Department of Computer Science & Engineering
The University of Texas at Arlington

System Requirements Specification
CSE 4316: Senior Design I
Spring 2020

Team Friendship
Back Burner Brew

Luke Brown
Marcos Juarez Casillas
Ju Young Isa Jung
Sujan Dumaru
Sunghwa Cho
Matthew Francis Schultz

Revision History

Revision	Date	Author(s)	Description
0.1	03.22.2021	LB	document creation
0.2	04.01.2021	ALL	complete draft
1.0	04.06.2021	ALL	official release

CONTENTS

List of Figures

1 Product Concept

The "Back Burner Brew" is built with the sole purpose of brewing large batch of beer in the home environment. This product provides home brewers with a low-cost electric home brewing system that allow them to have precise control over the brewing process. The brewing process can be automated with the help of micro-controllers like the ESP32 which is then hosted to a local website or an app interface.

1.1 Purpose and Use

ESP32 is a micro-controller that can receive data such as current temperature of the water or mash from the heat sensors located inside the kettles which can be converted to either analog or digital input. The heating coil can be control using the input from the user as per their desired either to increase or to decrease the temperature. The electric pump can also be controlled by the user through micro-controller to regulate the flow of the water in the kettles. The user will be able to communicate with the brewing system through a web interface or app interface.

1.2 Intended Audience

The foremost intended audiences for this product would be home brewers or person interested in brewing beer only. Provided that the user manual would be present in the product, any person who wants to brew beer in his local environment can easily use this product. This product is made focusing on how effortless can the brewing process gets simply with the use of micro-controller.

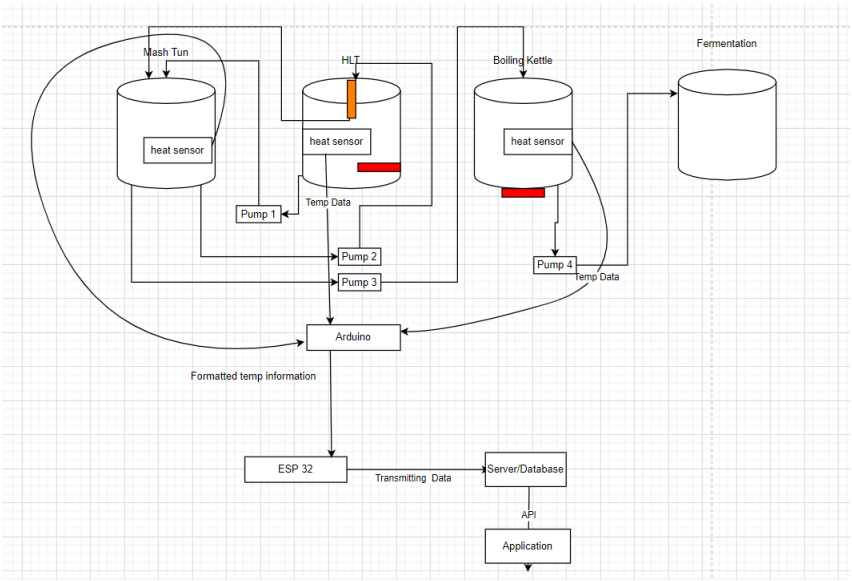

Figure 1: Product Concept

2 Product Description

This section provides the reader with an overview of The Back Burner Brew and will discuss the primary features, functions and interfaces the product will have. The goal of The Back Burner Brew is to automate most of the brewing process to ensure all the liquids heat up and stay at the desired the temperature the brewer wants.

