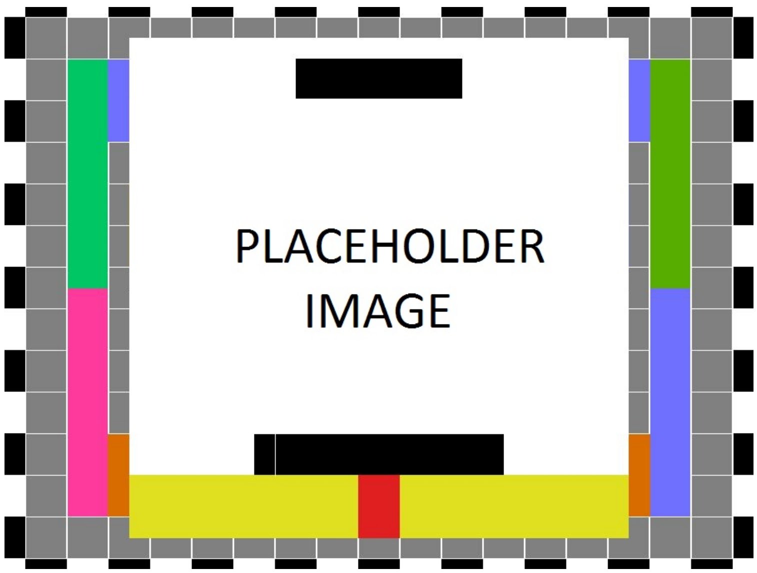
Department of Computer Science & Engineering
The University of Texas at Arlington

Project Charter
CSE 4316: Senior Design I
Spring 2021



Team Friendship
Product Name

Luke Brown
Grace Hopper
John Von Neumann
Ada Lovelace
Charles Babbage

Revision History

Revision	Date	Author(s)	Description
0.1	10.01.2020	GH	document creation
0.2	10.05.2020	AT, GH	complete draft
0.3	10.12.2020	AT, GH	release candidate 1
1.0	10.20.2020	AT, GH, CB	official release
1.1	10.31.2020	AL	added customer change requests

CONTENTS

List of Figures

1 PROBLEM STATEMENT

The problem statement defines the "Why" of the project. This is the higher purpose, or the reason for the project's existence. This section should avoid mentioning implementation details, and focus more on what the current problem is and what would be gained if the problem were to be solved. In short, the is the reason that you are going to be working on something, not the method(s) that you will be employing.

The reason we chose this project was because it was suggested to us by Chris Conly and everyone though it had something in it that interested them. It will help us learn about programming microcontrollers, handling communications between devices via different protocols, and web development.

2 METHODOLOGY

This is the "What" of the project and it states what will be done to address the problem statement. This section should focus mostly on what your solution is going to be and what it is going to do (i.e., we are going to build an app, robot, device, etc. to perform some task which mitigates the problem). If someone were to ask you "*What are you doing for your senior design project?*", this is section is basically what you would tell them.

Our goal is to be able to plug a recipe into a web application, that will tell the sensors, heat source, and pumps exactly how to produce the beer.

Our project will simplify the brewers job by heating and pumping the water to the containers through automation. This will ensure the water is kept at the desired temperature throughout the whole process and will help the brewer monitor all parts of the brewing process without requiring them to manually reheat/distribute the water. Automation will be done through the use of heat sensors, pumps, and microcontrollers that will communicate to a web-app/phone app. The brewer will be able to control the what temperatures the water will be set to and the amount of time they would like to spend on any step of the brewing process.

- put together sensors into something that can interface with a web server
- interface with the sensors through a web portal of some kind
- program the sensor to complete recipe specifications
- to control the heating coils to positively impact the beer
- the variables that concern
- wait and monitor, upon completion enjoy beer

3 VALUE PROPOSITION

The Value Proposition explains how the sponsors will benefit from your work, and why they should invest funding, time, and expertise in supporting your team. Here, you are essentially making a case for the project. There are many ways in which value can be returned to your stakeholders (industrial sponsors, instructors, the university, etc.), list any that may help you convince them to "buy in".

