    }
    catch { }
    finally
    {
        if (socket != null)
        {
            socket.Close();
        }
    }
    return iResult;
}
```

- PING Test와 유사, 특정 원격 포트 확인 및 표시.

### 3. 측정 프로그램

#### 3.1 대시보드



구분	설명	비고
1	메뉴 버튼, 클릭시 좌측 메뉴 탭 확장 및 축소	
2	회사 로고	
3	현재 시간	
4	아이디, 권한, 로그아웃 버튼	
5	메뉴 네비게이션	
6	일간 측정 통계	행 클릭시 7, 8출력
7	측정 결과 데이터(차트)	마우스 휠로 줌아웃, 및 드래그 가능
8	측정 결과 데이터(표)	NG: 빨간색, OK: 초록색
9	로그아웃	로그인 윈도우로 전환

### 3.1.1 대시보드 메서드

```
private void AnalyzeDayMeasureRow_Click(object sender, MouseEventArgs e)
{
    log.Info(wl.DayAnalyzeRowChooseEventLog);
    StdLowMinpoints.Clear();
    DataRowView AnalyzeDayMeasureRowChoose = (DataRowView)dgTableAnalyzeDayMeasure.SelectedItem;

    string starttime = ((DateTime)AnalyzeDayMeasureRowChoose.Row["mspacqstartdate"]).ToString("yyyy-MM-dd HH:mm:ss");
    string endtime = ((DateTime)AnalyzeDayMeasureRowChoose.Row["mspacqenddate"]).ToString("yyyy-MM-dd HH:mm:ss");

    try
    {
        string sql = $"select (@num:=@num+1) as no, mspresultevaluate, truncate(msmeasurevalue, 4) as mspmeasurevalue
        from (select @num:=0) as s, tbl_measurevo as t
        where t.mbno = {mbno} and t.mspacqstartdate between '{starttime}' and '{endtime}'";
        DataSet ds = msdto.SelectSqlToDataSet(sql, "TimeAccumStatusData");
        dgTableTimeAccumStatus.DataContext = ds;

        List<Dictionary<string, string>> rows = msdto.SelectSql2(sql, "no", "msmeasurevalue");

        foreach (Dictionary<string, string> row in rows)
        {
            StdLowMinpoints.Add(new DataPoint
            {
                xPos = Convert.ToDouble(row["no"]),
                yPos = Convert.ToDouble(row["msmeasurevalue"])
            });
        }
    }
    catch (Exception exc)
    {
        MessageBox.Show("일간 측정 데이터 로드 실패");
        log.Error(wl.DashBoardDayAnalyzeRowChooseLoadError);
    }
}
```

일간 통계를 보여주는 그리드의 한 행을 클릭했을 때 클릭한 행의 측정 상세 정보를 아래 차트와 데이터 그리드에 바인딩시켜 출력해 보여주는 코드

## 3.2 레시피 관리



구분	설명	비고
1	메뉴 네비게이션(레시피 관리 페이지)	
2	메뉴 네비게이션(측정관리 페이지)	
3	레시피 요약 정보	한 행 클릭시 상세정보(4) 출력
4	레시피 상세 정보	
5	레시피 수정 버튼	수정 버튼 페이지로 이동
6	레시피 삭제 버튼	
7	전체 보기 버튼	
8	검색 텍스트 입력 박스	
9	검색 버튼	입력한 검색 텍스트와 한 글자라도 같으면 검색하여 레시피 요약 정보(3)에 출력
10	레시피 측정 선택 버튼	레시피 리스트에서 한 레시피를 클릭 후 측정 관리 페이지로 이동
11	레시피 신규 등록 버튼	레시피 신규 등록 페이지로 이동

### 3.3 레시피 등록

구분	설명	비고
1	레시피 등록 텍스트 입력 창	
2	표준시편 선택 버튼	표준시편 관리 페이지로 이동
3	초기화 버튼	텍스트 박스 초기화
4	등록 버튼	레시피 등록 후 레시피 관리 페이지로 이동, 텍스트 박스에 입력 값이 없는 것이 있으면 에러 메시지 박스 출력
5	등록 취소 버튼	레시피 관리 페이지로 이동



### 3.4 레시피 수정

DW SMART ICT - PH 모듈 V1

**DOOWON GLOBAL**

22-08-22 14:42:50 ID0001 [관리자] 로그아웃

대시보드

TASK 관리

- 레시피관리

- 측정관리

PLC 연동관리

DBMS 연동관리

의뢰자 관리

표준시편 관리

PC 관리

설정

**1 레시피 수정**

1	레시피 명	rcpname_3
2	레시피 아이디	
3	레시피 기준값	5.5
4	오차(%)	2
5	시편 제조사	not set
6	표준시편 값	<b>2</b> 표준시편 선택

**3** 초기화

**4** 수정등록확인

**5** 수정취소

구분	설명	비고
1	레시피 수정 텍스트 입력 창	
2	표준시편 선택 버튼	표준시편 관리 페이지로 이동
3	초기화 버튼	텍스트 박스 초기화
4	수정등록 버튼	레시피 수정 후 레시피 관리 페이지로 이동, 텍스트 박스에 입력 값이 없는 것이 있으면 에러 메시지 박스 출력
5	수정 취소 버튼	레시피 관리 페이지로 이동

### 3.5 측정관리



구분	설명	비고
1	메뉴 네비게이션(레시피 관리 페이지)	
2	메뉴 네비게이션(의뢰자 관리 페이지)	
3	메뉴 네비게이션(의뢰자 관리 페이지)	
4	레시피 선택 버튼	레시피 관리 페이지(1)로 이동
5	의뢰자 선택 버튼	의뢰자 관리 페이지(2)로 이동
6	선택된 레시피 정보 출력	
7	레시피에 사용된 표준시편 정보 출력	
8	측정 반복 횟수 선택	1~50회 까지 설정 가능
9	측정 시작 버튼	
10	측정 중지 버튼	
11	측정 정보 출력(차트)	x축: 시간, y축: 측정 값
12	평가 결과 정보 출력	OK: 녹색, NG: 빨간색
13	측정 정보 출력	완료된측정/전체반복횟수, 측정시작시간, 진행률
14	측정 정보 출력	현재까지 측정한 최대, 최소, 평균, 편차
15	측정 정보 출력	측정 시작시간, 측정값

### 3.5.1 측정관리 측정 메서드