2.1 Features & Functions

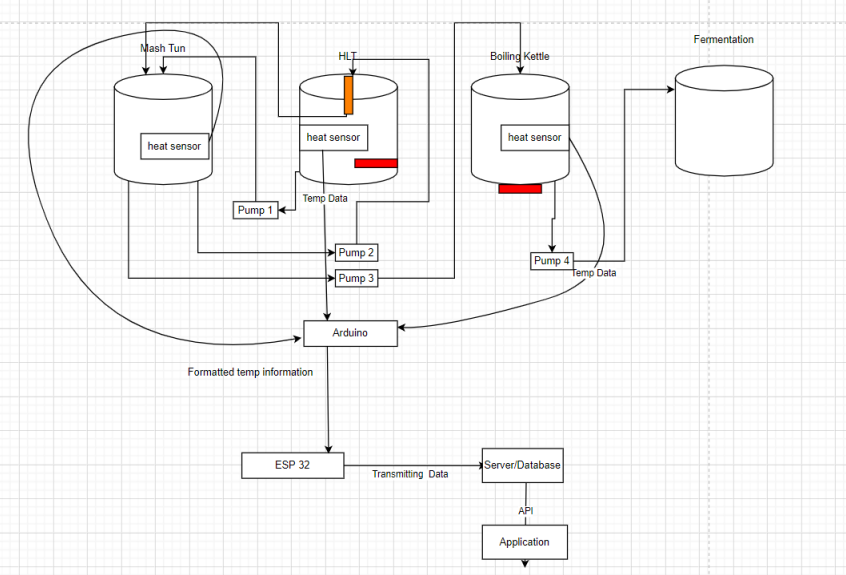

Figure 2: System Overview

The Back Burner Brew will utilize three main kettles and an extra for fermentation as seen in figure 2 and the liquids will be transferred from each kettle through the use of pumps. The Back Burner Brew will have a user web interface connected to a Raspberry Pi. Through this interface, the brewer will be able to control the temperature of all the liquids inside of the Hot Liquor Tank (HLT) and the mash tun. Along with controlling the temperature, the user will be able to set when water from the HLT will be sent to the mash tun and when the mash tun will send the liquids through the HLT for heating. The liquids' temperatures will be monitored through heat sensors and will be sending that data to the web application. The web application will display the current temperature of the liquids to the user and will have a kill switch in case the heating element does not control the temperature of the liquids as requested by the user.

The product will not control the temperature of the boiling kettle, rather the user will be able to control the power output of the heating element inside of the boiling kettle and will have to monitor the temperature and the amount of water loss through the boiling process. Because there will only be three pumps, the user must move connections manually between the pumps in order to have the liquid inside of the boiling kettle be transferred over to the cooler and fermentation kettle. The user must also add in the grains and hops into the different kettles depending on what stage of the brewing process they are in manually.

2.2 External Inputs & Outputs

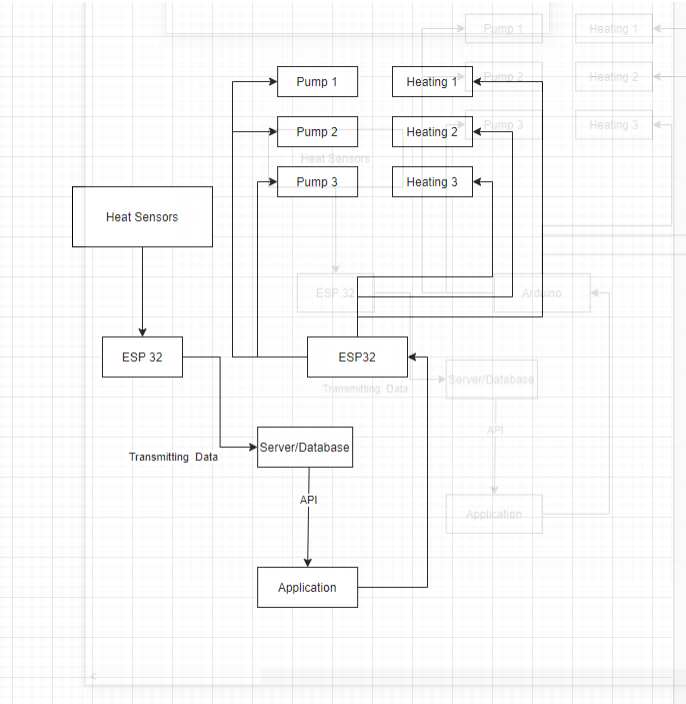

Figure 3: Data Flow and Control

The Back Burner Brew will utilize multiple ESP32's. The ESP32 will be used for reading the temperatures of the different kettles and the Raspberry Pi will be used to control the heating elements and the pumps as well.The ESP32 will read the temperatures of HLT and the mash tun and send them to web server. Once the data has been received by the server, the server will then display that data to the user on the web application. The server will also be able to tell the ESP32 which heating element must be turned on to maintain the desired temperature and which pump to turn on as well.