There is a non-zero chance Chris Conly will get beer. The project will help automate the process of homebrewing, saving time, and enabling it to be a more frequent activity. It would allow for a better end product that is more consistent than what a human could manage. It would enable for larger scale brewing operations due to the increased organization. Resume bolstering capabilties.

4 Development Milestones

This list of core project milestones should include all major documents, demonstration of major project features, and associated deadlines. Any date that has not yet been officially scheduled at the time of preparing this document may be listed by month.

Provide a list of milestones and completion dates in the following format:

- Project Charter first draft - Month Year
- System Requirements Specification - Month Year
- Getting all the materials to build the project
- Architectural Design Specification - Month Year
- Detailed Design Specification - Month Year
- Assembly of the hardware/sensors
- Demonstration of <feature or implementation milestone> - Month Year
- Demonstration of <feature or implementation milestone> - Month Year
- Demonstration of <feature or implementation milestone> - Month Year
- CoE Innovation Day poster presentation - Month Year
- Demonstration of <feature or implementation milestone> - Month Year
- Demonstration of <feature or implementation milestone> - Month Year
- Demonstration of <feature or implementation milestone> - Month Year
- first full beer brew completed
- Final Project Demonstration - Month Year
- Completion of the web app with successful communication to the sensors
- COMPLETION OF PROJECT AUGUST 2021

5 Background

A typical brewing day involves monitoring and keeping a hot liquor tank (HLT) at a desired temperature, then the brewer has to send the hot water from the HLT to the empty mash tun for pre-heating. After a set amount of time, the water from the mash tun will be sent back to the HLT to make sure the mash tun is at the desired temperature. While in the process of pre-heating, the HLT has to be monitored and will be continuously heated up to the desired temperature.

The next step is to start the mash by mixing in the grain with the hot water inside the mash tun. The water inside the mash tun must be kept at a specific temperature for a set amount of time. If the water drops below the desired temperature, a pump is turned on and the water is sent through a pump to a coil inside the HLT to heat the water.

Once the desired amount of time has passed, the HLT will be heated up to a higher temperature and then sent to the mash tun to rinse the sugars. At the same time, beer inside the mash tun will be pumped into the boil kettle. After all the beer has been sent to the boil kettle, the brewer must boil the beer and set a timer to add hops to the beer after each cycle. Once the boil is finished, the beer inside the boil kettle will be sent through a chiller to the fermenter.

The process of brewing without automation requires constant monitoring and heating of the temperature of the liquids inside the brewing kettles. Without constant supervision during the brewing process, it would be easy for a beginner to make mistakes and ruin the batch. The main idea of the our project is to simplify the brewing process by monitoring and controlling the temperature of the liquids inside of the different kettles. While also controlling where the liquids will go during all steps of the brewing process.

6 Related Work

Discuss the state-of-the-art with respect to your product. What solutions currently exist, and in what form (academic research, enthusiast prototype, commercially available, etc.)? Include references and citations as necessary using the *cite* command, like this [5]. If there are existing solutions, why won't they work for your customer (too expensive, not fast enough, not reliable enough, etc.). This section should occupy 1/2 - 1 full page, and should include at least 5 references to related work. All references should be added to the *.bib* file, fully documented in IEEE format, and should appear in the *references* section at the end of this document (the IEEE citation style will automatically be applied if your reference is properly added to the *.bib* file).

ProTip: Consider using a citation manager such as Mendeley, Zotero, or EndNote to generate your *.bib* file and maintain documentation references throughout the life cycle of the project.

There are a few options available, but it depends on how much you'd like to pay. On the more budget end, the BrewPI is a DIY kit that you could put together. You have to buy all the parts seperately, but they are all available off of one website. The BrewPI Spark is capable of controlling the temperature of a "hacked" fridge, the temperature of the heater inside of the fridge, and it monitors the temperature of the beer itself. The BrewPI Spark then communicates with another Raspberry PI that runs a web server that can be viewed on a computer or phone. It does not have mash temperature control or a way to control pumps or valves. The system has commercially available parts, but there is a very large do-it-yourself component to this project. All the parts can be purchased for around nine hundred dollars. [4]

