

## **Letter of Transmittal**

Feb 13, 2022

Dr. Mike Hegedus  
School of Engineering Science  
Simon Fraser University  
8888 University Dr.  
Burnaby, BC V5A 1S6

Re: ENSC 405W/440 Requirements Specification

Dear Dr. Hegedus and teaching assistants,

The attached document is the requirements specification for our project, LocalHost. Our primary goal is to create a system that helps waiting staff provide more efficient service to restaurants by providing real time information of table statuses.

Our product consists of two main components: the hardware device that is mounted by each restaurant table and the software interface that the waiter will be using. The mounted device located at each restaurant table will provide information to the software that the waiters will be using.

The following requirements document will outline all the required features and functionality our product should have. We will start by covering our system overview and how it works. Then, this document will cover requirements such as general functional requirements, electrical requirements and software requirements. Finally, this document will go over the safety and sustainability, including the engineering standards we would like to meet.

Thank you for taking the time to review our requirements specification document. Feel free to contact us if there are any questions and concerns.

Sincerely,

Members of LocalHost Services Inc.



LocalHost  
Services Inc.

# Requirements Specification:

## LocalHost

**Partners:**

Angus Kan  
Irene Leung  
Kevin Cao  
Patrick Cong  
Rico Chao  
Yoel Yonata

**Contact:**

Kevin Cao  
hca119@sfu.ca

**Submitted To:**

Dr. Mike Hegedus, ENSC 405W  
School of Engineering Science  
Simon Fraser University

**Issue Date:**

Feb 13, 2022

# Abstract

This document will go over all the requirements of our project, LocalHost. LocalHost is a restaurant management system that provides waiters real-time information of the tables that they are serving. We will start by covering the overview of the functionality of our system and breaking down all the different components. We will then go into each individual component and specify the requirements. Requirements such as electronic requirements and software requirements will be covered in this document. This document will also specify the safety and engineering standards that need to be met. This requirement specification document aims to the purpose of our product, how the system works and what requirements we will meet with our product.

LocalHost consists of two main components, the buzzer that is located at each table and the software that the waiter uses to obtain table information. The buzzer device also contains a camera that will allow us to detect and retrieve information for the waiter's software interface in order to automate the given information. This device will have a button at the center of it to allow the customers to buzz the waiters.

The three main requirements in our product will be: general requirements, software requirements and hardware requirements. General requirements will include the overall functional requirements of our system. The software requirements will go over performance requirements such as image detection accuracy and functional requirements of what the software needs to be able to perform. Lastly, the hardware requirements will cover requirements such as power and output requirements of the device.

This document will then cover the safety, sustainability standards of our product. This document will finally conclude with the planned delivery time of our prototype.

## List of Figures

<b>Figure 2.1</b> - System overview	9
-------------------------------------	---

## List of Tables

<b>Table 3.1</b> - General Requirements	12
<b>Table 3.2</b> - Software Requirements	13
<b>Table 3.3</b> - Hardware Requirements	14
<b>Table 4.1</b> - Safety Requirements	15
<b>Table 4.2</b> - Sustainability Requirements	16
<b>Table 5.1</b> - Engineering standards	17
<b>Table 7.1</b> - Proof-of-Concept Test Plan	19

## Glossary

AI	Artificial Intelligence
GPIO	General Purpose Input/Output
GPU	Graphics processing unit
I/O	Input/Output

# Table of Contents

Abstract	<b>3</b>
List of Figures	<b>4</b>
List of Tables	<b>4</b>
Glossary	<b>4</b>
Introduction	<b>6</b>
1.1 Background	6
1.2 Purpose	6
1.3 Scope	6
1.4 Product overview	7
1.4.1 Product functions	7
1.4.2 User characteristics	7
1.4.3 Limitations	8
1.4.4 Assumptions and dependencies	8
System Overview	<b>9</b>
System Requirements	<b>11</b>
3.1 General Requirements	11
3.2 Software Requirements	12
3.3 Hardware Requirements	13
Safety and Sustainability	<b>15</b>
4.1 Safety	15
4.2 Sustainability	15
Engineering Standards	<b>17</b>
Conclusion	<b>18</b>
Appendix	<b>19</b>
References	<b>20</b>

# 1. Introduction

## 1.1 Background

Currently, many restaurants are short on manpower because restaurant owners are trying to cut down the cost of running the business, which potentially compromises the quality of service and also the efficiency, especially during peak dining hours. It is difficult for waiters to track which table's customers need attention, and also the waiters need to go back and forth to fulfill customer's inquiries which adds extra stress to their workload: they need to go to the customer first to acknowledge what they need, go get the requested item, and then go back to the customers again. Furthermore, sometimes waiters can lose track of which tables need to be cleaned and prepared, while the next group of customers has to wait extra long outside.