2.3 Product Interfaces

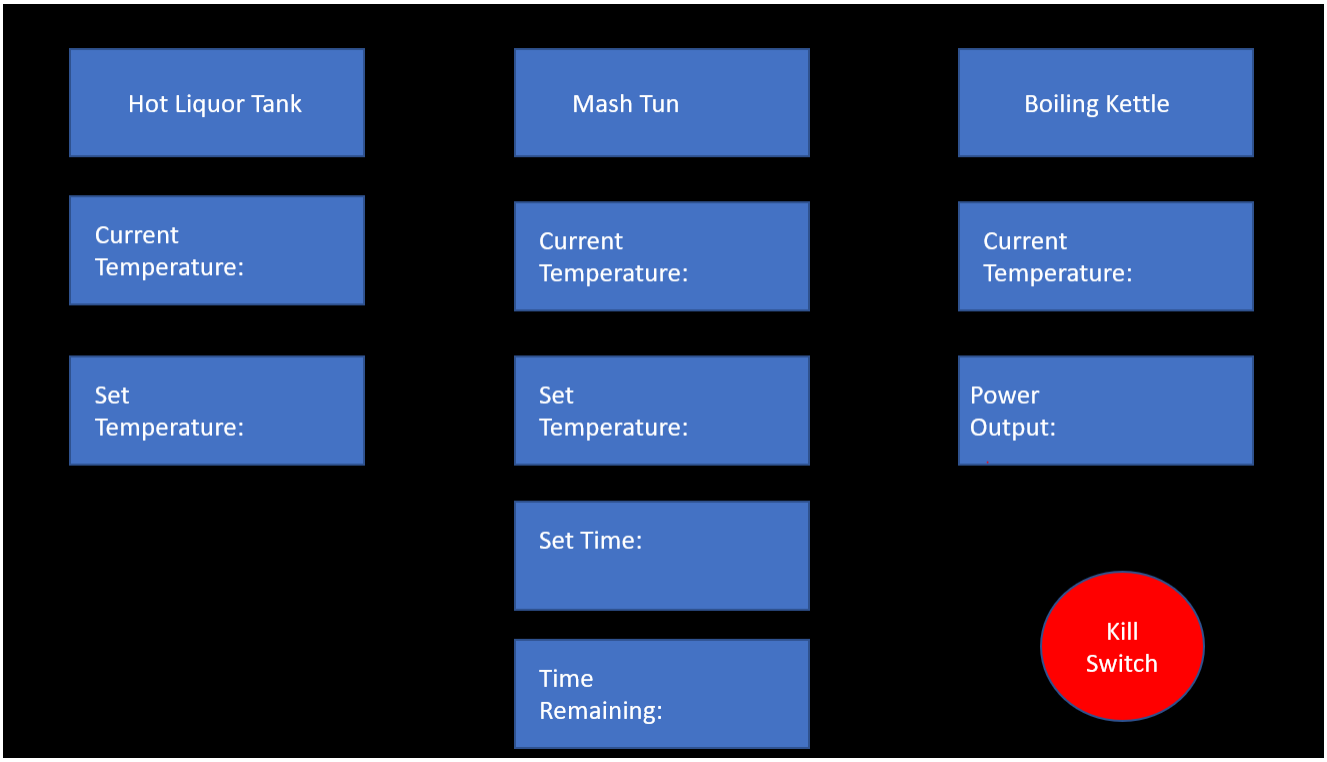

Figure 4: Web Interface Template

The Back Burner Brew user interface template can be seen in the figure above. The web application will allow the user to set the temperature of both the HLT and the Mash Tun and will display the current temperatures as well. The mash tun will also allow for the user to enter the amount of time the mashing process will take, while also displaying the remaining time left on that process. The boiling kettle does not let the user control the temperature of the boiling kettle, rather, it lets the user control the power output of the boiling kettle and they must monitor the temperature of the boiling kettle themselves.

3 Customer Requirements

The user should expect to input desired commands, controls, and specific settings such as temperature and length of time by a easily through web interface. The user can expect that whichever temperature they set for their desired application, that the temperature will remain constant.

3.1 Programmable HLT Heating Element Temperature Management

3.1.1 Description

The heating element on the HLT will need to maintain the temperature that is set by the user via the web interface. The reason for customizable temperature is due to the varying temperature requirements set by the recipe chosen by the user. This temperature control will be managed by an ESP32 microcontroller. The temperature of the vessel will be monitor by a DS18B20 digital thermometer. The microcontroller will take continuous readings from the thermometer. When it begins dropping below the desire temperature, the microcontroller will send a signal to a MOSFET relays. When the relay receives that signal, it switch states and allow for power to be delivered to the heating element. Once the desired temperature is reached, the microcontroller will send another signal to the MOSFET relays. The relay will then turn off, and this would turn off the power to the heating element.