The Pico Pro is an "all in one" brewing machine. You don't need to purchase pieces of a kit and put it together yourself. The brewing capacity is 1.75 gallons. It handles the heating of liquids and transfers the contents of the machine into a small keg via a pump. It doesn't offer any special capabilities other than being able to scan proprietary PicoBrew recipes. Setup is very minimal, there is not a lot of components that need to be put together. The product is commercially available for five hundred and fifty dollars. [1]

The Grainfather G30 uses a single chamber for all of it's brewing except when it chills the wort in the wort chiller. The brewing capacity of the Grainfather is a little over 6 gallons. The grain father is all electric and is intended to be used mainly indoors. [2] It has a phone application that is used to control heat and power. It has a pump for transferring liquids between the main vessel and the wort chiller. There is minimal set up and the product can be purchased for one thousand dollars. [3]

The Brewie is one of the more expensive options. It has two very large chambers for brewing. It is capable of mashing, sparging, hopping, cooling, draining, and fermentation. It has built in weight sensors, pressure sensors, self cleaning capabilities, and a built in cooler for wort chilling. It has a brewing capacity of up to 5.28 gallons. Set up is minimal and the product can be purchased for one thousand five hundred dollars. [6]

The closest competitor to our product is the BrewPI. The big disadvantage that it has is that you need to buy each individual part from BrewPI, then put it together yourself. There could arise an issue with sourcing so many different parts from a single small scale business. We want to offer a similar product, but with significantly less reliance on doing it yourself. Our team is positive we can produce a product that costs well under the full price of the BrewPI.

7 System Overview

Our solution to automating several parts of the brewing system will require us to monitor and control the temperatures of the liquids inside of the HLT and the mash tun to the desired temperature that the brewer sets. Along with controlling the temperatures of the liquids, we will also control when liquid will be passed through the different kettles using pumps.

The main workhorse of the project will be the microcontrollers. We've chosen the ESP32 to be the main microcontroller in the project. The ESP32 will control heating the water in the hot liquor tank, which will pump the hot water from the HLT to the mashing tun for preheating. The microcontrollers will control the pumps and the heating elements as well.

Equipment:

- Four brewing kettles
- Three heat sensors
- Three pumps
- heating element
- heat exchanger coils
- microcontrollers/ESP32

8 Roles & Responsibilities

Who are the stakeholders of the project? Who will be the point of contact from the sponsor or customer side? Who are the team members, and what will be their areas of responsibility? Will your team maintain the product owner and scrum master for the whole project, or will that role change periodically? This section should occupy 1/2 - 1 full page.

The stakeholders are the members of this team, Chris Conly, and the CSE Department. The main point of contact will be our mentor Chris Conly.

Ju Sujan Marco Sunghwa

Hardware:
Who deals mainly with the microcontoller
Who handles the data and web server side of things
who handles the creation of the web app
Application:
Server Acquisition & Maintanence
Which web stack - LAMP
Web (react)
Conversion of react web app to android/ios Android
Brewmasters:

9 Cost Proposal

This section contains the approximate budget for the project, where that money will come from, and any other support. This text should be replaced with a discussion and justification of major expenses, but not the actual monetary amounts (that will go in the preliminary budget section below).

9.1 Preliminary Budget

Include a high level budget table for components, fabrication, software licensees, development hardware, etc. This should be in a tabular format broken up into appropriate line items.

9.2 Current & Pending Support

What are all of the funding sources for the project, and are there any potential funding sources that haven't been secured yet? List all funding sources (including the default funding amount provided by the CSE department) and their dollar amounts.

10 Facilities & Equipment

What lab space, testing grounds, makerspaces, etc. will you need to complete the project? Will you require any specific equipment, and if so, where will you get it (borrow, lease, purchase, outsource, already present in the lab, etc.). This section should occupy 1/2 page.

Since we are currently facing an unusual situation in the COVID-19 pandemic, we will be conducting lab experiments and building equipment at a place where not many people present. Facilities can be either home or UTA CSE Senior Design lab.

Facilities has to have enough space to work on equipment, a power supply, and a table with a flat surface. Requirements for the facilities will be either leased or already present in the lab.

