



FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

IMPLEMENTACIÓN DE UN PROTOTIPO DE UN SISTEMA DE CONTROL,
MONITOREO Y GESTIÓN DE ALARMAS PARA LA ELABORACIÓN DE LA
CERVEZA ARTESANAL.

Autor

Carlos Andrés Yajamin Guañuna

Año
2018



FACULTAD DE INGENIERÍAS Y CIENCIAS APLICADAS

IMPLEMENTACIÓN DE UN PROTOTIPO DE UN SISTEMA DE CONTROL,
MONITOREO Y GESTIÓN DE ALARMAS PARA LA ELABORACIÓN DE LA
CERVEZA ARTESANAL.

Trabajo de Titulación presentado en conformidad con los requisitos
establecidos para optar por el título de Ingeniero en Redes y
Telecomunicaciones.

Profesor Guía

Mg. Johanna Rafaela Ortega Briones

Autor

Carlos Andrés Yajamin Guañuna

Año

2018

DECLARACIÓN DEL PROFESOR GUIA

“Declaro haber dirigido el trabajo, Implementación de un prototipo de un sistema de control, monitoreo y gestión de alarmas para la elaboración de la cerveza artesanal, a través de reuniones periódicas con el estudiante Carlos Andres Yajamin Guañuna, en el semestre 2018-2, orientando sus conocimientos y competencias para un eficiente desarrollo del tema escogido y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación”

Johanna Rafaela Ortega Briones
Magister en Telecomunicaciones
C.I. 171457898-4

DECLARACIÓN DEL PROFESOR CORRECTOR

“Declaro haber revisado este trabajo, Implementación de un prototipo de un sistema de control, monitoreo y gestión de alarmas para la elaboración de la cerveza artesanal, del Carlos Andres Yajamin Guañuna, en el semestre 2018-2, dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación”

Iván Ricardo Sánchez Salazar
Magister en Calidad Seguridad y Ambiente
C.I. 1803456142

DECLARACIÓN DE AUTORÍA DEL ESTUDIANTE

“Declaro que este trabajo es original, de mi autoría, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes”

Carlos Andrés Yajamin Guañuna

C.I. 1717748113

AGRADECIMIENTOS

Agradezco a Dios por brindarme la oportunidad de llegar a esta etapa de mi vida, y al apoyo incondicional de mi familia, mi novia y de todos mis amigos.

.

RESUMEN

En el presente proyecto de titulación, se diseñará e implementará un prototipo automatizado para un sistema de elaboración de cerveza artesanal en el cual se realizará un monitoreo mediante la gestión de alarmas las cuales realizan el envío de mensajes de texto de acuerdo a la etapa de elaboración en la cual se encuentre el proceso de fabricación, esto con la finalidad de optimizar el tiempo empleado en la elaboración del producto, de tal manera que el operador pueda desarrollar otras actividades inherentes al proceso y únicamente a partir de recibir dichos mensajes de texto acudir a realizar las acciones necesarias para continuar con el proceso de elaboración de la cerveza artesanal.

Al desarrollar un sistema automatizado de control y envío de alertas al operador se consigue disminuir los tiempos efectivos que el mismo debe dedicar a cada etapa de producción, reduciendo así a largo plazo el tiempo de fabricación del producto y además una economización de la materia prima invertidos en cada proceso, debido a que se reduce notablemente los errores por factores humanos.

El proyecto consta de un sistema de control y gestión de alarmas desarrollado en Arduino mediante el envío de mensajes de texto a través de un módulo GSM, sensores de temperatura cuyo encapsulado cumpla con características de impermeabilidad y sean capaces de trabajar a elevadas temperaturas, sensor ultrasónico programado de tal manera que este nos proporcione una medición del nivel de líquido en una de las calderas del sistema, todos estos elementos compatibles con el controlador antes mencionado, así como un software desarrollado en MYSQL en el cual se registrarán las variables del proceso, además de mostrar gráficos de tendencias en el desarrollo del proceso y guardar dichos valores en una base de datos, los cuales son necesarios para realizar comparaciones y desarrollar nuevas recetas mejorando cada vez los procesos de elaboración y la optimización de recursos. Es necesario citar que todo el entorno de programación del presente proyecto se lo desarrollo en plataforma de software libre, lo cual no representa ningún

inconveniente en la obtención de licencias para el desarrollo de los algoritmos necesarios.

ABSTRACT

It is designed and implemented an automated prototype system brewing craft in which monitoring is performed by managing alarms which perform sending text messages according to the processing stage in which it is found the process manufacturing, this in order to optimize the time used in the preparation of the product, in such a way that the operator can develop other activities inherent to the process and only from receiving these text messages go to take the necessary actions to continue with the process of craft brewing.

By developing an automated control system and sending alerts the operator is able to reduce the actual time that it should devote to each stage of production, reducing long-term manufacturing time of the product and also an economization of raw material invested in each process, because errors are significantly reduced by human factors.

The project consists of an alarm control and management system developed in Arduino by sending text messages through a GSM module, temperature sensors which meets encapsulated impermeability characteristics and are capable of working at high temperatures, Ultrasonic sensor programmed in such a way that it provides us with a measurement of the liquid level in one of the boilers of the system, all these elements compatible with the controller above, as well as a software developed in MYSQL in which the process variables will be registered, In addition to showing charts trends in the development process and store these values in a database, which are necessary to make comparisons and develop new recipes, improving each time the elaboration processes and the optimization of resources. It is necessary to mention that the entire programming environment of this project is developed on a free software platform, which poses no problem in obtaining licenses for the development of the necessary algorithms.

INDICE

1 Capítulo I. Introducción	1
1.1 Alcance	1
1.2 Justificación	2
1.3 Objetivos	3
1.3.1 Objetivo General.....	3
1.3.2 Objetivos específicos.....	3
2 Capítulo II. Marco Teórico.....	4
2.1 Estudio sobre el arte de elaboración.....	4
2.1.1 Definición.....	4
2.1.2 Historia.	4
2.2 Componentes.....	5
2.2.1 Cebada / Malta	5
2.2.2 Malteado	6
2.2.3 Lúpulo	6
2.2.4 Levadura	7
2.2.5 Agua.....	8
2.3 Tipos de cervezas.....	8
2.3.1 Ales	8
2.3.2 Ordinary Bitter	8
2.3.3 American Pale Ale	8
2.3.4 Oatmeal Stout	8
2.4 Diferenciadores respecto a la cerveza Industrial.....	9
2.5 Aspectos técnicos.....	9

2.5.1	Tratamiento del agua.....	9
2.5.2	Trituración de granos.....	10
2.5.3	Cocción del mosto.....	10
2.5.4	Recirculación y muestreo.....	10
2.5.5	Macerado.....	10
2.5.6	Enfriamiento.....	10
2.5.7	Fermentación.....	11
3	Capítulo III. Diseño e implementación.....	12
3.1	Adquisición de datos.....	13
3.1.1	Monitoreo.....	13
3.1.2	Control.....	13
3.1.3	Interfaz Gráfica.....	13
3.1.4	Base de datos.....	14
3.2	Análisis de componentes electrónicos.....	14
3.2.1	Microcontrolador.....	14
3.2.2	Modulo Gsm.....	15
3.2.3	Termocupla.....	15
3.2.4	Sensor de nivel.....	16
3.2.5	Bomba.....	17
3.2.6	Electroválvula.....	18
3.3	Diseño de hardware.....	18
3.3.1	Dimensionamiento de calderas.....	20
3.3.2	Arduino Mega 2650.....	22
3.3.3	GPRS/GSM Shield SIM900.....	24
3.3.4	Termocupla DS18B20.....	25
3.3.5	Sensor ultrasónico HC-SR04.....	26

3.3.6	Módulo Relé 16 Canales 5V Optoacoplador	27
3.3.7	Bomba BYT-7A006.....	29
3.3.8	Electroválvula HFS 12V DC.....	30
3.3.9	Modulo convertidor DC-DC XI6009.....	31
3.3.10	Software Arduino IDE	32
3.4	Sistema de control	34
3.5	Proceso de implementación.....	37
3.6	Diagrama de conexión.....	39
3.7	Diseño de Software	41
3.7.1	Software Microsoft Visual Studio 2017	41
3.7.2	Diseño del programa con interfaz gráfica.	42
3.7.3	Base de datos.	46
4	Capítulo IV Protocolo de pruebas.....	47
4.1	Registro de datos.....	48
4.2	Captura de datos	48
4.2.1	Monitoreo de temperatura.....	48
4.3	Gestión Funcional del prototipo electrónico.....	51
4.3.1	Mezclado	51
4.3.2	Recirculación.....	51
4.3.3	Cambio de caldera	52
4.3.4	Cambio de temperatura.....	53
4.3.5	Paro de emergencia	53
4.3.6	Gestión de alarmas	54
4.3.7	Consulta de etapas.....	56
4.4	Propuesta para sistema de respaldo	59

5 Capítulo V Guía de usuario.....	60
5.1 Procedimientos.....	60
6 CONCLUSIONES Y RECOMENDACIONES	65
6.1 Conclusiones	65
6.2 Recomendaciones.....	66
REFERENCIAS	68
ANEXOS.....	72

ÍNDICE DE FIGURAS

Figura 1 Cebada de dos carreras	6
Figura 2 Pellet de Lúpulo	7
Figura 3 Diagrama de procesos general	11
Figura 4 Diagrama de interconexión.....	12
Figura 5 Diagrama en bloques.....	13
Figura 6 Diagrama de los componentes de hardware	19
Figura 7 Diagramas de equipos.....	20
Figura 8 Dimensionamiento de los calderos	21
Figura 9 Cocina Industrial	22
Figura 10 Arduino Mega.....	24
Figura 11 GPRS/GSM Shield SIM900.....	25
Figura 12 Termocupla DS18B20	26
Figura 13 Sensor ultrasónico HC-SR04	27
Figura 14 Módulo Relé 16 Canales 5V Optoacoplador	28
Figura 15 Bomba BYT-7A006.....	30
Figura 16 Electroválvula HFS 12v DC.....	31
Figura 17 Módulo convertidor DC-DC XI6009.....	32
Figura 18 Interfaz principal Arduino IDE.....	32
Figura 19 Diagrama de flujo de control.	36
Figura 20 Diagrama de funcionamiento	38
Figura 21 <i>Diagrama de conexiones</i>	40
Figura 22 Detalles del programa.....	41
<i>Figura 23</i> Interfaz de ingresar Receta	43
Figura 24 Interfaz de registro y monitoreo de la Temperatura de la Caldera 1	43

Figura 25 Interfaz de registro y monitoreo de la Temperatura de la Caldera 2	44
Figura 26 Interfaz de registro de fermentación.	45
Figura 27 Interfaz de gráfica de consulta de receta.	45
Figura 28 Interfaz de gráfica de consulta de caldero 1 y 2	46
Figura 29 Tablas de MySQL	47
Figura 30 Registro de datos.....	48
Figura 31 Primera etapa de maceración	49
Figura 32 Segunda etapa de maceración	49
Figura 33 Tercera etapa de maceración	50
Figura 34 Etapa de hervido.....	50
Figura 35 Accionamiento del mezclador.....	51
Figura 36 Recirculación	52
Figura 37 Cambio de caldera.....	52
Figura 38 Cambio de temperatura	53
Figura 39 Paro de emergencia	54
Figura 40 Notificaciones del proceso de maceración	55
Figura 41 Notificaciones del proceso de hervido.	56
Figura 42 Notificaciones del paro de emergencia.....	56
Figura 43 Consulta de receta ingresada.	57
Figura 44 Gráfico de la temperatura del proceso de maceración antes del prototipo.....	58
Figura 45 Gráfico de la temperatura del proceso de maceración. ..	58
Figura 46 Grafico del proceso de hervido	59
Figura 47 Sistema Energizado y Listo.....	60
Figura 48 Búsqueda de puerto serial.....	61
Figura 49 Conexión de puerto serial.....	62
Figura 50 Ingreso de receta	62

Figura 51 Ingreso de procedimientos caldera 1	63
Figura 52 Ingreso de procedimientos caldera 2	63
Figura 53 Ingreso de parámetros de fermentación	64
Figura 54 Consulta de Receta	64
Figura 55 Consulta de calderas	65

ÍNDICE DE TABLAS

Tabla 1 Diferencias: cerveza artesanal e industrial	9
Tabla 2 Comparación entre Arduino Mega y Logo 8	14
Tabla 3 Comparación entre SIM900 y SIM800L	15
Tabla 4 Comparación entre Driver max6675 y DS18B20	15
Tabla 5 Comparación entre HC-SR04 Y K-0135	16
Tabla 6 Comparación entre BTY- 7A006 Y bomba magnética 15R.	17
Tabla 7 Datos técnicos de Arduino Mega.....	23
Tabla 8 Especificaciones técnicas del Shield SIM900	24
Tabla 9 Especificaciones técnicas de la termocupla DS18B20	26
Tabla 10 Especificaciones técnicas del sensor ultrasónico HC-SR04	27
Tabla 11 Especificaciones técnicas del 4 módulo Relé 16 Canales 5V optoacoplador	28
Tabla 12 Especificaciones técnicas de la bomba BTY- 7A006	29
Tabla 13 Especificaciones técnicas de la Electroválvula HFS 12v DC	30
Tabla 14 Especificaciones técnicas del módulo convertidor DC-DC XI6009.....	31
Tabla 15 Tiempos de procesamiento de la Caldera 1	54
Tabla 16 Tiempos de procesamiento de la Caldera 2	55

Capítulo I. Introducción

La elaboración de cerveza artesanal ha sido uno de los emprendimientos de mayor acogida en los últimos años, debido a su alta demanda de consumo nacional e internacional, la cual se diferencian por sus características particulares como son el aroma, sabor y amargor.

En el caso del Ecuador, actualmente ha mostrado mayor crecimiento en la última década, de tal manera se han creado sociedades en distintas regiones del país para la creación de este producto.

Se han presentado distintos tipos de control y automatización enfocadas a las grandes productoras de cerveza artesanal, distanciándose considerablemente de los microempresarios. Por ende, se mantiene un enfoque en el sector PYMES. Para lograr optimizar los recursos en su fabricación, se ha visto la necesidad de tecnificar los procesos de elaboración, facilitando el monitoreo, registro y control en sus diferentes etapas de procesamiento, delimitando la complejidad en la toma de medida de muestras de los parámetros de temperatura con respecto al tiempo de la cocción en sus diferentes etapas, manteniendo un registro para su pertinente comprobación con sus respectivas recetas, además del control de las bombas succionadoras, el motor de accionamiento de mezclado y gestión de alarmas en sus respectivos tiempos de preparación.

El proyecto propone diferentes tipos de sensores que obtendrán medidas las cuales se podrán observar en la estación de trabajo, y se podrá procesar esta información y almacenarla o presentarla mediante una interfaz gráfica hacia un usuario final. Esta tesis propone además una gestión de alarmas mediante GSM, la cual mantendrán informado al coordinador a través de mensajes de texto.

1.1 Alcance

El alcance del presente trabajo de titulación es realizar el diseño e “implementación de un sistema en el cual contenga un monitoreo de la temperatura respecto del tiempo en las diferentes etapas de la maceración,

monitoreo del nivel de líquidos del recipiente con los respectivos sensores, el control ON/OFF de las válvulas, activación y bloqueo de las bombas de extracción para la recirculación del mosto”. Además de desarrollar un conjunto de alarmas las cuales serán notificadas mediante mensajes de texto directamente al operador y en su estación de trabajo.

Todo esto se implementará con la finalidad de obtener un sistema centralizado el cual registre los eventos suscitados durante el proceso de maceración para la fabricación de la cerveza artesanal, de esta manera se podrá tomar las acciones necesarias de manera oportuna, rápida y automatizada.

Se realizará un análisis de operatividad del sistema implementado con la finalidad de mitigar posibles errores o fallas que pudieren presentarse durante el proceso de maceración de cerveza artesanal.

1.2 Justificación

De acuerdo con el posicionamiento de la universidad como institución de calidad y alto nivel, capaces de dar soluciones prácticas, innovadoras y acordes al desarrollo tecnológico actual, permite la posibilidad de abrir nuevas relaciones con sectores productivos para la vinculación de estudiantes en el desarrollo de futuros proyectos.