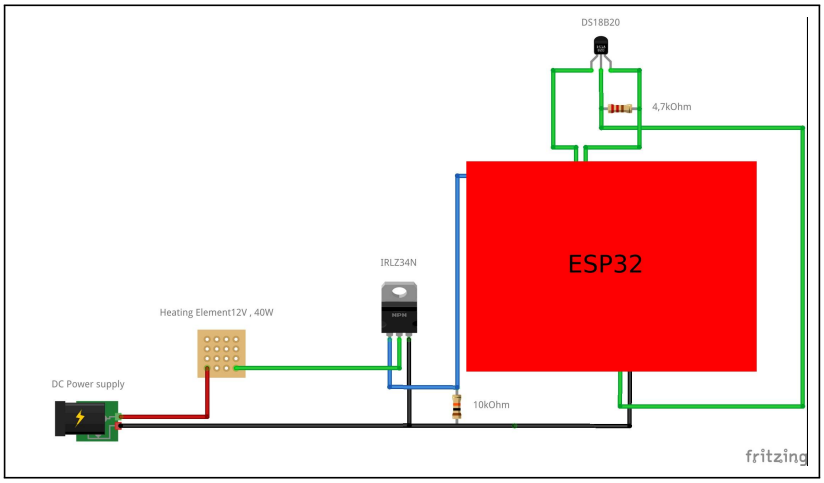

Figure 5: HLT Temperature Controller Diagram

3.1.2 Source

This requirement came from Dr. Chris Conly. He requested that our team enable the user to be able to set a specific temperature via the web interface. The water is to stay at a constant boiling temperature to maximize effectiveness of the mashing process.

3.1.3 Standards

- IEC 60730
- IEC 60335
- EN ISO 13849

3.1.4 Priority

This feature is of **Critical** priority. Without this feature, we would be unable to accomplish an effective mashing process.

3.2 Boiling Kettle Heating Element Temperature Management

3.2.1 Description

The heating element on the boil vessel will need to maintain a water temperature of 212 degrees fahrenheit for optimal extraction. This temperature control will managed by an ESP32 microcontroller. The temperature of the vessel will be monitor by a DS18B20 digital thermometer. The microcontroller will take continuous readings from the thermometer. When it begins dropping below the desire temperature, the microcontroller will send a signal to a MOSFET relays. When the relay receives that signal, it switch states and allow for power to be delivered to the heating element. Once the desired temperature is reached, the microcontroller will send another signal to the MOSFET relays. The relay will then turn off, and this would turn off the power to the heating element.

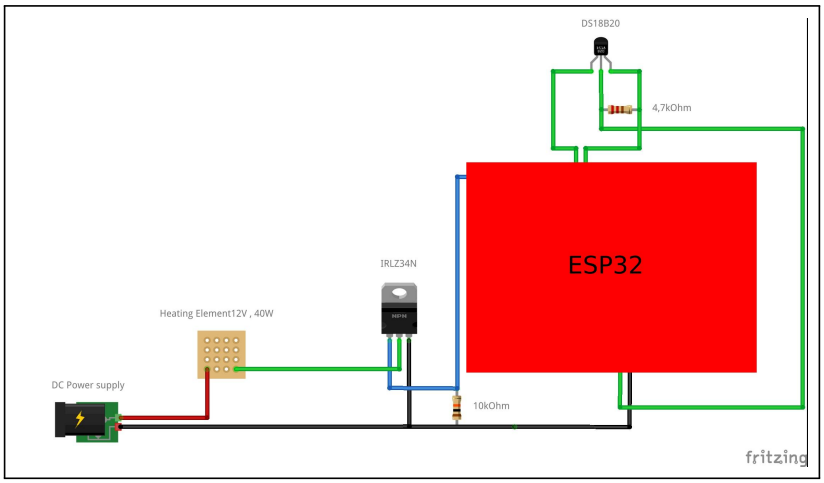

Figure 6: Boiling Kettle Temperature Controller Diagram

3.2.2 Source

This requirement came from Dr. Chris Conly. He requested that our team ensure that the water stay a constant boiling temperature to maximize extraction.

3.2.3 Constraints

Detailed description of applicable constraints...

3.2.4 Standards

- IEC 60730
- IEC 60335
- EN ISO 13849

3.2.5 Priority

This feature is of **Critical** priority. Without this feature we would be unable to accomplish any sort of extraction.

4 Packaging Requirements

The system will be packaged as a set of components that can be attached to any three kettle brewing system. As most customers will have some sort of three kettle brewing system that can simple be modified. All software will be pre-installed onto all electrical hardware.

4.1 Modular part storage

4.1.1 Description

All of the device will be packaged and shipped as a set of components that can be added onto a three kettle brewing system. Packaging will include instructions for hardware installation and setup. Software will be pre-installed in all locations applicable.