Required materials for the equipment for this project are three containers; one for a hot liquor tank, the other for a mash tun, and another for a storage kettle. Three pumps will be used to push the liquid in tanks to other tanks when the environment in tanks meets a certain condition. A heater is used in the hot liquor tank to keep the water at a certain temperature. A coil gets water from the mash tun when the temperate of the water in the mash tun drops below the targeted temperature and heats up the water for a certain temperature. A thermometer reading water temperature in a hot liquor tank, a timer for brewing time, and relays controlling the heater, pumps, and reading temperature sensor will be purchased or borrowed from UTA.

An application and a website will be created to provide an interface between users and the brewing system. Online database, the domain name for the website will be purchased.

11 Assumptions

An assumption is a belief of what you assume to be true in the future. You make assumptions based on your knowledge, experience or the information available on hand. These are anticipated events or circumstances that are expected to occur during your project's life cycle.

Assumptions are supposed to be true but do not necessarily end up being true. Sometimes they may turn out to be false, which can affect your project significantly. They add risks to the project because they may or may not be true. For example, if you are working on an outdoor unmanned vehicle, are you assuming that testing space will be available when needed? Are you relying on an external team or contractor to provide a certain subsystem on time? If you are working at a customer facility or deploying on their computing infrastructure, are you assuming you will be granted physical access or network credentials?

This section should contain a list of at least 5 of the most critical assumptions related to your project. For example:

The following list contains critical assumptions related to the implementation and testing of the project.

- A suitable outdoor testing location will be available by the 3rd sprint cycle
- The X sensing system developed by Sensor Consulting Company will be delivered according to specifications by the 4th sprint cycle
- Access to the customer installation site will be provided by the 5th sprint cycle
- The customer will provide ample power and network connectivity at the installation site
- The installation site network infrastructure will allow TCP network traffic on port 8080

12 Constraints

Constraints are limitations imposed on the project, such as the limitation of cost, schedule, or resources, and you have to work within the boundaries restricted by these constraints. All projects have constraints, which are defined and identified at the beginning of the project.

Constraints are outside of your control. They are imposed upon you by your client, organization, government regulations, availability of resources, etc. Occasionally, identified constraints turn out to be false. This is often beneficial to the development team, since it removes items that could potentially affect progress.

This section should contain a list of at least 5 of the most critical constraints related to your project. For example:

The following list contains key constraints related to the implementation and testing of the project.

- Final prototype demonstration must be completed by May 1st, 20XX
- The customer will provide no more than two maintenance personnel to assist in on-site installation
- Customer installation site will only be accessible by development team during normal business hours
- Total development costs must not exceed $800
- All data obtained from customer site must be reviewed and approved for release by the Information Security Office prior to being copied to any internet connected storage medium
- Final prototype demonstration must be completed by August 1st, 2021

- The entire brewing process can not be shown at the Summer 2021 Senior Design Presentations because brewing takes a certain time to produce beer.
- Total development costs must not exceed $800
- Social distancing requires virtual meetings which lead to a limitation on a tasting of the final product from brewing equipment.
- Extra searching and studying time is required since a lack of hardware-related experiences.

13 Risks

This section should contain a list of at least 5 of the most critical risks related to your project. Additionally, the probability of occurrence, size of loss, and risk exposure should be listed. For size of loss, express units as the number of days by which the project schedule would be delayed. For risk exposure, multiply the size of loss by the probability of occurrence to obtain the exposure in days. For example:

The following high-level risk census contains identified project risks with the highest exposure. Mitigation strategies will be discussed in future planning sessions.

Risk description	**Probability**	**Loss (days)**	**Exposure (days)**
Availability of X sensor module due to contractor delay	0.50	20	10
Outdoor testing grounds are not available	0.20	14	2.8
Internet access not available at installation site	0.30	9	2.7
Delays in shipping from overseas vendors	0.10	20	2.0
Certification delays at compliance testing facility	0.15	10	1.5

Table 1: Overview of highest exposure project risks

14 Documentation & Reporting

14.1 Major Documentation Deliverables

14.1.1 Project Charter

The project charter will be maintained on a GitHub repository. Updates will be suggested by all members and approved by Luke for pushing to the main branch. The initial version will be delivered by March 19, 2021. At the end of every sprint the team will decide if updates to this document are needed, and who will be in charge of them. Since the document may be updated throughout the whole development cycle of the project, the final version could be delivered during the last sprint in the summer semester, but a "final" version will be delivered at the end of sprint 4 on May 3rd, 2021.