En la actualidad debido a la creciente necesidad de optimizar tanto los procesos de elaboración como los productos resultantes, surge la necesidad de mantener un control automático en los mismos.

Actualmente en el mercado no se cuenta con un método que cumpla con dichas especificaciones necesarias para la producción de cerveza artesanal, es así que el sector micro empresarial, PYMES, requiere el desarrollo de nuevas técnicas para el monitoreo y control de procesos con la finalidad de llegar a una optimización de los recursos de fabricación y la mano de obra, las mismas que deben estar sustentadas con el uso de la tecnología de una manera amigable al usuario permitiendo fácil acceso para su operación.

Es así como, es sumamente necesario el desarrollo de un sistema integral para la producción de la cerveza, en el cual se pueda no solo visualizar los

parámetros involucrados en la elaboración, sino que también se pueda realizar las acciones necesarias y de manera inmediata, todo esto de forma automatizada, con la finalidad de aprovechar los recursos inmersos en el proceso obteniendo un producto de calidad.

1.3 Objetivos

1.3.1 Objetivo General

Implementar un prototipo de un sistema de control, monitoreo y gestión de alarmas para la elaboración de la cerveza artesanal.

1.3.2 Objetivos específicos

- Realizar el estudio del arte sobre el proceso de elaboración de cerveza artesanal.
- Diseñar un sistema electrónico para el monitoreo, control y gestión de alarmas.
- Implementar un sistema centralizado para el monitoreo, activación de los elementos de control del proceso y gestión de alarmas en tiempo real.
- Ejecución y pruebas de funcionamiento del sistema centralizado de monitoreo, control remoto y notificación de los eventos para la tecnificación del proceso de maceración de cerveza artesanal.

Capítulo II. Marco Teórico.

2.1 Estudio sobre el arte de elaboración.

2.1.1 Definición

La cerveza es una “bebida de moderado contenido alcohólico, resultante de un proceso de fermentación controlado, por medio de levadura cervecera proveniente de un cultivo puro, en un mosto elaborado con agua de características fisicoquímicas y bacteriológicas apropiadas, cebada malteada sola o mezclada con adjuntos, con adición de lúpulo y/o los derivados de lúpulo”. (INEN, 2003, pg. 1)

2.1.2 Historia.

De acuerdo con los historiadores creen que ya existía en Mesopotamia y Sumeria en el año 10.000 A.C, mientras se remojan en agua los cereales para la elaboración de pan. Los babilónicos procesaban 20 distintos estilos de cerveza. En Egipto se realizaban fermentaciones a gran escala de malta de cereales al año 3500 A.C. Mediante el proceso de la elaboración de la cerveza y obtuvo el primer descubrimiento bioquímico por parte de la humanidad. A esta cerveza se le denomina históricamente como Nekhen-Hoffman de tipo “ale”. Antes de la incorporación de lúpulo se vendían mezclas de hierbas llamadas gruit las cuales se vendían ya que tenían una “licencia de hierbas”.

Se tiene poco conocimiento acerca de la mezcla inicial llamada gruit, dado que existía una mezcla de composiciones heterogéneas, las cuales eran únicas que se caracterizaban y diferenciaban de las demás. Naturalmente se utilizaba diferentes tipos de hierbas como mirto, hidra entre otras.

Durante el siglo XV, ingresó el lúpulo como un nuevo ingrediente en la producto final, el cual brindaba cierto amargor a la cerveza, esto fue realizado por los comerciante de Holanda y Flandes.

En los años 1516 se introdujeron leyes a favor de la producción de dicho producto, las mismas que se denominaron; las leyes de pureza de la cerveza, en la cual detalla la utilización exclusiva de agua, lúpulo y malta, estos estatutos se dieron bajo la administración de Guillermo VI.

En Alemania en la última etapa del siglo XVII, se realiza la publicación de “Zymotechnica Fundamentalis”, que fue una obra en la que trata de establecer una teoría científica en la cual revele el proceso que conlleva la fermentación. Además, que Leeuwenhoek descubre por primera vez las células individuales de la levadura al mirarlas mediante el microscopio.

En los años 1750, mediante la revolución y migración a las ciudades, se obtuvo la elaboración y comercialización de cerveza a gran escala. La creación del termómetro, el hidrómetro y la máquina de vapor fueron de mayor aporte en el proceso de preparación.

En la década de los setenta fue el ocaso de la cerveza artesanal y predominó la cerveza industrial debido a su mayor comercialización. Dado que se presentó este inconveniente, Jimmy Carter legaliza el (homebrewing), la fabricación casera de cerveza ocho años después. En el siglo XIX, el investigador francés, Luis Pasteur descubre que las levaduras son las responsables del proceso de fermentación.

Mientras tanto los productores de cerveza necesitaban de la levadura natural y del aire para que se realice la fermentación.

En el año 1566 en el convento de San Francisco se creó la primera cerveza del continente en la ciudad de Quito, en la cervecería de San Francisco por el Fray Jodoco Rique (español). Las instalaciones que armó el sacerdote belga siguen en pie y fueron usadas hasta hace aproximadamente 50 años.

La Cerveza mantiene una larga historia, aún más con el arte de preparación y la ciencia que ha revelado los secretos de su elaboración, ya que no es un solo producto: son miles, ciento de miles a nivel mundial, los mismos que cuentan con algún tipo de singularidad que los caracteriza de las demás.

2.2 Componentes

2.2.1 Cebada / Malta

Es una planta gramínea anual, originaria de Asia occidental. Todos los cereales se dejan maltear, pero quizás el más idóneo sea la cebada, gracias a su cáscara pues facilita el proceso de maceración.

A continuación, se presenta la cebada de dos carreras en la figura 1.



Figura 1 Cebada de dos carreras

Adoptada de: (Huxley, 2011, pg. 100)

La figura 1 representa el tipo de cebada que es apropiada para la elaboración de cerveza artesanal, conocida como cebada de dos carreras debido a que genera sabores más delicados y de mayor deleite para el paladar.

2.2.2 Malteado

Es el proceso de germinación de la cebada pues genera las transformaciones que la planta conoce de forma natural durante su crecimiento deteniendo esta transformación rápidamente según las características esperadas para activar enzimas y empezar a producir azúcares fermentables.

Las enzimas del malteado liberan el ácido giberélico que es el responsable de activar las enzimas, que permiten cortar las moléculas de proteína y almidón para facilitar el proceso de metabolización.

Posteriormente se seca la malta a altas y bajas temperaturas con el fin de activar las enzimas. La mayor parte de las maltas especiales son tostadas directamente desde su estado verde.

2.2.3 Lúpulo

Es una planta perenne que se reproduce a partir de esquejes de una madre histórica, cruzada según criterios agronómicos y cerveceros, normalmente llega

a tener cinco metros de altura o más, siendo lo más utilizado las flores de la planta femenina, que a su vez se presenta en forma de pellet comprimido.

A continuación, se presenta el lúpulo en la figura 2



Figura 2 Pellet de Lúpulo

Adaptado de: (Cervecearte, s.f)

Los lúpulos contribuyen el amargor que compensa el dulzor de la malta, que posee propiedades antibacterianas, por lo que también son conservantes naturales.

Contribuyen a la formación y la retención de la espuma. Los polifenoles que contienen reaccionan con las proteínas indeseadas de la malta y las hacen insolubles, facilitando su separación del mosto. Su característica principal es proporcionar el sabor, aroma y amargor de la cerveza artesanal.

2.2.4 Levadura

Es un organismo eucariótico, unicelular, un tipo de levadura utilizado en la fabricación de pan, cerveza y vino. Hay muchas especies, y se las clasifica de acuerdo con las características de su forma celular, de reproducción, de su fisiología y de su hábitat.

Se caracteriza por metabolizar azúcares y producir alcohol, Cuando dejan de reproducirse por falta de oxígeno empiezan a metabolizar azúcares y a producir

alcohol etílico, dióxido de carbono (CO₂), calor, y otros subproductos, entre ellos ésteres, cetonas, fenoles y varios ácidos.

2.2.5 Agua

El agua comprende el 95 % de la cerveza, y los minerales disueltos en ella tienen mucha importancia en las reacciones necesarias durante la maceración, y algunos hacen importantes contribuciones al perfil sensorial de la cerveza. Los componentes del agua que más influyen en la cerveza son: dureza del agua, iones, pH, bicarbonatos, iones de sodio, iones de calcio, iones sulfato, iones Cl y cloro.

2.3 Tipos de cervezas

2.3.1 Ales

Dentro de este tipo de cerveza artesanal existen los estilos ingleses (Gales), las de origen escoses se caracterizan por tener menos lúpulo y más malta, sin olvidar que en Irlanda también se produce este tipo de cerveza con características similares a las inglesas.

2.3.2 Ordinary Bitter

Tienen aroma (lúpulo-medio), de color dorado pálido, con un sabor amargo medio alto, los lúpulos utilizados para la elaboración de este tipo de cerveza son ingleses, utilizando agua baja en carbonatos.

2.3.3 American Pale Ale

Tiene carácter propio, con aroma y sabor malta baja- media, de aspecto dorado y muy refrescante al paladar con cuerpo ligero medio, el agua posee un contenido bajo en carbonatos y rica en sulfatos.

2.3.4 Oatmeal Stout

Americana con aromas suaves y frecuentemente con sabor a avena, de color marrón oscuro negro pues posee un cuerpo más suave que los otros tipos de stouts.

2.4 Diferenciadores respecto a la cerveza Industrial.

La cerveza artesanal tiene varios factores diferenciadores en comparación con la cerveza industrial, tal es el caso que puede tener diversos estilos, sin agregar ningún tipo de conservantes, ni cereales adjuntos. Sin embargo, tiene un precio más elevado debido a que es un producto de calidad, artesano elaborado mediante un proceso riguroso con un contorno distintivo y personal. A comparación de las cervezas industriales, las que mantiene un proceso de producción más industrializado, asimismo la utilización de cereales adjuntos y aún más conservantes, por lo cual mantienen un estilo idéntico. Es decir, las grandes industrias cerveceras se interesan más en producir cantidad, descartando la calidad en su producto final. Y, por ende, la disminución de sus costos en la elaboración.

A continuación, se presenta las diferencias entre la cerveza artesanal e industrial en la tabla 1.

Tabla 1

Diferencias: cerveza artesanal e industrial

Cerveza Industrial	Cerveza Artesanal
Bajo costo	Costo elevado
Cereales adjuntos	No lleva adjuntos
Conservantes	No tiene conservantes
Estilos idénticos	Diversos estilos
Proceso industrial	Proceso artesanal

Adaptado de: (Cervesamontseny, s.f)

2.5 Aspectos técnicos

En esta parte se puntualizará el proceso de elaboración artesanal bajo ciertos parámetros necesarios en cada una de sus etapas, y bajo ciertos procedimientos para diferentes tipos de cerveza a procesar.

2.5.1 Tratamiento del agua

El agua potable puede tener una excelente condición para el consumo humano, pero no podría ser lo ideal para la fabricación de cerveza. Por tal razón se debe retirar el cloro del agua necesaria para la elaboración del producto, las cuales

se puede utilizar filtros de carbono activos, filtros de ozono, además de airear el agua y envasar en un recipiente descubierto durante un día.

2.5.2 Trituración de granos

El proceso trituration de granos consiste en romper el grano de cebada malteada en diferentes fragmentos y trozos, además de mantener a la cascara de grano lo más compacta posible. Es decir que se debe preservar ciertos porcentajes de las etapas de la desintegración. Los mismos que deben mantenerse en 30% de cascara, 10% de grano grueso, 30% grano fino y 30 % de harina. Para así obtener un buen mosto.

2.5.3 Cocción del mosto

En el proceso de cocción se debe agregar la malta molida con el agua a 70 °C, inmediatamente se debe mezclar para no generar grumos de la malta, durante el transcurso de 90 min se debe mantener a una temperatura de 65 a 66 °C, es de mayor importancia mantener un control de la temperatura con respecto al tiempo.

2.5.4 Recirculación y muestreo

El proceso consiste en extraer el líquido y mantener el mosto en una circulación contantes por un tiempo establecido, dado que es necesaria la extracción de los azúcares totales de la cebada malteada. La toma de muestras se lo realiza, por causas de obtención de la primera medida de densidad y de temperatura para obtener los grados alcohólicos requeridos en la receta.

2.5.5 Macerado

La maceración consiste en la segunda cocción del mosto además de la eliminación de la espuma durante el hervor y la agregación de los lúpulos para el sabor, aroma y amargor durante el tiempo y temperatura respectiva a la receta correspondiente al tipo de cerveza.

2.5.6 Enfriamiento

Se realiza el intercambio inmediatamente de temperatura de 90 °C a 20 °C, debido a que, si el mosto reposa mucho tiempo entre 30 y 50 °C, habrá mayor posibilidad de irrupción por organismos no invitados tal como son las bacterias

impuestas en el aire y además que la levadura se activa homogéneamente con el mosto a temperatura ambiente.

2.5.7 Fermentación

La actividad biológica se presentará desde el primer momento a través del acondicionamiento del mosto con la levadura. El tiempo de duración de la fase de adaptación depende de las enzimas, aún más se mostrarán influenciadas mediante los nutrientes y la temperatura disponibles. La cual se debe mantener a una temperatura ambiente.

Después de la primera fermentación se requiere realizar la filtración de la levadura y retirar las enzimas muertas. Y así mantener la última fermentación sin encimas ni levaduras antes del proceso de embotellamiento.

A continuación, se presenta un diagrama de bloques del proceso general en la figura 3.

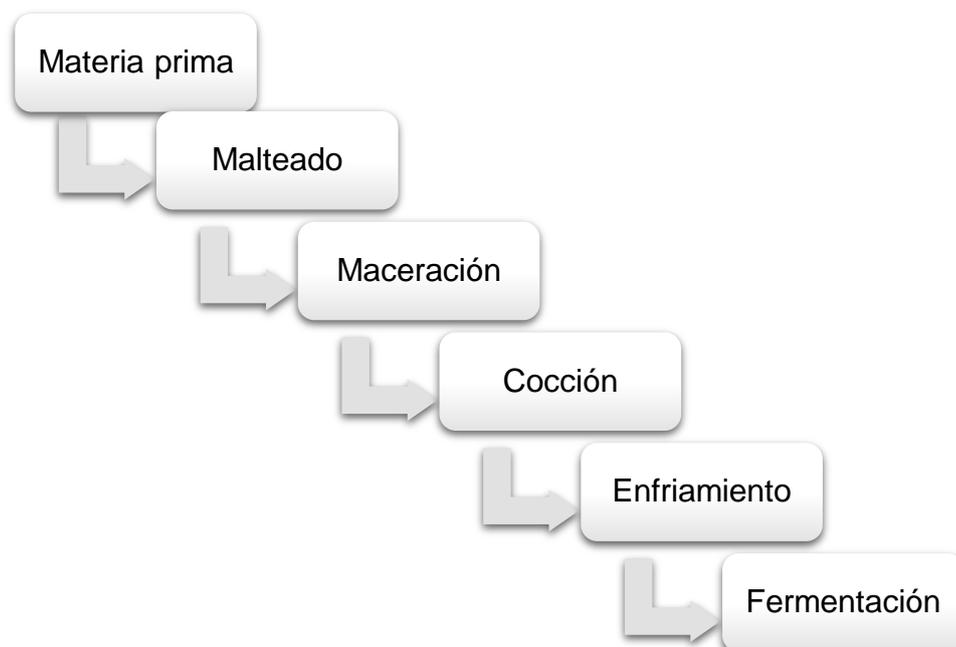


Figura 3 Diagrama de procesos general

Capítulo III. Diseño e implementación

En el presente capítulo se detalla el diseño del sistema de control, monitoreo y gestión de alarmas para la elaboración de la cerveza artesanal. Además, la descripción del hardware utilizado, así como la programación de los sensores, la interfaz gráfica y la adquisición y registro de la información en los sensores.