```
private void sampleGenerate(object sender, EventArgs e)
{
    if (errorFlag)
    {
        alarm.insertAlarm(mbno, alarm.aldivError, alarm.altitleMeasureMeasure, alarm.alcontentPLCFlagError);

        MessageBox.Show("PLC 플래그 에러");
        log.Error(wl.MeasureFlagError);
        timer.Stop();
        return;
    }
    //-plc start 2위치???

    //-읽어들 데이터 plc
    DWPLC_V1.Page1 tempPlc1 = new DWPLC_V1.Page1();
    double dMeasurementValueByPlcD224 = tempPlc1.GetDataD00224();
    double dMeasurementValueByPlcD226 = tempPlc1.GetDataD00266();

    //-data

    DateTime startTime = DateTime.Now;
    var rand = new Random();

    double stdValue = dMeasurementValueByPlcD224;

    double lowmin = 7 - 7 * 20 / 10;
    double lowmax = 7 + 7 * 20 / 100;
    double stdhigh = 10;
    double highmin = stdhigh - stdhigh * 20 / 100;
    double highmax = stdhigh + stdhigh * 20 / 100;

    StdLowMinpoints.Add(new DataPoint()
    {
        xPos = curXaxisVal,
        yPos = lowmin //5 - 5 * 20 / 100
    });
    StdLowMaxpoints.Add(new DataPoint()
    {
        xPos = curXaxisVal,
        yPos = lowmax //5 + 5 * 20 / 100
    });
    StdLowValpoints.Add(new DataPoint()
    {
        xPos = curXaxisVal,
        yPos = stdValue //5
    });
    StdHighMinpoints.Add(new DataPoint()
    {
        xPos = curXaxisVal,
        yPos = highmin //9 - 9 * 20 / 100
    });
}
```

```

StdHighMaxpoints.Add(new DataPoint()
{
    xPos = curXaxisVal,
    yPos = highmax//9 + 9 * 20 / 100
});
StdHighValpoints.Add(new DataPoint()
{
    xPos = curXaxisVal,
    yPos = stdhigh
});

double y = dMeasurementValueByPicD226;
MeasureValpoints.Add(new DataPoint()
{
    xPos = curXaxisVal,
    yPos = y
});

//---data 측정 데이터 최대/최소
if (y > max)
{
    max = y;
}
if (y < min)
{
    min = y;
}
//-----

string measureresult;