14.1.2 System Requirements Specification

Describe how this document will be maintained and updated (how often, under what circumstances, etc.). When will the initial version be delivered? When will the final version be delivered?

The System Requirements Specification (SRS) document will be maintained in an Excel document, shared through Google Drive to allow all members to collaborate on it. This document will be updated at the beginning and end of every sprint unless members decide that there are no updates needed. The initial version will be delivered on March 23rd, 2021, at the beginning of sprint 3. Due to the nature of SCRUM, this document will be constantly evolving, and a final version will not be ready until the team is close to the completion of the project. A somewhat final version will be delivered at the end of sprint 4 on May 3rd, 2021.

14.1.3 Architectural Design Specification

The architecural design specification (ADS) will be maintained throughout the development cycle to match the SRS. The initial version will be delivered on April 14, 2021. Any revisions made between then and the end of sprint 4 will result with the delivery of the final version on May 3rd, 2021.This document will be maintained in a GitHub repository where the whole team can see and make updates as needed.

14.1.4 Detailed Design Specification

The detailed design specification (DDS) document will be developed and maintainedalong with the ADS due to their similarities, with the goal of delivering the initial and final at the same time as the ADS. This document will be maintained in a GitHub repository where the whole team can see and make updates as needed.

14.2 Recurring Sprint Items

14.2.1 Product Backlog

How will items be added to the product backlog from the SRS? How will these items be prioritized? Who makes the decision (product owner, group vote, etc.)? What software will be used to maintain and share the product backlog with team members and stakeholders?

The product backlog will be created once the SRS initial version is complete. The team will work during sprint 3 to turn all the requirements from the SRS into products in the product backlog. The products will be listed in a priority order, decided by the team. This priority order may be changed at the end of every sprint if needed. This document will be maintained on a Google Drive file where the whole team can see and make updates as needed.

14.2.2 Sprint Planning

The team will meet at the beginning of each sprint to determine the sprint goals, and members will pick products from the product backlog to complete during the sprint. There will be a total of about 7 sprints, 4 in the spring and 3 in the summer. Sprints will be around 2 weeks in lenght, unless the team decides it would be beneficial to change the length.

14.2.3 Sprint Goal

The sprint goal will be decided by the team in the sprint planning meeting. This goal will set the theme for what the team will accomplish individually and as a team. Professor Conly will aid in the sprint goal if needed to ensure we are on track with the project goals. Each item will have an expected amount of effort needed, which can be updated at any time.

14.2.4 Sprint Backlog

During the sprint planning, the team will decide which items should be completed during the sprint, and put them on a Google Drive document. The items chosen will be decided based on the sprint goal.

14.2.5 Task Breakdown

Each team member will pick items from the product backlog at the sprint planning meeting. The rest of the team will give input on the priority and amount of items picked by each member to ensure enough progress is made, as well as not overwhelming people with too many items. The sprint backlog will be maintained and updated on Google Drive.

14.2.6 Sprint Burn Down Charts

The sprint burn down charts will be updated at the end of each sprint using the information from the sprint backlog. The graph will be created on Excel to easily access the data from the backlog. No specific person is assigned to this task, rather whoever is done with their tasks and would like to volunteer. We will try to rotate this responsability.

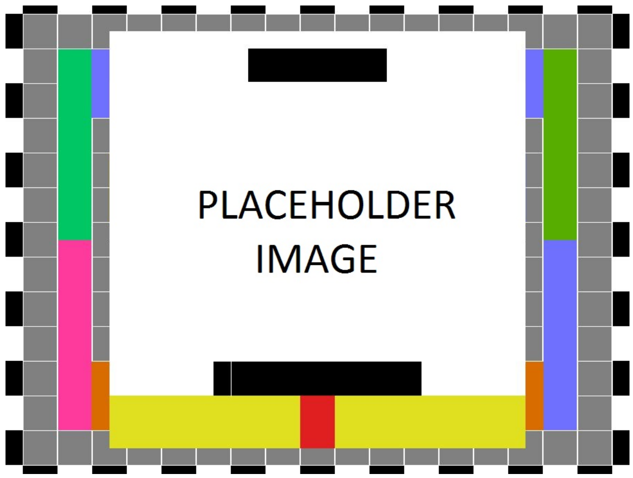


Figure 1: Example sprint burn down chart

14.2.7 Sprint Retrospective

At the end of every sprint, the team will meet and take some time to share what they think of the sprint system being implemented. Each team member will share what they think the team should start doing, what the team should stop doing, and what the team should continue to do. The conclusions will be applied on the following sprint.