En el diseño del sistema de control, monitoreo y gestión de alarmas se utilizaron componentes de hardware como software, se decidió implementar termocuplas y sensor de ultrasonido programado para utilizar como un sensor de nivel, electroválvulas, motor, bombas de extracción, que estarán conectados a un microcontrolador que accionara las funciones o activación de diferentes dispositivos, además de la conexión serial hacia un computador donde mantendrán el registro de datos. Por el cual se podrá realizar un monitoreo atreves de un interfaz gráfico.

A continuación, se presenta el diagrama de bloques, de la interconexión de los diferentes componentes en la figura 4.

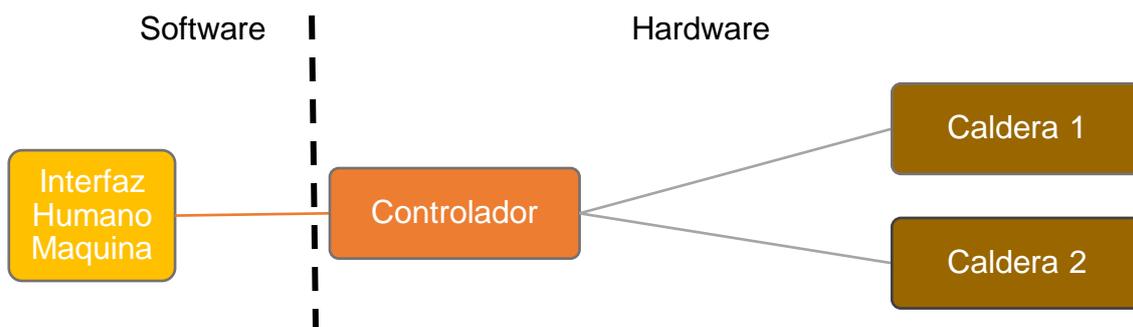


Figura 4 Diagrama de interconexión

Una vez que se tiene claro la interconexión de los elementos del prototipo, es preciso dividir el diseño en segmentos de tal manera que sea fácil definir su funcionamiento y ejecutar la implementación del prototipo. Para este proyecto se ha dividido en cinco bloques el desarrollo del diseño.

A continuación, se presenta el diagrama de bloques con sus diferentes componentes en la figura 5.

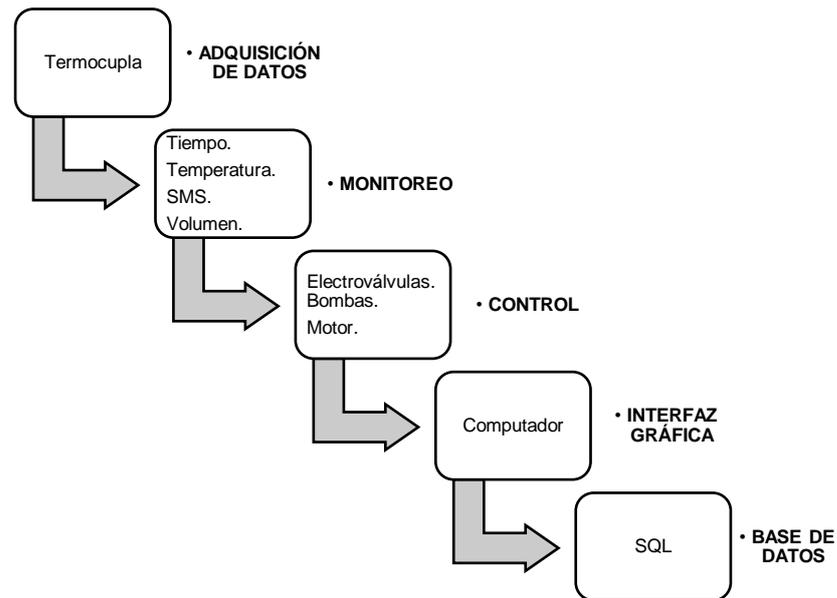


Figura 5 Diagrama en bloques

3.1 Adquisición de datos

Es necesario realizar una obtención de datos físicos del líquido en diferentes etapas, lo cual se obtiene mediante una termocupla programada por el Arduino.

3.1.1 Monitoreo

Es importante obtener parámetros de temperaturas en las diferentes etapas con respecto al tiempo, de igual manera de la medida del volumen final debido a que por razones de pérdida ocasionada por la evaporación del líquido a diferentes temperaturas. A demás de la notificación de estos eventos por medio de mensajes escritos (SMS).

3.1.2 Control

Este proceso se refiere al accionamiento de los dispositivos utilizados como el agitador de la malta, activación automática de la electroválvula que permitirá la circulación del mosto, así mismo de las bombas para la recirculación y transferencia de una caldera a otra. Cada etapa mantiene una secuencia la cual es controlada por tiempos correspondientes a la receta establecida.

3.1.3 Interfaz Gráfica

Para una fácil interpretación de los datos recibidos, se necesita desarrollar e implementar una interfaz gráfica en un computador con sistema operativo

Microsoft Windows 7, 8 o 10, dicha interfaz debe ser amigable con el usuario, en este proyecto se utilizará el software Microsoft Visual Studio.

3.1.4 Base de datos

Para almacenar la información recibida en el computador, es preciso implementar una base de datos de modo que posteriormente se pueda acceder a la información de los sensores y utilizarla para los fines que se deseen, tales como llevar un registro de eventos, e incluso el procesamiento de estos datos para la obtención de una función de transferencia de la planta lo cual no serviría para la automatización total de la misma.

3.2 Análisis de componentes electrónicos.

El análisis correspondiente sobre las necesidades y requerimientos para el diseño del prototipo de un sistema de control, monitoreo y gestión de alarmas para la elaboración de la cerveza artesanal. Para lo cual se realiza un cuadro comparativo de cada dispositivos a utilizar en el proyecto.

3.2.1 Microcontrolador

En el presente proyecto se consideró la posibilidad de utilizar el LOGO 8 Basic como sustituto del Arduino Mega para este proyecto, a continuación, una comparación entre los dos componentes en la tabla 2.

Tabla 2

Comparación entre Arduino Mega y Logo 8

Medidas	Arduino MEGA	LOGO 8 12/24 RCE
Dimensión	5,33 cm x 10,15 cm	6 cm x 9 cm
Memoria	0.002 MB	400 bloques
Velocidad de reloj	16 MHz	40
Red	No	si
Memoria Flash	256 KB	250
OS	No	LOGO Soft Comfort.

Arduino MEGA es un controlador robusto que cuenta con un lenguaje de programación de software libre de fácil acceso, otro de los factores más trascendentales de su bajo costo con respecto a otros tipos de controladores como son PLCs.

3.2.2 Modulo Gsm.

En el presente proyecto se analizó la posibilidad de implementar el SIM800L como sustituto del SIM900, para el mencionado proyecto, a continuación, una comparación entre ambos componentes en la tabla 3.

Tabla 3
Comparación entre SIM900 y SIM800L

Medidas	SIM900	SIM800L
Voltaje de funcionamiento	5V- 20	3.4V - 4.4V DC
Codificación	UART	UART
Frecuencia de funcionamiento	850/900/1800/1900 MHz	850/900/1800/1900 MHz
TCP/UDP	Si	No
Consumo de energía.	1.5 mA	500 mA

El SIM800L es un módulo Gsm, que cuenta con las características requeridas y adaptables al diseño del prototipo, debido a las especificaciones de voltaje funcionamiento como la ganancia de la antena de dicho dispositivo.

3.2.3 Termocupla

En el proyecto de titulación se consideró al sensor de temperatura driver K-0135 en remplazo de la termocupla DS18B20, para dicho proyecto a continuación, una comparación entre ambos componentes en la tabla 4.

Tabla 4
Comparación entre Driver max6675 y DS18B20

Medidas	DS18B20	Driver max6675
Voltaje de funcionamiento	3-5 V DC	3.3 a 5 V DC
Exactitud	$\pm 5^{\circ}\text{C}$	N/A
ID	Único 64 bits	12 bits
Rango de Temperatura	-55 a 125°C	0 ~ 800°C
Tiempo de captura	inferior a 750ms	0, 17
Interfaz	1 cable	2 cable
Longitud del cable	1 m (sumergible)	1m (No sumergible)

La termocupla DS18B20 cumple con las características requeridas para el proyecto, como es su exactitud de medida y la protección del cable que es sumergible y resistente a altas temperaturas. A demás de su mejor integración con Arduino.

3.2.4 Sensor de nivel

En el presente proyecto se realizó el análisis de la posibilidad de utilizar el sensor ultrasónico K-0135 como remplazo de HC-SR04 para este proyecto, a continuación, una comparación entre los dos componentes en la tabla 5.

Tabla 5
Comparación entre HC-SR04 Y K-0135

Medidas	HC-SR04	K-0135
DC de funcionamiento	5 V	5 V
Tamaño del circuito	43 x 20 x 17 mm	65 x 20 x 8 mm
Medida máxima	4.5 m	40 mm

Media mínima	1.7 cm	16 mm
Tiempo del pulso de disparo	10 μ S	no
Tiempo del pulso de la señal out	100-25000 μ S	no
Tiempo de espera mínimo entre una medidas	20 mS.	no

El sensor de nivel HC-SR04 mantiene características necesarias para el proyecto, como son los rangos de medidas y temperaturas de funcionalidad.

3.2.5 Bomba

En el proyecto de titulación se consideró la posibilidad de implementar la bomba magnética 15R como sustitución del BTY- 7A006, a continuación, una comparación entre los dos componentes en la tabla 6.

Tabla 6

Comparación entre BTY- 7A006 Y bomba magnética 15R.

Medidas	BTY- 7A006	Bomba magnética 15R
Voltaje de funcionamiento	12 VDC	110 V
Amperaje	0.7 A	N/A
Máxima altura de agua	3 metros	N/A
Velocidad máxima de alimentación	8 L / min (2.1GPM)	8 L / min
Temperatura máxima del agua circulante	100 ° C / 212 ° F	100 °C

La bomba de succión BTY- 7A006 mantiene las características necesaria para el prototipo, como es su voltaje de funcionamiento y su máxima altura.

3.2.6 Electroválvula

Electroválvula HFS 12v DC mantiene su característica de funcionalidad mediante voltaje DC, necesaria para el diseño.

3.3 Diseño de hardware

En el diseño del hardware para el prototipo se utilizaron varios elementos que se detalla a continuación:

Caldera 1:

- Termocupla
- Filtro.
- Electroválvula.
- Bomba.
- Agitador.

Caldera 2:

- Termocupla
- Filtro.
- Electroválvula.
- Sensor Ultrasónico.

Controlador:

- Microcontrolador
- GSM

A continuación, se presenta el diagrama de componentes de hardware en la figura 6

HARDWARE

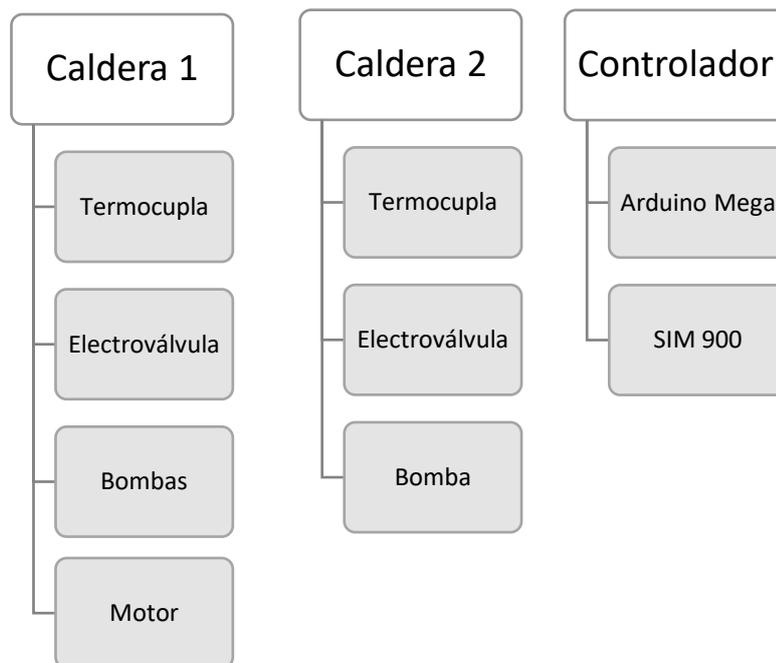


Figura 6 Diagrama de los componentes de hardware

A continuación, se presenta el diagrama de equipos utilizados además de. su caldera correspondiente, en la figura 7.

CALDERAS	DISPOSITIVOS	CONTROLADOR	INTERFAZ GRAFICA	BASE DE DATOS
<p>CALDERA 1</p> 	 			
<p>CALDERA 2</p> 				

Figura 7 Diagramas de equipos

De acuerdo con lo detallado, es trascendental describir cada uno de los componentes de manera que se pueda comprender el funcionamiento.

3.3.1 Dimensionamiento de calderas.

El dimensionamiento de los recipientes es necesario para la elaboración de este proyecto de tal manera que se representa en la forma de un cilindro y con un material óptimo para el procesamiento de productos alimenticios; como es el caso del acero inoxidable que mantienen una distribución homogénea del fuego, para el cálculo del volumen del cilindro se considera un volumen mínimo de 50 litros para la ejecución del proyecto, pero debido al alto costo de adquisición y la limitación de fabricación en esta capacidad, se establece un volumen máximo de 120 litros, proyectando así el crecimiento del negocio con una sola inversión.

Volumen del recipiente se obtiene de la siguiente fórmula:

$$V = \pi \times r^2 \times h \quad (\text{Ecuación 1 Volumen})$$

Para obtener la altura se ingresa las variables conocidas y se despeja la altura.

$$v = 150L, r = 25 \text{ cm.}$$

Adaptada de: (leon, Nuevo, s.f)

$$h = \frac{V}{\pi \times r^2}$$

$$h = \frac{120000 \text{ cm}^3}{3,14159 \times 625}$$

$$h = 61.1 \text{ cm}$$

De tal manera las calderas cuentan con una altura de 61.1 cm y un diámetro de 50 cm cada una.

A continuación, se visualiza la imagen del dimensionamiento de los calderos en la figura 8.

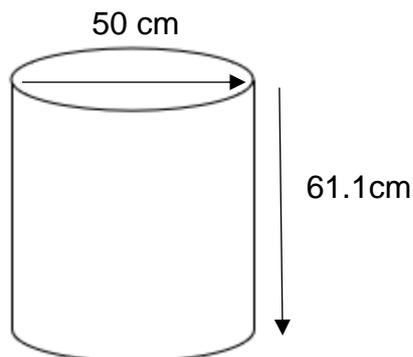


Figura 8 Dimensionamiento de los calderos

De acuerdo con el dimensionamiento de los recipientes se considera la implementación de una cocina industrial que soporte el peso del recipiente más el líquido y la materia prima para la elaboración de cerveza artesanal, de tal manera se implementa hierro en toda la estructura del soporte con sus respectivas hornillas.

A continuación, se muestra la imagen de la cocina industrial para la elaboración del producto en la figura 9.



Figura 9 Cocina Industrial

3.3.2 Arduino Mega 2650

El Arduino Mega es controlador electrónico que está basada en el ATmega2560, se caracteriza principalmente porque tiene 54 pines de entrada / salida digital, 16 entradas análogas, conexión USB y es muy sencillo de usar.

En el presente proyecto, Arduino Mega cumple con las características antes señaladas.

Las especificaciones técnicas del Arduino MEGA se describen en la siguiente tabla 7.

Tabla 7
Datos técnicos de Arduino Mega

Parámetros	Valores
Voltaje de funcionamiento	5V
Voltaje de entrada estándar	7-12V
Voltaje de entrada (máximo)	6-20V
IN/OUT digitales	54 (de los cuales 15 proporcionan salida PWM)
IN/ OUT análogas	16
Amperaje IN/OUT	20 mA
Amperaje	50 mA
Memoria Flash	256 KB de los cuales 8 KB se usan por el boot loader
Memoria SRAM	8 KB
Memoria EEPROM	4 KB
Velocidad del reloj	16 MHz
Dimensión altura	101.52 mm
Dimensión Grosor	53.3 mm
Peso	37 g

A continuación, se exhibe a la tarjeta Arduino Mega en la figura 10.



