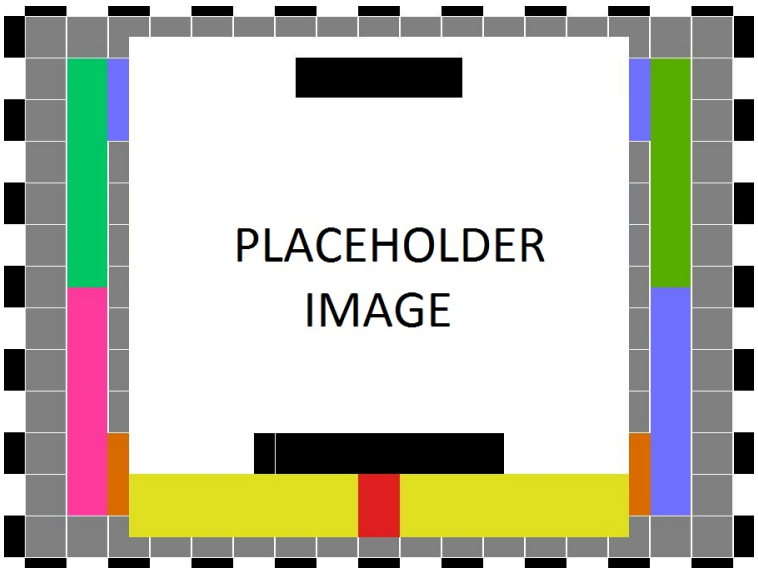
# Department of Computer Science & Engineering
# The University of Texas at Arlington

## 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

An in-depth explanation of the problem, including the "business case". What is wrong with the status-quo or what opportunity exists that justifies undertaking this project (expanding upon the problem statement)? If you have a clear customer or sponsor, why do they want you to work on this? What is the existing relationship, if any, between the development team and the customer? This section should occupy 1/2 - 1 full page.

The process of brewing can be something that is very easy to make mistakes on, espcially as a beginner. The idea of this project is to make it to where even the biggest laymen can accomplish a decent tasting beer.

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. Once all the beer has been sent to the boil kettle, the brewer has the 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.

## 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 [**?**]. 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.

BrewPI Spark - commercially available - 900

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.

PicoBrew Pico C - commercially available - 400

brewing capacity 1.3gal

Brewie - 1500

The brewie is one of the more expensive options. It is capable of mashing, sparging, hopping, cooling, drain, fermentation

citations: https://www.brewpi.com/ https://picobrew.com/store/products/picoc https://www.morebeer.com/proc fully-automated-brewing-system.html

## 7 System Overview

Explain, at a high level, how you will implement a solution to the problem. Include a diagram of major components to the system (not a full architectural design, but a high level overview of the major system components and how a user or external system might interface). Avoid specific implementation details (operating system, programming languages, etc.). This section should occupy at least 1 full page.

Equipment; giant pot thermometer pump heating element

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, that heated water will be pumped from the HLT to the mash tun o

Microcontroller controls pump

Microcontroller controls heat

when water reaches desired temp it will be sent to the mash tun via pumps to preheat the mash tun

when the mash tun is pre heated the water will go back to the HLT

## 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.

## 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

# 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

These deliverables are major grade components of the course. Completing these documents should generally be the sprint goal during the applicable sprint period. Refer to current and previous course syllabi and schedules to estimate the due dates of these items. Remove this explanatory paragraph from your draft, but leave the heading.

### 14.1.1 Project Charter

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?

### 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?

### 14.1.3 Architectural Design 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?

### 14.1.4 Detailed Design 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?

## 14.2 Recurring Sprint Items

The following items will be documented and maintained during each individual sprint. As above, remove this paragraph from your draft, but leave the heading.

### 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?

### 14.2.2 Sprint Planning

How will each sprint plan be planned? How many sprints will there be (you need to look at the schedules for this course and previous Senior Design II courses during the appropriate semesters to figure this out).

### 14.2.3 Sprint Goal

Who decides the sprint goal? How will you involve your customer in this process?

### 14.2.4 Sprint Backlog

Who decides which product backlog items make their way into the sprint backlog? How will the backlog be maintained (collaboration software, a "scrum board", etc.)?

### 14.2.5 Task Breakdown

How will individual tasks be assigned from the sprint backlog? Will it be up to each team member to voluntarily claim a task, or will it come from the product owner? How will time spent on tasks be documented?

### 14.2.6 Sprint Burn Down Charts

Who will be responsible for generating the burn down charts for each sprint? How will they be able to access the total amount of effort expended by each individual team member? What format will the burn down chart use (include an example burn down chart below).

### 14.2.7 Sprint Retrospective

How will the sprint retrospective be handled as a team? When will this discussion happen after each sprint? What will be documented as a group and as individuals, and when will it be due?

### 14.2.8 Individual Status Reports

What sort of status will be reported by each individual member, and how often will it be reported? What key items will be contained in the report?

### 14.2.9 Engineering Notebooks

How often will the engineering notebook be updated, at a minimum, by each team member? What is the minimum amount of pages that will be completed for each interval, and how long will that interval

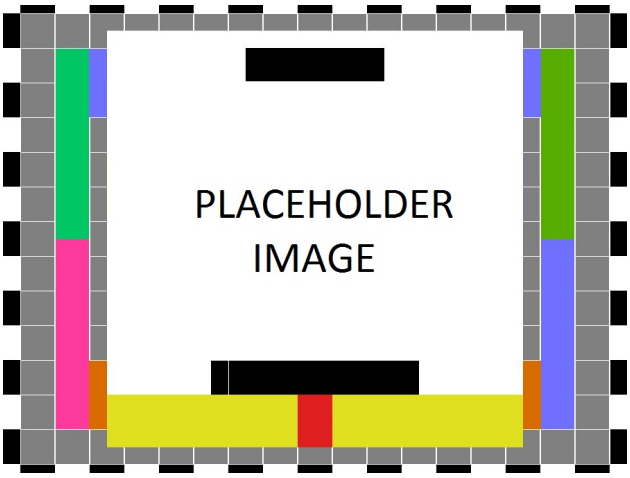



Figure 1: Example sprint burn down chart

be? How will the team keep each member accountable? Who will sign of as a "witness" for each ENB page?

## 14.3 Closeout Materials

The following materials, in addition to major documentation deliverables, will be provided to the customer upon project closeout. Remove this paragraph from your draft, but leave the heading.

### 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).

### 14.3.4 Demo Video

What will be shown in the demo video(s)? Will you include a B-reel footage for future video cuts? Approximately how long will the video(s) be, and what topics will be covered?

### 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.).

### 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.)?

### 14.3.7 Hardware Schematics

Will you be creating printed circuit boards (PCBs) or wiring components together? If so, list each applicable schematic and what sort of data it will contain (PCB layout, wiring diagram, etc.). If your project is purely software, omit this section.

### 14.3.8 CAD files

Will the project involve any mechanical design, such as 3D printed or laser-cut parts? If so, what software will you use to generate the files and what file formats will you provide in your closeout materials (STL, STEP, OBJ, etc.). If your project is purely software, omit this section.

### 14.3.9 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)?

### 14.3.10 User Manual

Will you customer need a printed or digital user manual? Will they need a setup video? Decide now what will be provided and discuss.