4.1.2 Constraints

Water tight packaging is required to protect electrical equipment.

4.1.3 Standards

Not applicable

4.1.4 Priority

This should be given low priority.

5 Performance Requirements

This section will provide performance requirements to maximize the quality of beer that is produced through this project. The performance requirements are response time of micro-controllers, sufficient power supply.

5.1 Response time of micro-controllers

5.1.1 Description

Since the micro-controller using for the brewing system needs to read data from several thermometers in tanks and communicate with a web server to provide a monitoring system, and control pumps, at least ESP32 or higher spec is required. The micro-controller has to follow a specific recipe including the temperature of the hot liquor tank and the mash tun, and time for brewing, etc. Fast response time is necessary to take instance reaction based on data read from the brewing equipment, manage communication with web server, and an internet connection.

5.1.2 Source

Source

5.1.3 Constraints

A monitor connected to ESP32 or higher spec micro-controllers is required for many purposes such as developing software, take immediate action if needed, etc.

5.1.4 Standards

Non-applicable.

5.1.5 Priority

Top-priority should be given as this is core equipment out of entire system.

5.2 sufficient power supply

5.2.1 Description

The time it takes to produce beer to go from raw materials to finished, ready-to-drink beer depends on a number of different factors; however, the typical Light Ales like Cream Ale and Honey Kolsch take a minimum of one week. Other drinks such as Amber Ales, Dark Ales, Light Lagers, and Dark Lagers take much longer to make. While brewing, the equipment has to stay up and keeps exchanging data between a web server. A sufficient power supply is required to keep the equipment and the micro-controller running.

5.2.2 Source

Basic electrical hazards and safety measures

5.2.3 Constraints

Constant checking for electricity is required to prevent any issue related to electricity.

5.2.4 Standards

Non-applicable.

5.2.5 Priority

Top-priority should be given.

6 Safety Requirements

This section will provide brief explanation on safety requirement while using this product. There are various risk factors included while operating the Brewing system. Since we are dealing with electrical appliances, there is always a risk of getting electrocuted if proper attention is not given.

6.1 Safety Hazards while Working with the Electricity

6.1.1 Description

Different electrical appliances like electric heater will be used in this project alongside with water. Even a small misplaced or leakage of water into the micro controller can damage it and can put the user in electrical hazards. While working with mash, it is important to note how hot the mash is in order to prevent burn.

6.1.2 Source

A basic safety measure taken while working with any electrical appliances, especially when a liquid is nearby.

6.1.3 Constraints

The product must be built closed to a place where there are electrical plugs. Once built, the product can be difficult and time-consuming to move. The user should be present at the site while brewing at first for some batches to make sure everything is running smoothly.

6.1.4 Standards

Non-applicable.

6.1.5 Priority

Top-priority should be given as this is concerned with the health of the user.

7 Maintenance & Support Requirements

This section provides the support the team will provide for the user after purchasing our system, such as ways we will be available for the customer to contact us.

7.1 Digital User Manual

7.1.1 Description

The web app will have a digital copy of the user manual available, which the user can download at any time.

7.1.2 Source

Marcos Juarez

7.1.3 Constraints

The server would have to be up at all times to respond to the download request

7.1.4 Standards

N/A

7.1.5 Priority

High

7.2 Bug Reporting

7.2.1 Description

The web app will have a way of reporting bugs to the server so the team can address them.

7.2.2 Source

Marcos Juarez

7.2.3 Constraints

The server would have to always listen for these reports.

7.2.4 Standards

N/A

7.2.5 Priority

Moderate

7.3 Contact us section

7.3.1 Description

A section on the web app with a way to contact the team will be available to the user for technical issues and/or support

7.3.2 Source

Marcos Juarez

7.3.3 Constraints

The team would have to regularly check the email or wherever the customer message is sent.

7.3.4 STANDARDS

N/A

7.3.5 PRIORITY

Low

8 Other Requirements

There are no other requirements needed for the "Back Burner Brew". All requirements have been listed in the previous sections.

8.1 Requirement Name

8.1.1 Description

N/A

8.1.2 Source

N/A

8.1.3 Constraints

N/A

8.1.4 Standards

N/A

8.1.5 Priority

N/A

9 Future Items

9.1 Requirement Name

9.1.1 Description

Due to the time limit of this project we can not accomplish some goals. Being a automated hop feeder. The second item we were considering added was a glycol chiller temp controller.

REFERENCES