Figura 10 Arduino Mega

Adatado de: (Arduino, s.f)

3.3.3 GPRS/GSM Shield SIM900

La tarjeta ultra compacta utiliza el sistema global de comunicaciones móviles; el cual es un estándar de comunicación libre para la telefonía móvil, cuyo funcionamiento se realiza mediante celdas. Generalmente formado por dos partes: una red de comunicaciones o estación base y estaciones móviles o terminales que permiten el acceso a la red.

Este dispositivo es compatible con todos los modelos de Arduino, además puede también controlar otros microcontroladores. El módulo es configurado y controlado mediante UART, utilizando comandos AT. Se realiza la conexión directa al microcontrolador, Arduino, e inicia la comunicación a través de comandos AT. Esta tarjeta es excelente para sistemas remotos, comunicación recursiva, puntos de control y envío de mensajes de texto.

Las especificaciones técnicas se describen en la tabla 8:

Tabla 8
Especificaciones técnicas del Shield SIM900

Parámetros	Valores
Voltaje de funcionamiento	5V-20
Codificación	UART
Frecuencia de funcionamiento.	850/900/1800/1900 MHz

M-Gprs	Slot 10/8
E.M.GPRS	Clase B
Clase 1/ 4	1 W a 1800 / 1900MHz - 2 W a 850 / 900 MHz
TCP/UDP	Acceso a un servidor web.
Consumo de energía.	1.5 mA
Temperatura de funcionamiento	-40°C A +85 °C

Adpatada de: (HetPro, 2014)

A continuación, se presenta la tarjeta electronica de GPRS/GSM Shield SIM900 en la figura 11.



Figura 11 GPRS/GSM Shield SIM900

Adaptada de: (Banshee, s.f)

3.3.4 Termocupla DS18B20

Es un sensor digital de temperatura impermeable por lo cual puede ser sumergido en líquido. Maneja el protocolo 1-Wire para establecer comunicación, este protocolo necesita solo de un pin de datos para comunicarse y admite conectar varios sensores en el mismo bus de datos, su encapsulado de fábrica es de tipo TO-92 similar a lo utilizado en transistores pequeños.

Las especificaciones técnicas de la termocupla DS18B20 se describen en la tabla 9:

Tabla 9
Especificaciones técnicas de la termocupla DS18B20

Parámetros	Valores
Voltaje de operación	3-5 V
Exactitud	$\pm 5\text{ }^{\circ}\text{C}$
ID	Único 64 bits
Rango de Temperatura	-55 a 125°C
Resolución:	9 a 12 bits (configurable)
Tiempo de captura	inferior a 750ms
Interfaz	1 cable (impermeable)

A continuación, se muestra la termocupla se muestra figura 12



Figura 12 Termocupla DS18B20

Adaptada de: (Bricogeek, s.f)

3.3.5 Sensor ultrasónico HC-SR04

Este dispositivo es un módulo que puede medir diferentes distancias, a través de una señal de ultrasonido. Su funcionamiento se base en el envío de un pulso de alta frecuencia (no audible). Este pulso rebota en los objetos cercanos y es reflejado hacia el sensor, que dispone de un micrófono adecuado para esa frecuencia.

Las especificaciones técnicas del sensor ultrasónico HC-SR04 se describen en la tabla 10:

Tabla 10
Especificaciones técnicas del sensor ultrasónico HC-SR04

Parámetros	Valores
DC de funcionamiento	5 V
Tamaño del circuito	43 x 20 x 17 mm
Frecuencia de funcionamiento	40 KHz
Medida máxima	4.5 m
Media mínima	1.7 cm
Tiempo del pulso de disparo	10 μ S
Tiempo del pulso de la señal out	100-25000 μ S
Tiempo de espera mínimo entre una medidas	20 mS.

A continuación, se muestra el sensor ultrasónico HC-SR04 en la figura 13.



Figura 13 Sensor ultrasónico HC-SR04

Adaptada de: (Diosdado, s.f)

3.3.6 Módulo Relé 16 Canales 5V Optoacoplador

Es un módulo compuesto por 16 relés con tensión de accionamiento 5VDC, puede controlar hasta 16 dispositivos, como son lámparas, electroválvulas, bombas, ventilador entre otros, es ideal para usarla en domótica controlados por Arduino, pic, Raspberry. Las entradas de control se encuentran aisladas

con optoacopladores para disminuir el ruido generados por el circuito de control.

Las especificaciones técnicas del 4.2.5 Módulo Relé 16 Canales 5V optoacoplador se describen en la tabla 11:

Tabla 11

Especificaciones técnicas del 4 módulo Relé 16 Canales 5V optoacoplador

Parámetros	Valores
Voltaje de funcionamiento.	5 V
Tamaño del circuito.	90 x 17 x 180 mm
Amperaje de funcionamiento.	15 a 20 mA
Capacidad de Relé.	250VAC/ 10A
Tiempo de repuesta	5-10 ms
Terminales de conexión	Común, NA y NC

A continuación, se presenta el dispositivo módulo Relé 16 Canales 5V Optoacoplador en la figura 14.

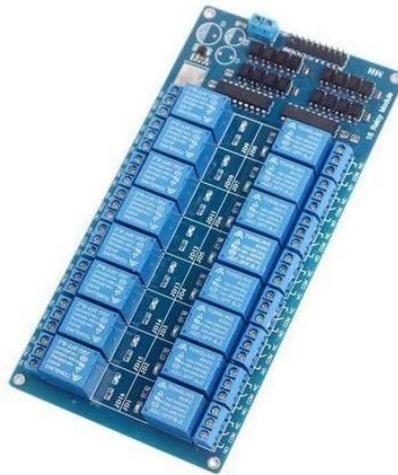


Figura 14 Módulo Relé 16 Canales 5V Optoacoplador

Adatado de: (Geekfactory, s.f)

3.3.7 Bomba BYT-7A006

Dispositivo encargado de extraer el líquido de un diferente caudal de distinta altura, presión. Es utilizado con el fin de reducir el desperdicio de agua y de aumentar el confort en las aplicaciones implementadas.

Las especificaciones técnicas de la bomba BYT- 7A006 se describen en la tabla 12:

Tabla 12

Especificaciones técnicas de la bomba BYT- 7A006

Parámetros	Valores
Voltaje de funcionamiento.	12 V
Amperaje	0.7 A
Velocidad máxima de alimentación	8 L / min (2.1GPM)
Máxima altura de agua	3 metros
Presión máxima del sistema	10 bar
Temperatura máxima del agua circulante	100 ° C / 212 ° F
Bajo nivel de ruido	30 dB por 1 metro

A continuación, se muestra la bomba BYT-7A006 en la figura 15

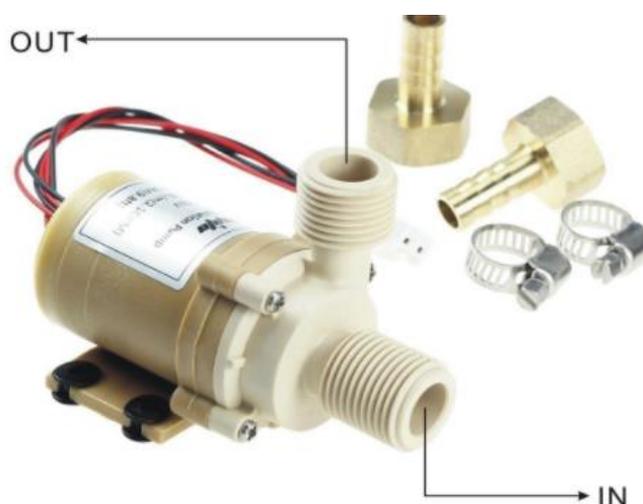


Figura 15 Bomba BYT-7A006

Adaptado de: (Ebay, 2018)

3.3.8 Electroválvula HFS 12V DC

Esta Electroválvula de 12 VDC reemplaza fácilmente las válvulas manuales, atizadas con agua, aire y otros fluidos de baja viscosidad, sus componentes de fabricación son de alta calidad como es latón y un solenoide de cobre, están diseñados para su máxima duración. Esta válvula es normalmente cerrada, dado que su activación se da cuando se lo energiza y está cerrada cuando no se suministra energía.

Las especificaciones técnicas de la Electroválvula HFS 12v DC se describen en la siguiente la tabla 13.

Tabla 13

Especificaciones técnicas de la Electroválvula HFS 12v DC

Parámetros	Valores
Voltaje de funcionamiento.	12 VDC
Exactitud	± 5 °C
Rango de temperatura	23F a 176F
Conectores	-1/2 "
Estado	Normalmente cerrada

A continuación, se muestra la Electroválvula HFS 12v DC en la figura 16.



Figura 16 Electroválvula HFS 12v DC

3.3.9 Modulo convertidor DC-DC XI6009

Este dispositivo estabiliza el voltaje de salida a través de un potenciómetro, el cual puede tener a su salida hasta 35 V DC, es utilizado en aplicaciones que requieren niveles de voltajes estables.

Las especificaciones técnicas del convertidor Modulo DC-DC XI6009 se describen en la siguiente la tabla 14.

Tabla 14

Especificaciones técnicas del módulo convertidor DC-DC XI6009

Parámetros	Valores
Voltaje de funcionamiento IN	3.2 a 32 VDC
Voltaje de funcionamiento OUT	5-35 V
Temperatura de funcionamiento	-45 a 85 °C
Amperaje	4 A.
Dimensiones	43x 21x 14

A continuación, presenta al módulo convertidor DC-DC XI6009 en la figura 17.



Figura 17 Módulo convertidor DC-DC XL6009

Adaptada de: (Naylampmechatronics, sf)

3.3.10 Software Arduino IDE

Este software cuenta con un ambiente integrado para el desarrollo, además de un editor de texto que se utiliza para ingresar el código con el cual se programará el Arduino MEGA, el programa contiene un área en el cual se despliegan mensajes de error o advertencia, una consola de texto y una barra de herramientas tal como se lo indica en la figura 18.

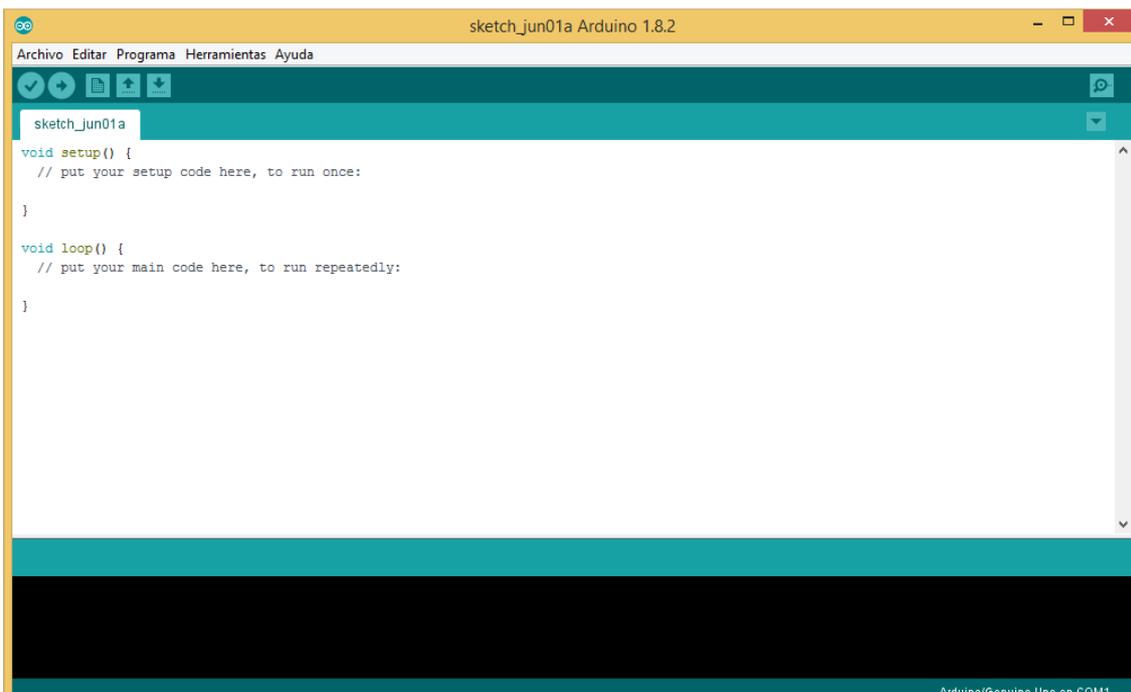


Figura 18 Interfaz principal Arduino IDE

El primer paso que se realiza para la inicialización de la programación con el Arduino MEGA es configura la comunicación entre la placa de Arduino y el software, para lo cual en el menú de herramientas se selecciona el ítem del puerto y se selecciona el puerto serial a utilizar en el Arduino.

Lo segundo que se realiza es seleccionar el modelo de placa de Arduino. En el menú de herramientas se escoge el ítem tarjeta y se selecciona Arduino Mega 2560 o Mega ADK , debido a que en este proyecto se usarán dispositivos Arduino Mega 2560.

Una vez realizado los pasos descritos, se iniciará con el desarrollo de la programación, de tal manera que se debe tener en cuenta la estructura del lenguaje a programar, de igual manera de las dos partes fundamentales para la inicialización y que se encierran en bloques que contienen las instrucciones o declaraciones.

- Función setup ()
- Función loop ()

La función setup () se la ingresa por única vez al inicializar el entorno de la programación, la misma que se utiliza para la inicialización de los pines de entrada o salida para la comunicación requerida, como también el ingreso de variables a utilizar.

La Función loop cuenta con el código ingresado como son lectura de entradas, activación de salidas, etc., que se ejecuta continuamente. Siendo así la función núcleo de todos los programas de Arduino, la cual realiza la mayor parte del trabajo

Adicionalmente a lo mencionado y dependiendo del tipo de proyecto a realizar, es necesario e importante incluir las librerías, las cuales son fragmentos de código que fueron realizados por terceros para el uso libre en otros proyectos, la utilización de las librerías proporciona la elaboración de diferentes programas, ya que se las puede incorporar sin inconveniente alguno y con la

ventaja de recorte tiempo en su realización ya que no es necesario elaborar uno nuevo. Adoptada de: (Arduino, s.f)

3.4 Sistema de control

Para realizar el prototipo de un sistema de control, monitoreo y gestión de alarmas para la elaboración de la cerveza artesanal es necesario cumplir con los siguientes requerimientos:

- Optimizar recurso en la elaboración
- Las variables inmersas en el proceso (Temperatura y tiempo), las cuales son las más importantes y más trascendentales en la elaboración.
- Se debe realizar eficientemente las acciones de control programadas.
- Debe ser seguro y amigable con el usuario operador.

Ademas se detalla el graficet del funcionamiento del sistema, en el cual se describe el proceso de control, monitoreo y gestión de alarmas.

Las variables utilizadas en el diagrama son:

- PC ON= Computador encendido
- PC OFF= Computador apagado
- ON= Encendido
- OFF= Apagado
- SW 1= Programa iniciado
- ES 1= Sistema energizado
- IDR 1= Datos Receta ingresados.
- T 1= Primera temperatura.
- T 2= Segunda temperatura.

- SMS X ON= Numero de mensaje iniciado.
- P1= 1 Pulsador de inicio de mezclado
- P2= 1 Pulsador de inicio de proceso de vaciado
- $t=Xmin$ Cumplimiento de x tiempo en minutos.
- $tx=ON$ Tiempo cumplido

A continuación, se presenta el diagrama de flujo del sistema de control diseñado en la figura 19.



Figura 19 Diagrama de flujo de control.

3.5 Proceso de implementación.

El proceso de elaboración del mencionado proyecto cuenta con un proceso de implementación, dicho de otra manera, el proceso es secuencial de tal manera no se puede saltar a la siguiente fase si no se realiza la anterior etapa, estas condiciones están establecidas en la programación del microcontrolador el cual se encarga de hacer cumplir el estado de cada una de sus etapas del proceso de maceración y hervido.

El proceso de maceración cuenta con la fase de mezclado la misma que se acciona de forma manual a través del panel de control, luego de esto se mantiene en constante monitoreo de la temperatura en relación con el tiempo para una correcta cocción y extracción de azúcares de la malta. Simultáneamente se efectúa de forma espontánea la etapa de recirculación mediante la activación de la electroválvula y la bomba uno, la misma que posee notificaciones mediante mensajes escritos del estado en que se encuentre (ON/OFF). Para la culminación de la maceración se activa manualmente en el panel de control la fase de cambio de caldera, y de igual manera se acciona los dispositivos antes mencionados. Todo esto se efectúa en la caldera 1.