Currently, there are some products on the market such as electrical buzz/bell which can inform the waiters which tables need assistance, but due to their limited features, they cannot fully optimize the use of manpower, or manage the customer traffic. LocalHost aims to help restaurants maximize their efficiency and also their profit, by providing them with a fully automatic system, which uses image processing technology to keep their staff updated on what actions need to be taken at which table.

## 1.2 Purpose

The purpose of this product is to provide restaurants with a device mounted on the wall for each table, along with a management software application to track the table statuses and respond to customers' needs more efficiently using the help of image processing technology and Artificial Intelligent (AI) systems. Our product provides this solution to help the restaurants improve their service and increase their profit, while reducing the stress of their staff and the business cost for the business owners.

## 1.3 Scope

The intention of this document is to outline the individual requirements of LocalHost, as well as the safety and sustainability requirements we will comply with. The requirements will be broken down into three sections: System, Software, and Hardware. Each individual requirement will also be categorized as "proof of concept", "prototype", or "final product".

## 1.4 Product overview

### 1.4.1 Product functions

The goal of LocalHost is to provide a fast, convenient, and intuitive way to track the status of tables and the needs of customers. These features best appeal to restaurant owners and waiters that are looking to increase the overall efficiency of their restaurant.

LocalHost is looking to develop a product which contains one hardware device per table, which is then matched with a central management software overlooking all hardware devices. The product will automatically identify the dining table's current status efficiently and accurately. Its functions can help workers in restaurants such as: table ready for the next customer, needs water refill, and needs to be cleaned. The hardware device includes an ESP32-CAM module to capture videos and photos as input, 10 LED lights to display different colors according to the table's current status, and a button to manually alert a waiter. The management software will be able to analyze the videos and photos and send feedback to the device to manage the LED light colors, and most importantly, display notifications to inform the software user what actions need to be taken.

From the moment of the video/photo of the table being captured, the product should be able to notify the user within a 15 seconds delay if there's any actions that need to be taken. The product should be easy to use. During the testing phase, 90% of the test participants should complete 80% of the tasks without intervention.

### 1.4.2 User characteristics

This product is suitable for users with all educational levels and technical expertise. Some basic experience of using computers may apply, including opening the software, registering an account and logging in, and navigating among different app frames. All users who satisfy the qualifications above should be able to operate the product without any difficulties. Documentation and user manual will be included to guide users to use the product properly.

### 1.4.3 Limitations

- The hardware device has to be installed on hard surface close to the tables
- The captured videos/photos cannot be saved due to privacy policies
- The camera must have sufficient and stable power supply
- The user's computer must have strong GPU in order to keep the delay manageable
- All customer's information may never be shared with any third party

The product must abide by the local laws that protect the safety and privacy of our end users, the product's development may also be limited by the budget and the deadline.

### 1.4.4 Assumptions and dependencies

- The system should use Artificial Intelligence (AI) and difference comparing for image processing
- The software should be able to analyze the data input from the camera, generate and display the result on the computer screen
- The software of this product should be available on Windows systems
- All hardware components should support 110V power supply
- The camera should be able to send data to the computer software
- The LED lights should be able to respond to the microcontroller and display colors accordingly



## 2. System Overview

In this section a high-level overview of the product system will be provided. Figure 2.1 is a diagram which demonstrates our product system in both hardware and software components.