//--- data 평가
if (y < (rcpbenchmarkvalue - rcpbenchmarkvalue * (rcpmarginoferror / 100)) || y > (rcpbenchmarkvalue + rcpbenchmarkvalue * (rcpmarginoferror / 100)))
    measureresult = "NG";
else
    measureresult = "OK";

double sum = 0;

```

```

//--- 평균, 편차
if (MeasureValpoints.Count() == 1)
{
    stdDeviation = 0;
    avg = y;
}
else
{
    avg = (avg * (MeasureValpoints.Count() - 1) + y) / MeasureValpoints.Count();

    foreach (DataPoint dp in MeasureValpoints)
    {
        double yVal = dp.yPos;
        sum += Math.Pow(yVal - avg, 2);
    }
    stdDeviation = (Math.Sqrt(sum) / MeasureValpoints.Count());
}

//---data 후처리
//소수점 5자리부터 버림
max = Math.Truncate(max * 10000) / 10000;
min = Math.Truncate(min * 10000) / 10000;
avg = Math.Truncate(avg * 10000) / 10000;
stdDeviation = Math.Round(stdDeviation * 10000) / 10000;

tbxMeasureValueMax.Text = Convert.ToString(max);
tbxMeasureValueMin.Text = Convert.ToString(min);
tbxMeasureValueAvg.Text = Convert.ToString(avg);
tbxMeasureValueDeviation.Text = Convert.ToString(stdDeviation);

//데이터 측정 결과 값
if (!measureStartCheck)
{
    tbxMeasureStartTime.Text = startTime.ToString("yyyy-MM-dd HH:mm:ss");
    measureStartCheck = true;
}

CurMeasureData.Add(new CurMeasureValue()
{
    measurestarttime = startTime.ToString("yyyy-MM-dd HH:mm:ss"),
    //measurevalue = Math.Round(y, 2).ToString(),
    measurevalue = Math.Round(y, 3).ToString()
});

tbxProcessPercent.Text = Convert.ToString(((totalMeasurelter - measurelter) * 60 + MeasureValpoints.Count) * 100 / (totalMeasurelter * 60)) + "%";

dgTableCurMeasureData.ItemsSource = CurMeasureData;
dgTableCurMeasureData.ScrollIntoView(CurMeasureData[CurMeasureData.Count() - 1]);

```

```

//-- do insert
try
{
    string sql = $"insert into tbl_measurevo (mbo, regbno, rcpno, rcpid, mspid, mspmeasurevalue, mspmeasuremax, mspmeasuremin, mspmeasurestddev, mspresultevaluate, mspacstartdate, mspacenddate)
    values ({mbo}, {regbno}, {rcpno}, {rcpid}, {rcpid}, {totalMeasurelter - measurelter}, {tbxMeasureStartTime.Text}, {y}, {max}, {min}, {stdDeviation}, {measureresult}),
    ({startTime.ToString( "yyyy-MM-dd HH:mm:ss")}, {startTime.AddSeconds(1).ToString( "yyyy-MM-dd HH:mm:ss")})";

    msdto.RunSql("INSERT", sql);
}
catch (Exception exc)
{
    MessageBox.Show("측정 데이터 삽입 실패");
    log.Error(wl.MeasureDataInsertError);
    timer.Stop();
    msdto.Close();
    return;
}

curAxisVal++;

//-- last 측정 후 통계do insert
if (MeasureValpoints.Count() == 60)
{
    //-- do read stop
    //-----

    if (avg < (rcpbenchmarkvalue - rcpbenchmarkerror / 100) || avg > (rcpbenchmarkvalue + rcpbenchmarkerror / 100))
    {
        measureresult = "NG";
    }
    else
    {
        measureresult = "OK";
    }

    try
    {
        string sql = $"insert into tbl_msresult_statisticsvo (mbo, regbno, regbid, rcpid, rcpname, mspid, mspacstartdate, mspacenddate, mspmeasureavg, mspmeasuremax, mspmeasuremin, mspresultevaluate)
        values ({mbo}, {regbno}, {rcpid}, {tbxChoseRecipeId.Text}, {tbxChoseRecipeName.Text}, {rcpid}, {totalMeasurelter - measurelter}, {tbxMeasureStartTime.Text}, {CurMeasureData[59].measurestarttime},
        {avg}, {max}, {min}, {measureresult})";