Continuando con la elaboración de la cerveza artesanal en el proceso de hervido, el mismo que posee diferentes etapas, la primera se establece con la temperatura y las cuatro siguientes con respecto a temperatura y tiempo de hervido, luego de que estas fases se cumplen de manera sincronizada la etapa de enfriamiento se acciona el paso del líquido mediante la electroválvula y la succión del líquido. Este procedimiento se lo plasma en la caldera 2.

Para explicar mejor el proceso de la elaboración de la cerveza artesanal en cada una de sus fases se presenta en la figura 20 un diagrama de flujo.

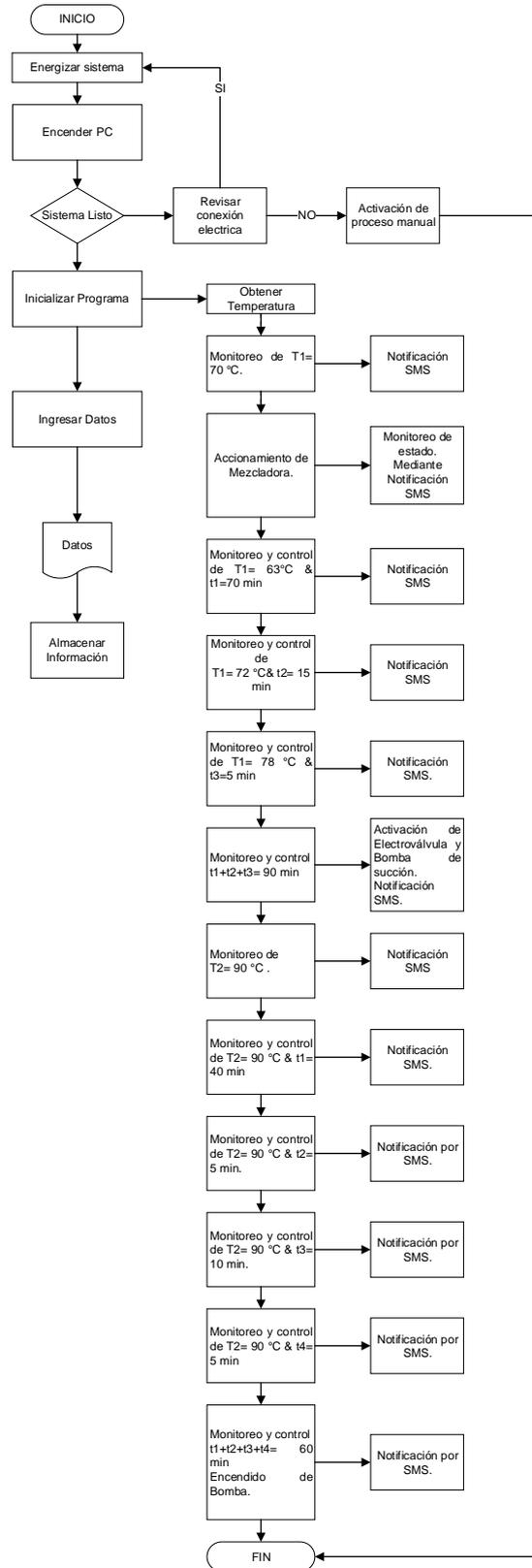


Figura 20 Diagrama de funcionamiento

3.6 Diagrama de conexión

El presente proyecto de titulación cuenta con un diagrama de conexiones en el cual se muestra el acoplamiento de las diferentes interfaces de los dispositivos, tales como: el regulador de voltaje DC que estabiliza el voltaje necesario para la inicialización del microcontrolador el cual controla a los distintos dispositivos, como las dos termocuplas, las mismas que obtienen la lectura constante de la temperatura de la primera caldera así como de la segunda, del mismo modo el accionamiento se realiza al cumplir condiciones con respecto al tiempo y de manera secuencial dentro del proceso de la elaboración del producto, es así se ha implementado luces piloto las cuales sirven de guía para conocer las etapas trascendentales del proceso

A continuación, se presenta la figura 21 correspondiente al diagrama de conexiones .

3.7 Diseño de Software

El software de este proyecto contiene los siguientes componentes:

- Programa
- Base de datos

El programa de monitoreo y registro se desarrolló en Visual Studio puesto que a su entorno de programación y la formas visuales que puede tomar e ideal para el presente proyecto

Mientras que la base de datos se lo implementa en MySQL dado que, a su distribución libre, con mejor rendimiento al realizar operaciones, soporta gran variedad de sistemas operativos, además de la conectividad, velocidad, y seguridad, hacen que MySQL Server sea altamente apropiado para acceder a las bases de datos a través de la red. (Mysql, 2009)

Para explicar el funcionamiento de la estación de trabajo se presenta los detalles del programa en la figura 22.

Software

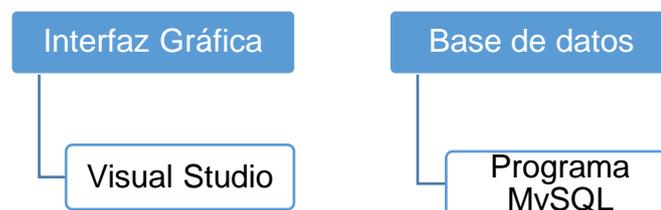


Figura 22 Detalles del programa

3.7.1 Software Microsoft Visual Studio 2017

El contorno de desarrollo de Visual Studio es un entorno de inicio creativo que se utiliza para mostrar y editar prácticamente cualquier tipo de código para

después, depurar, generar y publicar aplicaciones para Android, iOS, Windows, la Web y la nube. La misma que esta disponibles para sistemas operativos Mac y Windows. Esta plataforma mantiene una flexibilidad puesto que puede compilar cualquier tipo de aplicación, además de mejorar la productividad del equipo mediante la colaboración en la planificación de proyectos eficaces. (Microsoft, 2018)

Los procedimientos que se realiza en contorno general en la creación de un programa son:

- Crear nuevo proyecto
- Crear un interfaz de usuario
- Especificar el funcionamiento y el comportamiento
- Digitar el código de programación
- Ejecutar el programa

3.7.2 Diseño del programa con interfaz gráfica.

El lenguaje de programación implementada para el desarrollo de la interfaz gráfica de dicho proyecto es visual studio se eligió por razones de conocimientos adquiridos en el transcurso de la carrera y de forma profesional. El programa está diseñado en distintas pestañas que mostrará informaciones necesarias en para cada proceso de elaboración. Inicialmente se debe seleccionar el puerto en el cual está conectado el controlador y a continuación se debe conectar con el puerto al mismo tiempo para inicializar y recibir la trama de datos. De acuerdo con las necesidades; el programa cuenta con las siguientes pestañas

En la pestaña de INGRESAR RECETA se registra la materia prima a utilizar en la receta, las cuales que se deben seleccionar los tipos de maltas, cereal, endulzante y los diferentes lúpulos. Los mismos que se registraran en la base de datos al presionar GUARDAR, de igual manera se eliminan los datos ingresado al seleccionar LIMPIAR, en la figura 23 se indica estas opciones,

Figura 23 Interfaz de ingresar Receta

En la siguiente pestaña de CALDERA 1, se registra las tapas de elaboración, las mismas que se ingresan al seleccionar un valor establecido. Además, se visualiza constantemente la temperatura durante los 90 minutos cruciales, correspondiente al proceso de maceración, por otra parte, se realiza la captura de los datos mediante un intervalo de 8 segundos y su registro correspondiente en la base de datos, tal como se muestra en la figura 24.

Figura 24 Interfaz de registro y monitoreo de la Temperatura de la Caldera 1

A continuación, en la pestaña de CALDERA 2 se registra las acciones a realizar en el proceso, en las cuales se ingresan la confirmación del evento realizado. Además, se muestra el valor de la temperatura durante 60 minutos tiempo que corresponde al proceso de hervido. Por otra parte, se muestra la cantidad obtenida en litros después de las pérdidas registradas durante el proceso. De igual manera se configura un tiempo de 8 segundos para la adquisición de datos, con motivo de registrar en la base de datos, como se puede observar. En la figura 25 se indica el esquema de la pestaña denominada caldera 2.



Figura 25 Interfaz de registro y monitoreo de la Temperatura de la Caldera 2

En la siguiente pestaña de FERMENTACIÓN se integra el valor de la temperatura final, luego de realizar el intercambio de temperatura. También se registra los eventos de la primera y segunda semana de Efervescencia, las cuales se ingresa la confirmación de la etapa realizada y el ingreso de la cantidad de endulzante requerido.

El proceso de fermentación no se realizó la automatizo, debido a su alto costo en los equipo requeridos, de tal manera se lo realiza de forma manual, tal como se muestra en la figura 26.



Figura 26 Interfaz de registro de fermentación.

En la pestaña de CONSULTA RECETA sirve para consultar las recetas ingresadas y las realizadas con su respectivo nombre del tipo de cerveza e ingredientes utilizados, pero su distintivo será la fecha de elaboración, tal como se puede ver en la figura 27.



Figura 27 Interfaz de gráfica de consulta de receta.

En las pestañas de CONSULTA DE CALDERA 1 y 2 se muestra el parámetro almacenado en la base de datos, para observar la variación de temperatura en un gráfico de infusión simple que hace referencia a la temperatura con respecto

al tiempo, muy útil para la creación de nuevas recetas. A demás de sus eventos realizados. Un ejemplo se lo indica en la figura 28.

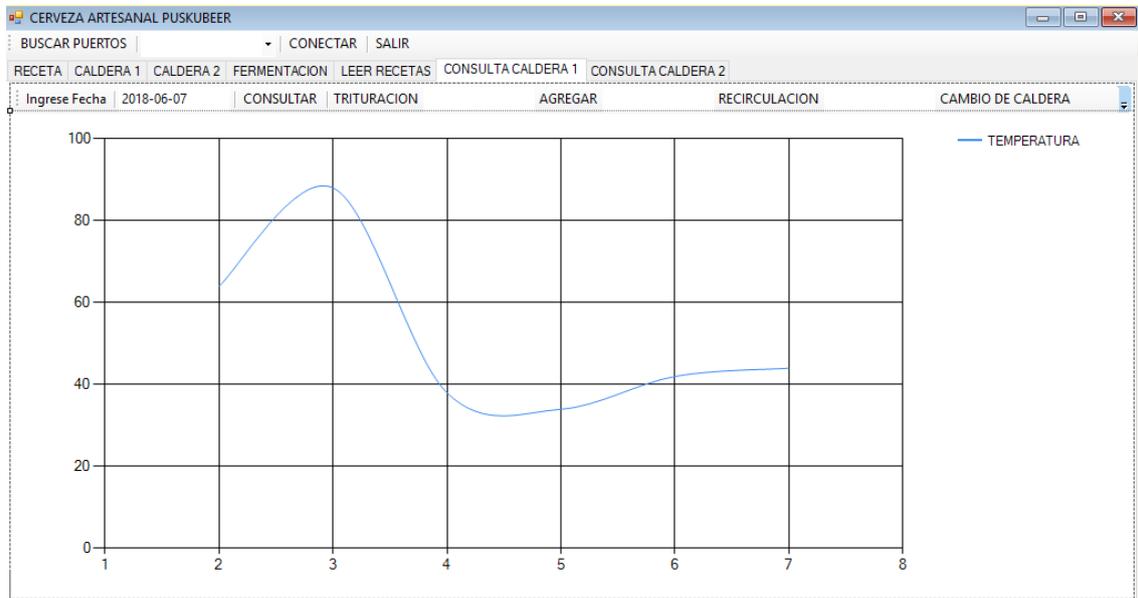


Figura 28 Interfaz de gráfica de consulta de caldero 1 y 2

El almacenamiento de los datos capturados se registra en una base de datos, la cual es de mayor utilidad para el correspondiente análisis del proceso o a su vez para modificar o elaborar de nuevas recetas.

3.7.3 Base de datos.

El almacenamiento de los registros y de los datos capturados se lo realiza en las tablas creadas bajo la interfaz de MySQL, en la cual se puede consultar los eventos obtenidos a lo largo de la elaboración de la cerveza artesanal. Tal como se puede observar en la figura 29.

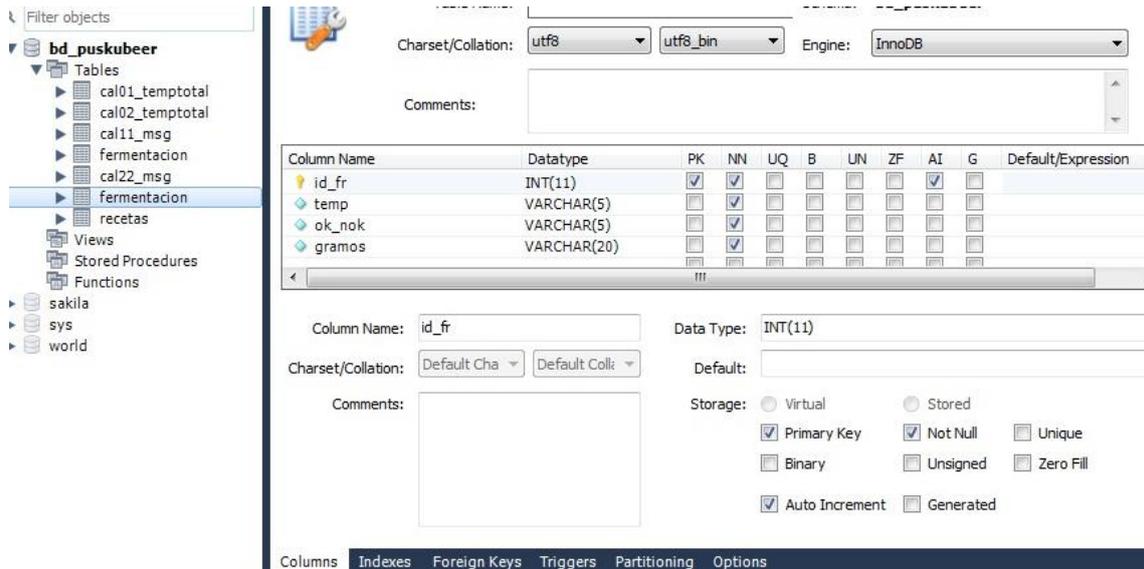


Figura 29 Tablas de MySQL

Capítulo IV Protocolo de pruebas.

El presente proyecto cuenta con la etapa de pruebas de funcionalidad, las cuales constan de la comprobación de la activación y lectura de dispositivos electrónicos de cada etapas de la elaboración del producto, con la secuencia del funcionamiento, para esto se realiza ensayos con los parámetros de mayor jerarquía en el proceso de elaboración como es el tiempo y la temperatura. Estas medidas en primera instancia se comparan los resultados con equipos similares de tal manera que se obtenga una referencia y se logre contrastar resultados.

La metodología que se tomó de referencia es la siguiente:

- Iniciar de forma secuencial la elaboración.
- Seleccionar el tipo de cerveza para su elaboración.
- Escoger el dispositivo de referencia con el cual se realizarán comparaciones.
- Analizar los resultados.

4.1 Registro de datos

El registro de datos se lo hace mediante el programa implementado para la elaboración, en el cual se ingresa el nombre, la densidad y la materia prima con su respectiva cantidad a utilizar en la elaboración de la cerveza artesanal, además de la confirmación de cada etapa realizada, como se puede observar en la figura 30.

Figura 30 Registro de datos

4.2 Captura de datos

4.2.1 Monitoreo de temperatura

La medición de temperatura se lo realiza a través de la termocupla encargada de recolectar estos datos DS18B20, el mismo que tiene la capacidad de medida en grados centígrados, es importante conocer la temperatura en cada fase del proceso de maceración como el de hervido, el mismo que cuenta con un nivel de volumen, ya que de ambos procesos son factores de importantes para la obtención de un producto de calidad.

Primera etapa de maceración con temperatura a 63 °C, tal como se indica en la figura 31.