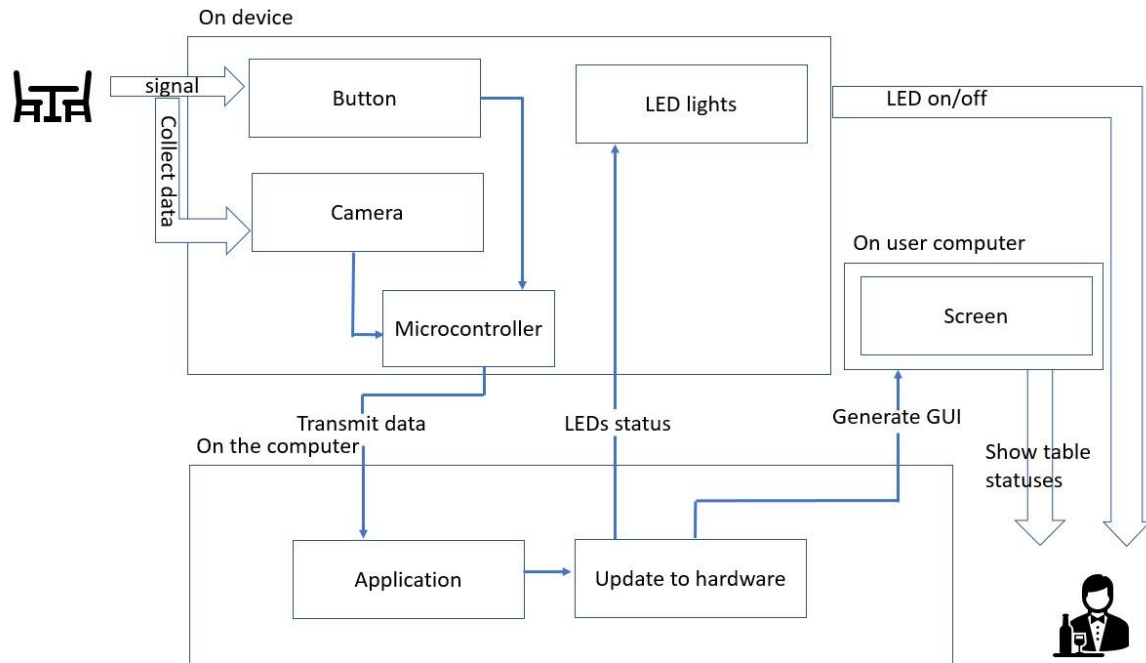


Figure 2.1 System overview

LocalHost is a semi-autonomous restaurant management system dedicated to help improve the experience of both waiters and customers while the customers are dining in. The device will have features that allow customers to gain attention from the waiters, automatically detecting customer's water level, pinging waiters to clean the table, and updating the status of the table.

Both the button and the camera will be used to obtain the information to determine the statuses of the table. Once the product is mounted near the table, it will start to collect and transmit required data to the microcontroller after turning on.

A microcontroller will be programmed to perform both data analyzing and I/O signal transmitting tasks through Wi-Fi. The camera, the button, and the LED lights will be connected to the microcontroller in order to process all the information, and output the correct table state as desired. Correct table state is then relayed to the waiter's software and the output LED lights in the device.

A computer screen will provide the waiters a visual display of the status of each table, to reduce customer waiting time and increase the rate of response to each table. There are in total of four table statuses:

- Occupied but need attention
- Occupied
- Need to clean
- Available

For the table status of "Occupied but need attention", the waiter can tap on the table with the status to get more detail. For example, if the table ran out of water, the "occupied but need attention" status will be triggered and the waiter will tap on the table to see a detailed message saying "Table may need refill of water".

# 3. System Requirements

The requirements and scope for our product, LocalHost, will be listed in the following section. These requirements will provide all the functionalities and deliverables of our product, as well as being a good reference for the scope of our project. Requirements will be split into three sections: General requirements, Software requirements, and Hardware requirements. There will be three project stages: proof-of-concept stage, prototyping stage, and production stage; the requirements will be organized into these three project stages.

The following format will be used as part of the labeling for our requirements:

## [Req R.S.T-stage]

'R' represents the number for the requirements section, 'S' represents the number for the sub-section of the requirements section, 'T' represents the requirement number, and 'stage' represents the specific project stage where the requirement should be met. For the three project stages, letters 'a' would be used to represent proof-of-concept stage, letter 'b' would be used to represent prototyping stage, and letter 'c' would be used for production stage.