14.2.8 Individual Status Reports

On the weekend landing in the middle of the sprint, each team member will give an update on their current progress for the sprint. The update will contain informatino such as what has been accomplished, what still needs to be accomplished, and any problems encountered that may delay progress. This update will be shared on Discord.

14.2.9 Engineering Notebooks

Each member will update their engineering notebook as many times as needed during each sprint, but at least one page per week. Due to the pandemic, each member can choose a family member, roommate, friend, or teammate to sign as a witness if they feel comfortable meeting in person. Pictures of each members progress can be shared on Discord to keep each other accountable.

14.3 Closeout Materials

14.3.1 System Prototype

What will be included in the final system prototype? How and when will this be demonstrated? Will there be a Prototype Acceptance Test (PAT) with your customer? Will anything be demonstrated off-site? If so, will there be a Field Acceptance Test (FAT)?

14.3.2 Project Poster

What will be included on the poster, what will be the final dimensions, and when will it be delivered?

14.3.3 Web Page

What will be included on the project web page? Will it be accessible to the public? When will this be delivered? Will it be updated throughout the project, or just provided at closeout (at a minimum, you need to provide a simple web page at the end).

The project web page will contain instructions to brew a patch, as well as information on the current patch.

14.3.4 Demo Video

The demo video will be us giving a short explanation of what the brewing process looks like using our project, including the setup, the brewing itself, the monitoring done, and the final product. The video will be short, around 5 minutes or less.

14.3.5 Source Code

How will your source code be maintained? What version control system will you adopt? Will source code be provided to the customer, or binaries only? If source code is provided, how will it be turned over to the customer? Will the project be open sourced to the general public? If so, what are the license terms (GNU, GPL, MIT, etc.). Where will the license terms be listed (in each source file, in a single readme file, etc.).

The code will be maintained on GitHub, available to the public. The licence terms will be listed in a readme file on the repository.

14.3.6 Source Code Documentation

What documentation standards will be employed? Will you use tools to generate the documentation (Doxygen, Javadocs, etc.). In what format will the final documentation be provided (PDF, browsable HTML, etc.)?

Doxygen will be used to generate documentation and the final documentation will be provided in PDF format.

14.3.7 Hardware Schematics

The hardware shematic will be provided once the team has receieved their equipment and have received proper training on how a brewing system should be set up.

14.3.8 Installation Scripts

How will the customer deploy software to new installations? Will you provide installation scripts, install programs, or any other tools to improve the process? Will there be multiple scripts provided (perhaps separate scripts for the graphical front end and back end server software)?

14.3.9 User Manual

A digital user manual will be provided.

References

[1] Pico Brew. Pico Pro. BrewPI, 2021. [Online]. Available: https://picobrew.com/store/products/pico.

[2] GrainFather Brewing. All grain brewing equipment. BrewPI, 2020. [Online]. Available: https://grainfather.com/all-grain-brewing-equipment.

[3] Grainfather Brewing Company. Grainfather g30, 2019. [Online]. Available: https://grainfather.com/grainfather-g30.

[4] Elco Jacobs. BrewPI Spark 3. BrewPI, 2017. [Online]. Available: https://www.brewpi.com/now-available-brewpi-spark-3.

[5] Kenneth S Rubin. *Essential Scrum: A Practical Guide to the Most Popular Agile Process*. Addison-Wesley Professional, 1st edition, 2012.

[6] MoreBeer Staff. Brewie fully automated brewing system, 2020. [Online]. Available: https://www.morebeer.com/products/brewie-fully-automated-brewing-system.html.