Figura 31 Primera etapa de maceración

Segunda etapa de maceración con temperatura a 72 °C se presenta en la figura 32.



Figura 32 Segunda etapa de maceración

Tercera etapa de maceración con temperatura a 78 °C, como se puede observar en la figura 33.



Figura 33 Tercera etapa de maceración

Etapa de hervido con temperatura a 90 °C, como se observa en la figura

La datos presentados se muestran en la imagen 34.



Figura 34 Etapa de hervido.

4.3 Gestión Funcional del prototipo electrónico.

4.3.1 Mezclado

El accionamiento del agitador se lo realiza de forma manual luego de obtener una temperatura de condición. La misma que se realiza por 10 minutos determinados y se puede observar la activación a través de una luz piloto indicada en el panel de control, debido a que es una de las etapas más trascendental, tal como se indica en la figura 35.



Figura 35 Accionamiento del mezclado

4.3.2 Recirculación.

La etapa de recirculación se lo realiza de forma automática y secuencial por un tiempo de 20 minutos luego de culminación de los tres procesos de maceración con una temperatura final de 78 °C, como se puede apreciar en la figura 36.



Figura 36 Recirculación

4.3.3 Cambio de caldera

El cambio de cambio de caldera se lo realiza de forma manual luego de haber terminado la etapa de recirculación. La misma que se realiza por 20 minutos determinados que es el tiempo en el cual se termina de traspasar todo el líquido de una caldera a otra, igualmente se puede observar la activación a través de una luz piloto indicada en el panel de control, debido a que es de mayor importancia, tal como se puede observar en la figura 37.



Figura 37 Cambio de caldera

4.3.4 Cambio de temperatura.

El proceso de cambio de temperatura se lo realizó gracias a la inicialización de manera automática y secuencial al culminar los 60 minutos y con una temperatura promedio de 90 ° C, mediante el accionamiento de la bomba succionadora la misma que se conecta al intercambiador de temperatura y a su salida se obtiene el líquido a temperatura ambiente. Como se puede apreciar en la figura 38.

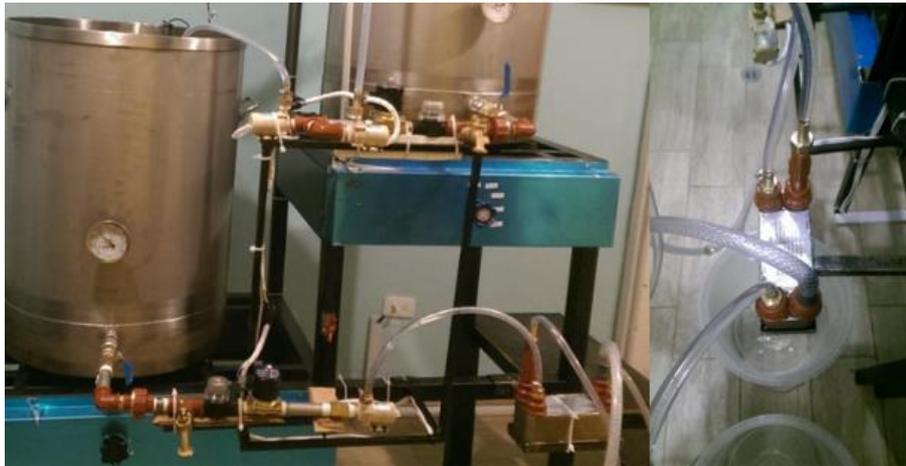


Figura 38 Cambio de temperatura

4.3.5 Paro de emergencia

El presente sistema cuenta con botón de paro de emergencia, que al accionarse detiene el proceso actual del sistema en ejecución. Al ser accionado dicho botón se muestra visualmente la intermitencia de un led lo cual se indica que existe un bloqueo de emergencia accionado, éste botón al desenclavar regresa al inicio del proceso. Adicionalmente se realizará una gestión de alarmas mediante mensajes de texto al activar o desactivar el paro de emergencia, tal como se puede observar en la figura 39.



Figura 39 Paro de emergencia

4.3.6 Gestión de alarmas

Las notificaciones se realizan desde la inicialización del sistema y todas las etapas de los procesos de maceración los mismo que se lo adquiere mediante mensajes escritos, tal como el monitoreo de las tres fases de la maceración, también el estado de inicio y fin de la agitadora, recirculación, cambio de recipiente. Para entender mejor se explica la tabla 15 con los tiempos de procesamiento.

Tabla 15
Tiempos de procesamiento de la Caldera 1

MACERACION 90 min				
Maceración			Recirculación	Cambio de Caldera
60 min	15 min	5 min	20 min	15 min
63 °C	72 °C	78 °C		

En la figura 40 se indica las notificaciones.



Figura 40 Notificaciones del proceso de maceración

Igualmente, los periodos del proceso de hervido, como es el monitoreo de la temperatura y las etapas para agregar las diferentes especies.

En seguida se muestra la tabla 16 , con los tiempos de procesamiento de la caldera 2.

Tabla 16
Tiempos de procesamiento de la Caldera 2

Hervido 60 min				
Azúcar	L. Sabor	L. aroma	Fin de Hervido	Cambio de temperatura
40 min	5 min	10 min	5 min	15 min

En la figura 41 se muestran las notificaciones de las etapas..

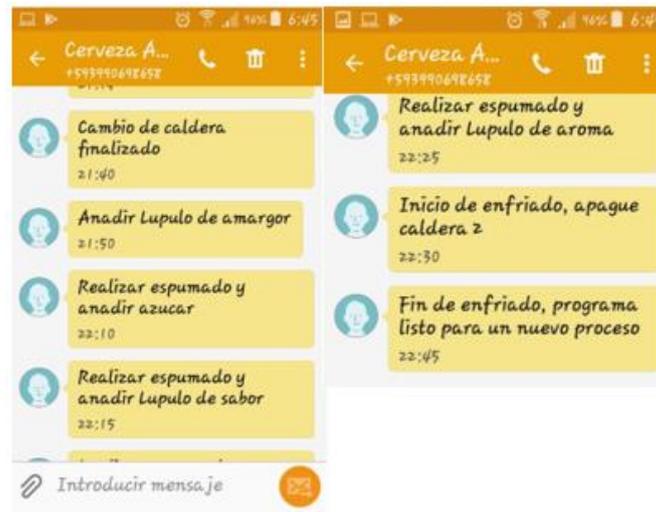


Figura 41 Notificaciones del proceso de hervido.

Cada etapa cuenta con un tiempo respectivo, ya que se especifica en cada procedimiento a realizar. De la misma forma el sistema de emergencia notifica sus ambos estados,

Como se puede observar en la figura 42.

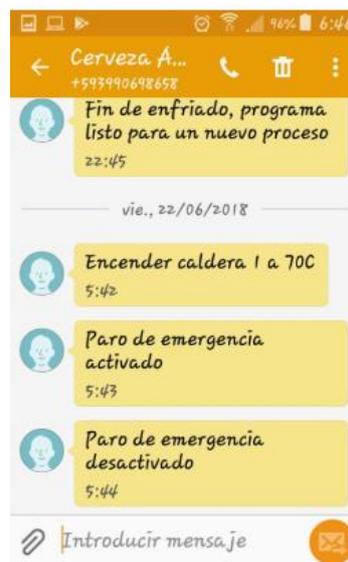


Figura 42 Notificaciones del paro de emergencia

4.3.7 Consulta de etapas

La consulta se realiza en el programa de la elaboración de cerveza, el mismo que contiene; el registro de la receta realizada, las temperaturas obtenidas durante los procesos de maceración y hervido, además de los procedimientos

realizados. Todas las consultas se realizan con la fecha de elaboración que es el parámetro válido, tal como se indica en la figura 43, 44 y 45.

Figura 43 Consulta de receta ingresada.

Este procedimiento se lo realizaba mediante un registro manual, en el cual contemplaba obtener las temperaturas del proceso de maceración, las mismas que se lo realizan con respecto al tiempo. De tal manera se obtenía las cuatro medidas a través de un termómetro de mercurio. En la figura 44 se muestra las medidas del proceso de maceración de forma manual.

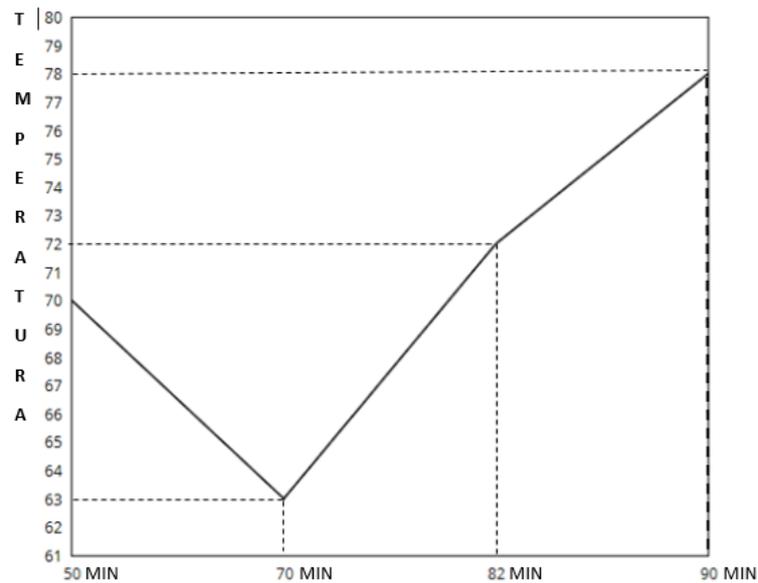


Figura 44 Gráfico de la temperatura del proceso de maceración antes del prototipo.

En la figura 45 se muestra los valores obtenidos al momento de realizar el proceso de maceración, en la cual registra el incremento de la temperatura desde las 19:00 y las variantes respectivas durante la elaboración.

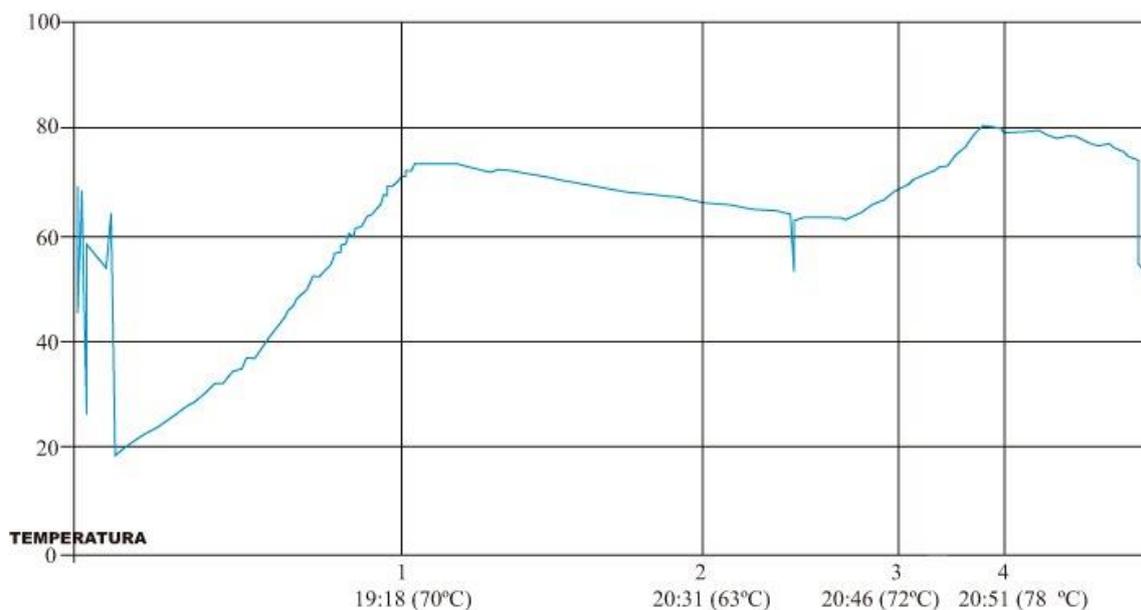


Figura 45 Gráfico de la temperatura del proceso de maceración.

Antes de implementar el prototipo no se pudo mantener un registro total de la temperatura durante la etapa de hervido.

En la figura 46 se muestra las temperaturas obtenidas durante el proceso de hervido, en la cual se puede observar que a las 21:48 se establece una temperatura de 90 °C la misma que se mantiene hasta la culminación de la etapa, a esta tendencia se la conoce como una infusión simple.

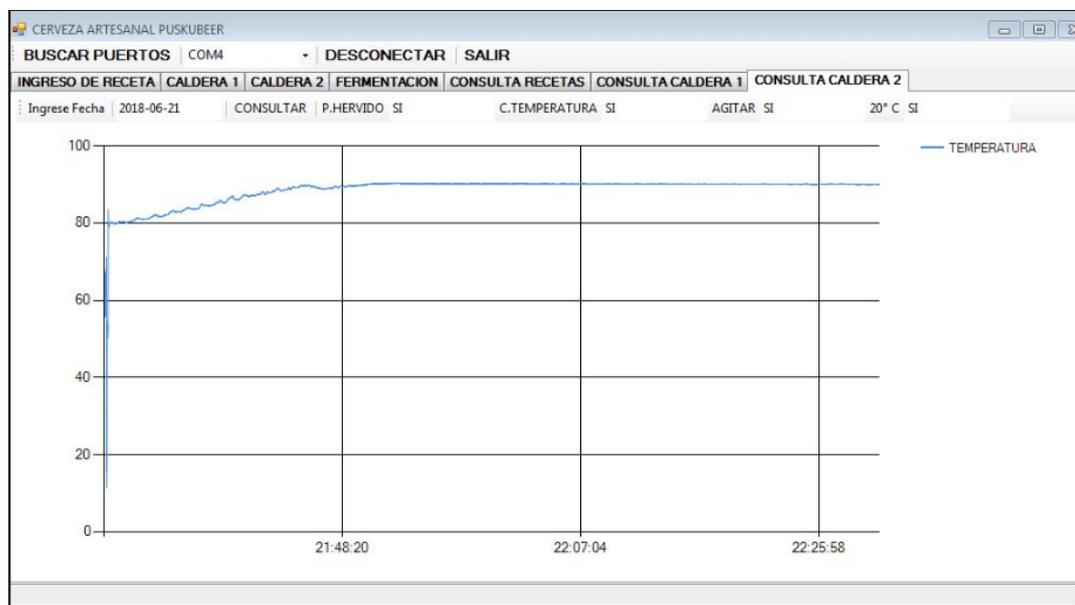


Figura 46 Grafico del proceso de hervido

El análisis de la temperatura se lo realiza con respecto al tiempo, el cual muestra un gráfico de tendencia la misma que es de interés para la creación de nuevos productos.

Una comparación entre ambos procesos de elaboración se puede resaltar que en la implementación del prototipo se obtiene las medidas de temperatura durante todo el proceso de elaboración, el cual sirve para previo análisis y comparativas de nuevos procesos.

4.4 Propuesta para sistema de respaldo

El sistema de respaldo puede tener un método de monitoreo, en el cual conste de focos indicadores el mismo que notifique cada evento del proceso de elaboración, los cuales que deberán encenderse para cada etapa de la maceración como el de hervido, además en el prototipo puede implementarse un display, en el cual muestre los diferentes parámetros, tales como la temperatura y el tiempo.

El motivo de implementar un sistema de respaldo es facilitar el manejo y manipulación del prototipo al usuario, pretendiendo suministrar una solución en caso de falla en algún sitio del sistema.

Capítulo V Guía de usuario

Para el presente proyecto se ha decidido implementar una guía de usuario para utilizar el software de la realización de la cerveza artesanal. En la cual refleje los pasos de cada etapa de funcionamiento.

5.1 Procedimientos.

Primer paso: se debe energizar el sistema desde el panel de control, ubicarlo en estado ON, posteriormente se encenderá la luz piloto la indica que el sistema se encuentra listo. Tal como se indica en la figura 47



Figura 47 Sistema Energizado y Listo.

Segundo paso: ejecutar el programa Puskubeer.

Tercero paso: Buscar puertos de comunicación y presionar aceptar, tal como se puede observar en la figura 48.