        msdto.RunSql("INSERT", sql);
    }
    catch
    {
        MessageBox.Show("측정 결과 데이터 삽입 실패");
        log.Error(wl.MeasureResultDataInsertError);
        timer.Stop();
        msdto.Close();
        return;
    }

    brdFinalEvaluationOK.Background = (SolidColorBrush)new BrushConverter().ConvertFrom("Gray");
    brdFinalEvaluationNG.Background = (SolidColorBrush)new BrushConverter().ConvertFrom("Gray");

    if (measureresult == "OK")
    {
        brdFinalEvaluationOK.Background = (SolidColorBrush)new BrushConverter().ConvertFrom("Green");
    }
    else
    {
        brdFinalEvaluationNG.Background = (SolidColorBrush)new BrushConverter().ConvertFrom("Red");
    }
}

```

```

curAxisVal = 0;
max = 0;
min = 999.000;
measureStartCheck = false;

measurelter--;
tbxMeasurelterCount.Text = $"{totalMeasurelter - measurelter}/{totalMeasurelter}";

if (measurelter == 0)
{
    MessageBox.Show(" 측정이 완료 되었습니다. ");
    tbxMeasureStartTime.Text = "측정이 완료 되었습니다. ";
    timer.Stop();
    msdto.Close();

    log.Info(wl.MeasureEndEventLog);
}
else
{
    CurMeasureData.Clear();