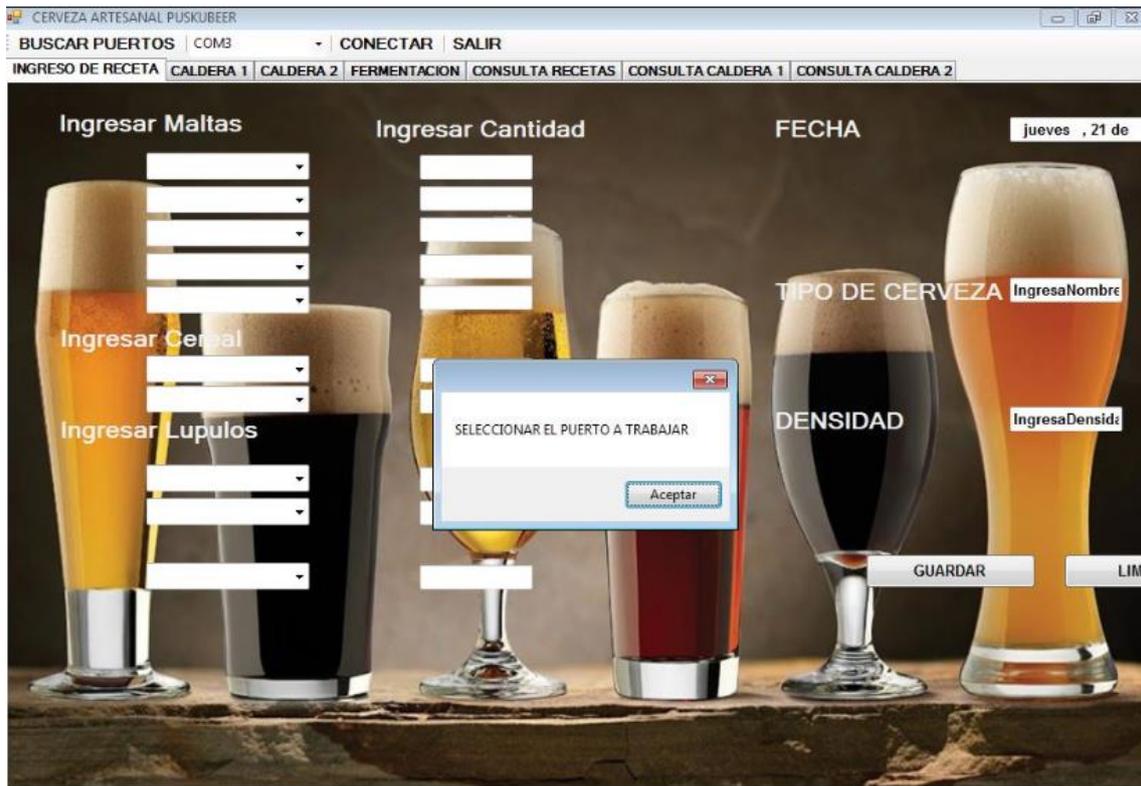


Figura 48 Búsqueda de puerto serial

Cuarto paso: seleccionar el puerto serial COM4 y establecer conectar, tal como se indica en la figura 49.



Figura 49 Conexión de puerto serial

Quinto paso: Ingreso de ingredientes, cantidad, nombre y densidad. A continuación, presionar GUARDAR. Y confirmar los que los datos fueron ingresados correctamente.

De no tener conformidad con los datos ingresados presionar LIMPIAR, como se puede observar en la figura 50.

The screenshot shows a web application interface for 'CERVEZA ARTESANAL PUSKUBEER'. The main content area is divided into several sections for entering recipe data:

- Ingresar Maltas:** A list of malt types with dropdown menus: MALTA PALE ALE, MALTA CARA RUBY, MALTA CHOCOLATE, MALTA BLACK, MALTA SPECIAL B.
- Ingresar Cereal:** A list of cereal types with dropdown menus: CEREAL, CAMELO.
- Ingresar Lupulos:** A list of hop types with dropdown menus: LUPULO DE AMARCO, LUPULO DE SABOR, LUPULO DE AROMA.
- Ingresar Cantidad:** A column of input fields for quantities, with values: 2250, 360, 300, 60, 20, 200, 200, 15, 15, 15.
- FECHA:** A date input field showing 'jueves , 21 de'.
- TIPO DE CERVEZA:** A dropdown menu showing 'STOUT'.
- DENSIDAD:** An input field showing '1045'.

A modal dialog box is open in the center of the screen with the text 'Dato Ingresado Correctamente' and an 'Aceptar' button. At the bottom right, there are buttons for 'GUARDAR' and 'LIMPIAR'.

Figura 50 Ingreso de receta

Sexto paso: seleccionar caldera 1 e ingresar los procedimientos realizados en su totalidad. Luego presionar GUARDAR .No dejar espacios en blanco, tal como se presenta en la figura 51.



Figura 51 Ingreso de procedimientos caldera 1

Sexto paso: seleccionar caldera 2 e ingresar los procedimientos realizados en su totalidad. Luego presionar guardar .No dejar espacios en blanco. Tal como se muestra en la figura 52.



Figura 52 Ingreso de procedimientos caldera 2

Séptimo paso: Ingresar los datos solicitados y presionar GUARDAR, como se presenta en la figura 53.



Figura 53 Ingreso de parámetros de fermentación

Octavo paso: para realizar una consulta de receta se requiere conocer la fecha de elaboración.

Seleccionar leer receta

Ingresar la fecha de la receta para visualizar.

A continuación, se presenta la consulta de receta en la figura 54.



Figura 54 Consulta de Receta

Noveno paso: para realizar una consulta de las calderas se requiere ingresar la fecha de la elaboración producto y presionar CONSULTAR.

A continuación, se muestra la consulta de calderas en la figura 55.



Figura 55 Consulta de calderas

CONCLUSIONES Y RECOMENDACIONES

En el presente se capítulo se manifestará las conclusiones que se pudieron obtener en el transcurso de la implementación del prototipo, así como los resultados observados luego de las pruebas, consecuentemente se incluyen recomendaciones que se pueden tomar como fuentes para el desarrollo y o continuación del protocolo.

6.1 Conclusiones

Se logro diseñar un sistema electrónico para el proceso de elaboración de cerveza artesanal, por lo que se concluye que este sistema ayudo a optimizar los tiempos de cocción , debido a que permite controlar las temperaturas, de tal estas dos variables son directamente proporcionales como se indica en las figuras 44 y 45 que representan a las calderas 1 y 2 respectivamente.

La implementación del sistema de control y la adquisición de nuevos equipos como: calderas, agitador, intercambiador de calor, panel de control, hardware, software, facilito la elaboración de cerveza artesanal permitiendo controlar estos procedimientos mediante la gestión de alarmas por mensaje.

La red GSM facilitó la gestión de alarmas debido a que su costo por SMS es accesible por las diferentes operadoras.

Se utilizó la red móvil de Claro debido a que tiene mejor cobertura en el Sur de Quito.

Los dispositivos electrónicos utilizados requieren de un voltaje de 5V DC, debido a que por problemas de voltaje no se accionan las termocuplas correctamente y no es posible en envío de mensajes de texto.

La fuente de voltaje DC utilizada se instaló fuera del panel de control debido a que este dispositivo genera una intermitencia de comunicación con el módulo SIM 900.

La antena del módulo Gsm se la instaló fuera del panel por razones de cobertura con la red móvil.

Los gráficos visualizados a través del programa fueron de mayor importancia en la elaboración del producto, puesto que los datos visualizados se utilizan para la preparación de nuevas recetas.

La base de datos se realizó en MySQL Workbench dado que su requerimiento de hardware es bajo a comparación de otros

6.2 Recomendaciones

Para mejorar la automatización de la elaboración de cerveza artesanal se recomienda adquirir un equipo de fermentación y así optimizar el tiempo, puesto que de forma manual debe fermentarse en aproximadamente es de 15 a 30 días.

Verificar la cobertura de la operadora a utilizar en este tipo de proyectos, de tal manera nos permita conocer la ubicación adecuada del panel de control.

Se debe adquirir una fuente DC para la energización ideal para que no interfiera en la señal de la red móvil.

Se debe mantener las conexiones eléctricas adecuadas en el área, en donde se va a realizar la instalación, dado que en el sur de Quito la variación de voltaje son continuas, así como formación de tormentas eléctricas.

REFERENCIAS

- Arduino. (s.f). ARDUINO MEGA 2560 R3. Recuperado el 19 de marzo de 2018, de <http://arduino.cl/arduino-mega-2560/>
- Arduino. (s.f). Estructura de un programa. Recuperado el 29 de mayo de 2018, de <https://playground.arduino.cc/ArduinoNotebookTraduccion/Structure>
- Banshee. (s.f). GPRS/GSM Shield SIM900 Arduino (Celular Shield). Recuperado el 27 de Mayo de 2018, de <https://banshee.mx/shields-arduino/gprsgsm-shield-sim900-arduino-celular-shield/62>
- Bricogeek. (s.f). Sensor DS18B20 estanco. Recuperado el 27 de abril de 2018, de <http://tienda.bricogeek.com/sensores-temperatura/510-sensor-ds18b20-estanco.html>
- Calderoni Padilla, J. L. (2012). Proceso de Molienda de la Malta para cerveza. Recuperado el 21 de febrero de 2018, de Verema: <https://www.verema.com/blog/el-blog-del-cervecerero/1000485-proceso-molienda-malta-para-cerveza>
- Ccm. (s.f). Estándar GSM (Sistema global de comunicaciones móviles). Recuperado el 20 de marzo de 2018, de <https://es.ccm.net/contents/681-estandar-gsm-sistema-global-de-comunicaciones-moviles>
- Cervecearte. (s.f). Guía rápida con consejos prácticos para la elaboración de cerveza en casa a partir de una receta de America Pale Ale. Recuperado el 2 de abril de 2018, de http://cervecearte.com/wp-content/uploads/Al_Grano__Guia_rapida_con_consejos_practicos_para_la_elaboracion_de_cerveza_en_casa.pdf
- Cervesamontseny. (s.f). Las 7 diferencias entre la cerveza artesana y la industrial. Obtenido de <https://cervesamontseny.cat/es/las-7-diferencias-entre-la-cerveza-artesana-y-la-industrial/>

- Daniels, R. (1998). *Designing Great Beers*. Estados Unidos: Brewers Publications .
- Diosdado, R. (s.f). Sensor de ultrasonidos HC-SR04. Recuperado el 2 de abril de 2018, de Zona Marquer: <https://www.zonamaker.com/arduino/modulos-sensores-y-shields/ultrasonido-hc-sr04>
- Ebay. (2018). Bayite BYT-7A006. Recuperado el 20 de abril de 2018, de https://www.ebay.com/itm/Bayite-BYT-7A006-DC-12-V-Calentador-Solar-de-Agua-Caliente-Bomba-de-circulacion-3-M-de-bajo-ruido-/282865602660?_ul=AR
- Ebay. (sf). Hydraulic Equipment. Recuperado el 13 de abril de 2018, de https://www.amazon.com/HFS-Electric-Solenoid-Valve-Water/dp/B018WRJYX6/ref=sr_1_4?s=digital-text&ie=UTF8&qid=1522962352&sr=8-4&keywords=valves%2B1%2F2%22&th=1
- Fain, P. A. (2009). pablofain. Recuperado el 2 de abril de 2018, de <https://www.pablofain.com/como-funciona-la-red-gsm/>
- Geekfactory. (s.f). Productos. Recuperado el 31 de 05 de 2018, de <https://www.geekfactory.mx/tienda/relevadores-y-switches/modulo-de-16-relevadores-con-optoacoplador/>
- HetPro. (2014). Modulo de Comunicacion Inalambrica GSM Shield. Recuperado el 23 de Mayo de 2018, de <https://hetpro-store.com/TUTORIALES/modulo-de-comunicacion-inalambrica-gsm-shield/>
- Huxley, S. (2011). *La cerveza poesia Líquida* . España: Ediciones Trea, S. L.
- INEN. (2003). *Bebidas alcohólicas. Cerveza. Requisitos*. NTE INEN2262:2003. Quito, Pichincha, Ecuador: INEN.

La hora. (2008). Quito fue la cuna de la cerveza. Recuperado el 23 de marzo de 2018, de <https://lahora.com.ec/noticia/757382/quito-fue-la-cuna-de-la-cerveza--->

La molienda del grano. (2012). Recuperado el 15 de febrero de 2018, de La molienda del grano: <http://eltallercervezero.blogspot.com/2012/08/la-molienda-del-grano.html>

leon, Nuevo. (s.f). Volumen y capacidad. Recuperado el 21 de marzo de 2018, de http://nuevoleon.inea.gob.mx/MEVyT/Disco3/cursos/numhogar/nch02_50.html

Microsoft. (2015). Introducción al lenguaje C# y .NET Framework. Recuperado el 19 de Abril de 2018, de Introducción al lenguaje C# y .NET Framework: <https://docs.microsoft.com/es-es/dotnet/csharp/getting-started/introduction-to-the-csharp-language-and-the-net-framework>

Microsoft. (2018). Introducción al IDE de Visual Studio. Obtenido de <https://docs.microsoft.com/es-es/visualstudio/ide/visual-studio-ide>

Microsoft. (s.f.). Access SQL: conceptos básicos, vocabulario y sintaxis. Recuperado el 2 de Mayo de 2018, de Access SQL: conceptos básicos, vocabulario y sintaxis: <https://support.office.com/es-es/article/access-sql-conceptos-b%C3%A1sicos-vocabulario-y-sintaxis-444d0303-cde1-424e-9a74-e8dc3e460671#bm1>

Mysql. (2009). MYSQL-WORKBENCH, UNA BUENA HERRAMIENTA PARA LAS BASES DE DATOS. Obtenido de <https://rinconinformatico.net/mysql-workbench-una-buena-herramienta-para-las-bases-de-datos/>

Naylampmechatronics. (sf). Naylampmechatronics. Recuperado el 4 de junio de 2018, de <https://naylampmechatronics.com/conversores-dc-dc/125-convertidor-voltaje-dc-dc-step-up-2p5a-xl6009.html>

Tinyosshop. (s.f). GSM/GPRS Shield Datasheet. Recuperado el 24 de marzo de 2018, de Tinysine: <https://www.tinyosshop.com/datasheet/GSM%20Shield%20Datasheet.pdf>

Verema, & Calderoni Padilla, J. L. (2012). Verema. Obtenido de Verema.com: <https://www.verema.com/blog/el-blog-del-cervecerero/1000485-proceso-molienda-malta-para-cerveza>

Vialfa, C. (2017). Ccm. Obtenido de <https://es.ccm.net/contents/681-estandar-gsm-sistema-global-de-comunicaciones-moviles>