    StdLowMinpoints.Clear();
    StdLowMaxpoints.Clear();
    StdLowValpoints.Clear();
    StdHighMinpoints.Clear();
    StdHighMaxpoints.Clear();
    StdHighValpoints.Clear();
    MeasureValpoints.Clear();
}
}
}

```

측정 데이터를 PLC에서 1초마다 읽어와 DB에 삽입, 1분 단위로 측정이 되고 1분이 되면 1분 동안 측정된 데이터의 통계를 DB에 삽입

### 3.6 의뢰자 관리

DOOWON GLOBAL 22-08-22 13:47:07 ID0001 [관리자] 로그아웃

의뢰자리스트

구분	등록일자	의뢰자명	부서	등급	연락처1	연락처2	모바일번호	담당공제	의뢰자 선택	의뢰자 수정	의뢰자 삭제
REQ0001	7/22/2022 4:21:14 PM	두원회사		2	82 - 063 - 214 - 855			0	선택	수정	삭제
REQ0002	7/22/2022 6:21:56 PM	DW회사		123	82-063-123-1234			0	선택	수정	삭제
REQ0003	7/22/2022 6:22:16 PM	회사3		1	82-02-123-1234			0	선택	수정	삭제
and	8/4/2022 4:20:52 PM	and	and	1	--	--	1	8/4/2022 4:21:11 PM	선택	수정	삭제
	8/4/2022 4:59:13 PM			--	--	--	1	8/5/2022 1:26:35 PM	선택	수정	삭제
	8/5/2022 8:41:50 AM			--	--	--	0		선택	수정	삭제

1. 측정관리 2. 의뢰자 관리 3. 의뢰자 선택 4. 의뢰자 수정 5. 의뢰자 삭제 6. 검색 7. ID 8. 검색 9. 초기화 10. 의뢰자 신규등록

구분	설명	비고
1	메뉴 네비게이션(측정 관리 페이지)	
2	메뉴 네비게이션(의뢰자 관리 페이지)	
3	의뢰자 선택 버튼	의뢰자 선택 후 측정 관리 페이지(1)로 이동
4	의뢰자 수정 버튼	의뢰자 수정 페이지로 이동
5	의뢰자 삭제 버튼	
6	검색 내용 입력	
7	검색 내용 선택	ID, 기관명, 부서, 등급
8	의뢰자 검색 버튼	
9	초기화 버튼	비활성
10	의뢰자 신규등록 버튼	의뢰자 신규 등록 페이지로 이동

### 3.7 의뢰자 수정

DOOWON GLOBAL

22-08-22 13:47:49 ID0001 [관리자] 로그아웃

대시보드

TASK 관리

- 레이피관리

- 측정관리

PLC 연동관리

DBMS 연동관리

의뢰자 관리

표준시편 관리

1

의뢰자 ID REQ00011

종류 2

소속기관 두원회사

부서

연락처1 82 - 063 - 214 - 855

연락처2

2 수정

3 취소

구분	설명	비고
1	의뢰자 정보 입력 창	수정할 의뢰자의 정보가 미리 입력돼 있음
2	수정 버튼	수정 후 의뢰자 관리 페이지로 이동
3	취소 버튼	의뢰자 관리 페이지로 이동

### 3.8 의뢰자 등록

구분	설명	비고
1	의뢰자 정보 입력 창	
2	등록 버튼	등록 후 의뢰자 관리 페이지로 이동
3	취소 버튼	의뢰자 관리 페이지로 이동



### 3.9 표준시편 관리

DOOWON GLOBAL

22-08-22 15:17:02 ID0001 [관리자] 로그아웃

표준시편(PH)

구분번호	ID	샘플명	온	인덕시타	최대	최소	적용온도	단위	등록일자	삭제일자	사용유형	사용여부	표준시편 관리	표준시편 수정	표준시편 삭제
PH11	IDPH11	PH11NAME	9	3000	9.9	8.1	10	pH	7/22/2022 6:35:55 PM	8/3/2022 11:43:26 AM	1	선택	수정	삭제	
PH	IDPH10	PH10NAME	10	10000	14	8	20	pH	7/22/2022 6:36:12 PM	8/3/2022 11:43:43 AM	1	선택	수정	삭제	

선택된 표준시편

구분번호	IDPHS	PHNAME	온	인덕시타	최대	최소	적용온도	단위	등록일자	삭제일자	사용유형	사용여부	표준시편 관리	표준시편 수정	표준시편 삭제
PH11	IDPH11	PH11NAME	9	3000	9.9	8.1	10	pH	7/22/2022 6:35:55 PM	8/3/2022 11:43:26 AM	1	선택	수정	삭제	

표준시편 신규 등록

표준시편을 레시피로 등록

구분	설명	비고
1	표준시편 정보 출력	
2	표준시편 선택 버튼	선택된 행 (8)로 내용 복사
3	표준시편 수정 버튼	표준 시편 수정 페이지로 이동
4	표준시편 삭제 버튼	
5	표준시편 검색 내용 선택	샘플명, 구분번호, 아이디
6	표준시편 검색 내용 입력	
7	표준시편 검색 버튼	
8	선택된 표준시편 정보 출력	
9	선택 취소 버튼	선택된 표준시편에서 삭제
10	표준시편 신규등록 버튼	표준시편 신규 등록 페이지로 이동
11	선택된 표준시편 등록	선택된 표준시편 정보 레시피 등록, 수정 페이지에 저장

### 3.10 표준시편 수정

DW SMART ICT - PH 측정 V1

22-08-22 13:49:59 ID0001 [관리자] 로그아웃

1

구분번호	PH
샘플 아이디	IDPH5
샘플명	PH5NAME
값	5
인텐시티	5000
허용오차	20
단위	pH

2 초기화 3 수정 등록 4 취소

구분	설명	비고
1	표준시편 입력 창	수정 할 표준시편 정보가 미리 입력 되어 있음
2	초기화 버튼	표준시편 입력 창 초기화
3	수정 등록버튼	수정 후 표준시편 관리 페이지로 이동
4	취소 버튼	표준시편 관리 페이지로 이동

### 3.11 표준시편 등록

DW SMART ICT - PH 측정 V1

DOOWON GLOBAL

22-08-22 13:50:25 ID0001 [관리자] 로그아웃

1

구분번호

샘플 아이디

샘플명

값

인텐시티

허용오차

단위

2 초기화

3 등록

4 취소

대시보드

TASK 관리

- 레시피관리

- 측정관리

PLC 연동관리

DBMS 연동관리

의뢰자 관리

표준시편 관리

표준시편 등록

표준시편 관리

구분	설명	비고
1	표준시편 등록 입력 창	
2	초기화 버튼	표준시편 등록 입력 창 내용 초기화
3	표준시편 등록 버튼	표준시편 등록 후 표준시편 관리 페이지로 이동
4	취소 버튼	표준시편 관리 페이지로 이동

## 5. 보안 요구사항

요구사항식별자	요구 사항	출처	비고

## 6. 품질 요구사항

품질속성	품질 요구사항 이름	설명	측정요소
유지보수성 (Maintenainability)	신규 포맷 데이터	관리자가 새로운 포맷으로 데이터를 보고 싶을 경우, 시스템 유지보수자가 1일 이내에 이를 시스템에 반영할 수 있어야 한다.	1일 이내
재사용성 (Reusability)			

## 7. 스마트팜ICT 표준 채택

본 센서노드 시스템은 표준 규격이 정하는 데이터 송수신 사양을 준수한다.