ANEXOS

Anexo 1 Programación Arduino MEGA

```
#include <OneWire.h>
```

```
#include <DallasTemperature.h>
```

```
#include "SIM900.h"
```

```
#include <SoftwareSerial.h>
```

```
//SoftwareSerial SIM900(18, 19);
```

```
#include "sms.h"
```

```
MSGSMS sms;
```

```
int numdata;
```

```
boolean started=false;
```

```
char smsbuffer[160];
```

```
char n[20];
```

```
char sms_position;
```

```
char phone_number[20]; // array for the phone number string
```

```
char sms_text[100];
```

```
int i;
```

```
OneWire ourWire(2); //Se establece el pin 2 como bus OneWire
```

```
DallasTemperature sensors(&ourWire); //Se declara una variable u objeto para nuestro sensor
```

```
DeviceAddress address2 = {0x28, 0xFF, 0x4F, 0xF, 0xA3, 0x16, 0x5, 0x3F}; //dirección del sensor 1
```

```
DeviceAddress address1 = {0x28, 0xFF, 0x11, 0x1C, 0xC0, 0x17, 0x2,  
0xBE};//dirección del sensor 2
```

```
float sensorvalu;
```

```
float sensorvalu2;
```

```
int listo= (20);
```

```
unsigned long periodo = 30000; // tiempo para que el led se encienda
```

```
unsigned long periodo1 = 30000; // tiempo para que el led se encienda
```

```
unsigned long periodo2 = 30000; // tiempo para que el led se encienda
```

```
unsigned long periodo3 = 30000; // tiempo para que el led se encienda
```

```
unsigned long periodo4 = 60000; // tiempo para que el led se encienda
```

```
unsigned long periodo5 = 20000; // tiempo para que el led se encienda
```

```
unsigned long periodo6 = 30000; // tiempo para que el led se encienda
```

```
unsigned long periodo7 = 20000; // tiempo para que el led se encienda
```

```
unsigned long periodo8 = 20000; // tiempo para que el led se encienda
```

```
unsigned long periodo9 = 20000; // tiempo para que el led se encienda
```

```
unsigned long periodo10 = 20000; // tiempo para que el led se encienda
```

```
float temp00;
```

```
float temp02;
```

```
double valor =0;
```

```
byte bandera =0;
```

```
byte temp0 = 1; //estado del temporizador, 1= activo 0= inactivo
```

```
byte temp1 = 0;
```

byte temp2 = 0;//estado del temporizador, 1= activo 0= inactivo

byte temp3 = 0;

byte temp4 = 0;

byte temp5 = 0;

byte temp6 = 0;

byte temp7 = 0;

byte temp8 = 0;

byte temp9 = 0;

byte temp10 = 0;

byte temp11 = 0;

byte temp12 = 0;

byte temp13 = 0;

byte temp14 = 0;

byte temp15 = 0;

byte temp16 = 0;

byte temp17 = 0;

byte temp18 = 0;

byte temp19 = 0;

byte temp20 = 0;

byte temp21 = 0;

byte temp22 = 0;

byte temp23 = 0;

```
byte temp24 = 0;

byte temp25 = 0;

byte temp26 = 0;

byte temp27 = 0;

byte temp28 = 0;

byte tempE = 0;

byte tempE1 = 1;

unsigned long tiempoAnterior = 0; //guarda tiempo de referencia para comparar

unsigned long tiempoAnterior1 = 0;

unsigned long tiempoAnterior2 = 0;

unsigned long tiempoAnterior3 = 0;

unsigned long tiempoAnterior4 = 0;

unsigned long tiempoAnterior5 = 0;

unsigned long tiempoAnterior6 = 0;

unsigned long tiempoAnterior7 = 0;

unsigned long tiempoAnterior8 = 0;

unsigned long tiempoAnterior9 = 0;

unsigned long tiempoAnterior10 = 0;

void setup() {

    pinMode(listo,OUTPUT); // gsm listo

    pinMode(11,OUTPUT); //configura la salida motor

    pinMode(12,OUTPUT); //configura la salida ev1
```

```
pinMode(23,OUTPUT); //configura la salida b1

pinMode(24,OUTPUT); //configura la salida b2

pinMode(25,OUTPUT); //configura la salida ev22

pinMode(26,OUTPUT); //configura la salida b3

pinMode(40,OUTPUT); // paro de emergencia

pinMode(8,INPUT); //configura la entrada

pinMode(9,INPUT); //configura la entrada vaciado

pinMode(10,INPUT); //configura la entrada vaciado

//Serial.begin(9600);

//sensors.begin(); //Se inicia el sensor

//delay(1000);

Serial.begin(9600);

Serial.println("GSM Shield testing.");

if (gsm.begin(4800))

{

    Serial.println("\nstatus=READY");

    started=true;

    digitalWrite(listo, LOW);

}

else

    Serial.println("\nstatus=IDLE");

for(i=1;i<=20;i++)
```

```
    {  
        sms.DeleteSMS(i);  
    }  
}  
  
void loop() {  
  
    if (digitalRead(10)==1 ){  
        tempE=1;  
  
        digitalWrite(40,HIGH);  
  
    }  
  
    if (digitalRead(10)==0 && tempE1==1){  
  
        digitalWrite(11,HIGH);  
  
        digitalWrite(12,HIGH);  
  
        digitalWrite(23,HIGH);  
  
        digitalWrite(24,HIGH);  
  
        digitalWrite(25,HIGH);  
  
        digitalWrite(26,HIGH);  
  
tempE=0;  
  
temp0 = 1; //estado del temporizador, 1= activo 0= inactivo  
  
temp1 = 0;  
  
temp2 = 0;//estado del temporizador, 1= activo 0= inactivo  
  
temp3 = 0;  
  
temp4 = 0;
```

temp5 = 0;

temp6 = 0;

temp7 = 0;

temp8 = 0;

temp9 = 0;

temp10 = 0;

temp11 = 0;

temp12 = 0;

temp13 = 0;

temp14 = 0;

temp15 = 0;

temp16 = 0;

temp17 = 0;

temp18 = 0;

temp19 = 0;

temp20 = 0;

temp21 = 0;

temp22 = 0;

temp23 = 0;

temp24 = 0;

temp25 = 0;

temp26 = 0;

```
tempE1= 0;

temp27 = 1;

temp28 = 1;

}

if (temp27==1 && digitalRead(10)==0 ) {

    if(sms.SendSMS("+593980294805", "Paro de emergencia activado"))

        Serial.println("\nSMS00 sent OK");

    delay (100);

    temp27 = 0;

}

if (temp28==1 && digitalRead(10)==0 ) {

    digitalWrite(40,LOW);

    delay (300);

    digitalWrite(40,HIGH);

}

if (temp28==1 && digitalRead(10)==1) {

    temp28 = 0;

    tempE1 = 1;

    if(sms.SendSMS("+593980294805", "Paro de emergencia desactivado"))

        Serial.println("\nSMS0 sent OK");

    delay (100);

}
```

```

if (digitalRead(8)==0 && temp0==1 && tempE==1){

    digitalWrite(11,HIGH);

    digitalWrite(12,HIGH);

    digitalWrite(23,HIGH);

    digitalWrite(24,HIGH);

    digitalWrite(25,HIGH);

    digitalWrite(26,HIGH);

}

if (sensorvalu>=28 && sensorvalu<=40 && temp0==1 && tempE==1){ // 70°C
si el pulsador fue presionado

    //sms.SendSMS("+593995026750", "Anadir Mosto, bajar temperatura a
63°C e iniciar proceso");

    if(sms.SendSMS("+593980294805", "Anadir Mosto, bajar temperatura a 63C
e iniciar proceso"))

        temp0=0;

        temp1=1;

        delay (100);

        Serial.println("\nSMS1 sent OK");

delay (100);

}

if (digitalRead(8)==1 && temp1==1 && tempE==1) { //si el pulsador fue
presionado

    if(sms.SendSMS("+593980294805", "Proceso de mezclado iniciado"))

```

```

temp1 = 0;

temp3 = 1;

delay (100);

//sms.SendSMS("+593995026750", "Proceso de mezclado iniciado");

digitalWrite(11,LOW); //enciendo el LED

Serial.println("\nSMS2 sent OK");

delay (100);

}

if (temp3==1 && tempE==1) {

    tiempoAnterior=millis(); //guarda el tiempo actual como referencia

    temp3 = 0;

    temp4 = 1;

}

if((millis())>tiempoAnterior+periodo) && temp4==1 && tempE==1){ //si ha
transcurrido el periodo programado

    digitalWrite(11,HIGH); //apago el LED

    temp4 = 0;

    temp5 = 1; //y desactivo el temporizador

}

if (temp5==1) {

if(sms.SendSMS("+593980294805", "Proceso de mezclado finalizado"))

    Serial.println("\nSMS3 sent OK");

```

```

    delay (100);

}

if (temp5==1 && tempE==1) {

    tiempoAnterior1=millis(); //guarda el tiempo actual como referencia

    temp5 = 0;

    temp6 = 1;

}

if((millis())>tiempoAnterior1+periodo1) && sensorvalu<=17 && temp6==1 &&
tempE==1){ // 63°C si ha transcurrido el periodo programado

    if(sms.SendSMS("+593980294805", "Subir temperatura a 72C"))

        // delay (100);

        //sms.SendSMS("+593995026750", "Proceso de mezclado finalizado");

        Serial.println("\nSMS4 sent OK");

        delay (100);

        temp6 = 0;

        temp7 = 1; //y desactivo el temporizador

}

if (temp7==1 && tempE==1) {

    tiempoAnterior2=millis(); //guarda el tiempo actual como referencia

    temp7 = 0;

```

```

temp8 = 1;

}

if((millis())>tiempoAnterior2+periodo2) && sensorvalu>=22 && temp8==1 &&
tempE==1){ // 72°C si ha transcurrido el periodo programado

    if(sms.SendSMS("+593980294805", "Subir temperatura a 78C"))

//delay (100);

//sms.SendSMS("+593995026750", "Proceso de mezclado finalizado");

Serial.println("\nSMS5 sent OK");

delay (100);

temp8 = 0;

temp9 = 1; //y desactivo el temporizador

}

if (temp9==1 && tempE==1) {

    tiempoAnterior3=millis(); //guarda el tiempo actual como referencia

    temp9 = 0;

    temp10 = 1;

}

if((millis())>tiempoAnterior3+periodo3) && sensorvalu>=25 && temp10==1 &&
tempE==1){ // 78°C si ha transcurrido el periodo programado

    if(sms.SendSMS("+593980294805", "Inicio recirculacion"))

//delay (500);

digitalWrite(12,LOW); //guarda el tiempo actual como referencia

digitalWrite(23,LOW);

```

```

Serial.println("\nSMS6 sent OK");

delay (100);

temp10 = 0;

temp11 = 1; //y desactivo el temporizador
}

if (temp11==1 && tempE==1) {

    tiempoAnterior4=millis(); //guarda el tiempo actual como referencia

    temp11 = 0;

    temp12 = 1;

    }

if((millis())>tiempoAnterior4+periodo4) && temp12==1 && tempE==1){ // 78°C
si ha transcurrido el periodo programado

    if(sms.SendSMS("+593980294805", "Fin recirculacion"))

//delay (500);

digitalWrite(12,HIGH); //guarda el tiempo actual como referencia

digitalWrite(23,HIGH);

Serial.println("\nSMS7 sent OK");

delay (100);

temp12 = 0;

temp13 = 1; //y desactivo el temporizador

}

if (digitalRead(9)==1 && temp13==1 && tempE==1) { //si el pulsador fue
presionado

```

```
if(sms.SendSMS("+593980294805", "Cambio de caldera iniciado, encienda caldero 2 a 90C"))
```

```
digitalWrite(12,LOW); //enciendo el LED
```

```
digitalWrite(24,LOW); //enciendo el LED
```

```
Serial.println("\nSMS8 sent OK");
```

```
delay (100);
```

```
temp13 = 0;
```

```
temp14 = 1;
```

```
}
```

```
if (temp14==1 && tempE==1) {
```

```
tiempoAnterior5=millis(); //guarda el tiempo actual como referencia
```

```
temp14 = 0;
```

```
temp15 = 1;
```

```
}
```

```
if((millis())>tiempoAnterior5+periodo5) && temp15==1 && tempE==1){ //si ha transcurrido el periodo programado
```

```
if(sms.SendSMS("+593980294805", "Cambio de caldera finalizado"))
```

```
//delay (500);
```

```
//sms.SendSMS("+593995026750", "Proceso de mezclado finalizado");
```

```
digitalWrite(12,HIGH); //apago el LED
```

```
digitalWrite(24,HIGH); //apago el LED
```

```
Serial.println("\nSMS9 sent OK");
```

```
delay (100);
```

```

temp15 = 0;

temp16 = 1; //y desactivo el temporizador
}

if (temp16==1 && tempE==1) {

    tiempoAnterior6=millis(); //guarda el tiempo actual como referencia

    temp16 = 0;

    temp17 = 1;

    }

if (sensorvalu2>=28 && temp17==1 && tempE==1){ // 70°C si el pulsador fue
presionado

    //sms.SendSMS("+593995026750", "Anadir Mosto, bajar temperatura a
63°C e iniciar proceso");

    if(sms.SendSMS("+593980294805", "Anadir Lupulo de amargor"))

temp17=0;

temp18=1;

Serial.println("\nSMS10 sent OK");

delay (100);

}

if((millis())>tiempoAnterior6+periodo6) && temp18==1 && tempE==1){ //si ha
transcurrido el periodo programado

    if(sms.SendSMS("+593980294805", "Realizar espumado y anadir azucar"))

```

```

//delay (500);

temp18 = 0;

temp19 = 1; //y desactivo el temporizador

Serial.println("\nSMS11 sent OK");

delay (100);

}

if (temp19==1 && tempE==1) {

    tiempoAnterior7=millis(); //guarda el tiempo actual como referencia

    temp19 = 0;

    temp20 = 1;

}

if((millis())>tiempoAnterior7+periodo7) && temp20==1 && tempE==1){ //si ha
transcurrido el periodo programado

    if(sms.SendSMS("+593980294805", "Realizar espumado y anadir Lupulo
de sabor"))

        //delay (500);

        //sms.SendSMS("+593995026750", "Proceso de mezclado finalizado");

        temp20 = 0;

        temp21 = 1; //y desactivo el temporizador

        Serial.println("\nSMS12 sent OK");

        delay (100);

```

```
}
```

```
if (temp21==1 && tempE==1) {
```

```
    tiempoAnterior8=millis(); //guarda el tiempo actual como referencia
```

```
    temp21 = 0;
```

```
    temp22 = 1;
```

```
}
```

```
if((millis())>tiempoAnterior8+periodo8) && temp22==1 && tempE==1){ //si ha transcurrido el periodo programado
```

```
    if(sms.SendSMS("+593980294805", "Realizar espumado y anadir Lupulo de aroma"))
```

```
        //delay (500);
```

```
        //sms.SendSMS("+593995026750", "Proceso de mezclado finalizado");
```

```
        temp22 = 0;
```

```
        temp23 = 1; //y desactivo el temporizador
```

```
        Serial.println("\nSMS13 sent OK");
```

```
        delay (100);
```

```
}
```

```
if (temp23==1 && tempE==1) {
```

```
    tiempoAnterior9=millis(); //guarda el tiempo actual como referencia
```

```
    temp23 = 0;
```

```
    temp24 = 1;
```

```

}

if((millis())>tiempoAnterior9+periodo9) && temp24==1 && tempE==1){ //si ha
transcurrido el periodo programado

    if(sms.SendSMS("+593980294805", "Inicio de enfriado, apague caldera 2"))

        //delay (500);

        //sms.SendSMS("+593995026750", "Proceso de mezclado finalizado");

        digitalWrite(25,LOW); //enciendo el LED

        digitalWrite(26,LOW); //enciendo el LED

        temp24 = 0;

        temp25 = 1; //y desactivo el temporizador

        Serial.println("\nSMS14 sent OK");

        delay (100);

}

if (temp25==1 && tempE==1) {

    tiempoAnterior10=millis(); //guarda el tiempo actual como referencia

    temp25 = 0;

    temp26 = 1;

}

if((millis())>tiempoAnterior10+periodo10) && temp26==1 && tempE==1){ //si
ha transcurrido el periodo programado

    if(sms.SendSMS("+593980294805", "Fin de enfriado, programa listo para
un nuevo proceso"))

        //delay (500);

```

```
//sms.SendSMS("+593995026750", "Proceso de mezclado finalizado");

digitalWrite(25,HIGH); //enciendo el LED

digitalWrite(26,HIGH); //enciendo el LED

temp26 = 0;

temp0 = 1; //y desactivo el temporizador

Serial.println("\nSMS15 sent OK");

delay (100);

}

sensors.requestTemperatures(); //Se envía el comando para leer la
temperatura

float temp00= sensors.getTempC(address1); //Se obtiene la temperatura en °C

float temp02= sensors.getTempC(address2); //Se obtiene la temperatura en °C

sensorvalu=temp00;

sensorvalu2=temp02;

Serial.print("Temperatura 1= ");

Serial.print(sensorvalu);

Serial.println(" C");

Serial.print("Temperatura 2= ");

Serial.print(sensorvalu2);

Serial.println(" C");

Serial.print("Temperatura 3= ");

Serial.print(temp12);
```

```
Serial.println(" C");
```

```
delay(100);
```

```
valor= random(40, 80);
```

```
Serial.print("CX"); Serial.print(valor); Serial.println("Y");
```

```
delay(8000);
```

```
valor= random(40, 80);
```

```
Serial.print("CA"); Serial.print(valor); Serial.println("B");
```

```
delay(8000);
```