## 3.1 General Requirements

[Req 3.1.1-c]	Multiple devices must be able to communicate with the management software at all times
[Req 3.1.2-b]	Device must be mounted such that the table is in view of the camera
[Req 3.1.3-a]	Device camera must not save footages, but monitor and provide feedback
[Req 3.1.4-b]	Device must be able to differentiate different table statuses and update accordingly
[Req 3.1.5-c]	Device must work during restaurant operating hours
[Req 3.1.6-b]	Device must contain a button, camera, and LED lights
[Req 3.1.7-a]	Management system must communicate with devices
[Req 3.1.8-b]	Management system must be able to change different table

	statuses
[Req 3.1.9-a]	Device camera must be working
[Req 3.1.10-b]	Device must have a working button
[Req 3.1.11-b]	Device must contain working LED lights

**Table 3.1 - General Requirements**

### 3.2 Software Requirements

The software will be utilizing OpenCV, which is an optimized library for facial recognition, image detection and image processing. The prototype will detect various states of the table, then send flags based on the table state to the management software.

To meet this requirement, we will be writing our program for the microcontroller using the Arduino IDE, as well as writing in Python to support the OpenCV library.

Connectivity will be achieved with 802.11b/g/n Wi-Fi, which has a wide connectivity range. Flags will be passed back and forth between the management software and the microcontroller to alert the waiters of any table status change.

For the proof of concept stage, we will demo table status updates by using the camera on the module and the output LEDs.

[Req 3.2.1-a]	The software must be able to identify the water level in a cup with a minimum of 80% accuracy
[Req 3.2.2-a]	The software must be able to identify if a plate has no more food with a minimum of 80% accuracy
[Req 3.2.3-a]	The software must be able to differentiate between a clean and dirty table with a minimum of 90% accuracy
[Req 3.2.4-a]	The software will use the OpenCV library
[Req 3.2.5-b]	The software must connect the management software and the microcontrollers using 802.11b/g/n Wi-Fi
[Req 3.2.6-b]	The Wi-Fi range must be at least 20 meters

[Req 3.2.7-b]	The software must send the correct table state flag back to the management software
[Req 3.2.8-a]	The software must be able to process the image provided by the camera
[Req 3.2.9-a]	The software must be able to handle data in real-time
[Req 3.2.10-c]	Management software must be able to create layout of restaurant
[Req 3.2.11-c]	Management software must display the correct table states

**Table 3.2** - Software Requirements

### 3.3 Hardware Requirements

The device powering our project will be the ESP32-CAM Module. This device was chosen for its small form factor as well as a port for a camera attachment. There will be enough GPIO ports to add additional features to our product. As a device that will be mounted on the wall, we must keep the weight to a minimum to prevent the risk of the device falling.

For the proof of concept stage, we will be using the camera module to ensure that the image detection is working.

[Req 3.3.1-b]	Device must have circuit protection for sudden surge of current or change in voltage
[Req 3.3.2-b]	Device must have thermal management to prevent the microcontroller from overheating
[Req 3.3.3-b]	Device must include a physical power button/switch
[Req 3.3.4-b]	Device must have 1 LED indicating power status
[Req 3.3.5-b]	Device must have a ring of LEDs to indicate table status
[Req 3.3.6-b]	Device must have a physical button for customer use

[Req 3.3.7-b]	Device must be enclosed in a hockey puck shape
[Req 3.3.8-a]	Device must include the ESP32-CAM module for image processing
[Req 3.3.9-b]	Microcontroller requires between 3.3V to 5V supplied by a power adapter
[Req 3.3.10-b]	Microcontroller requires no more than 500mA of current to operate
[Req 3.3.11-b]	Must be connected to power at all times
[Req 3.3.12-c]	Printed Circuit Board must be EMC compliant

**Table 3.3 - Hardware Requirements**

## 4. Safety and Sustainability

### 4.1 Safety

LocalHost always puts our customer's safety first. Our product will follow the guideline of engineering safety standards as our requirements. It ensures that our users, servers in restaurants, of our product will avoid any harm to them.

[Req 4.1.1-c]	Devices on the wall must not fall down once it is installed correctly.
[Req 4.1.2-c]	Videos and pictures captured must not be used for any other purpose.
[Req 4.1.3-c]	Devices must not have open ended wires when users are using it.
[Req 4.1.4-c]	User manual must be able to guide users to install the device correctly.
[Req 4.1.5-c]	Brightness of LED light from the device must not hurt the eyes of users and customers in restaurants.
[Req 4.1.6-c]	Temperature for devices in use must not exceed 50°C.
[Req 4.1.7-c]	Voltage for operating the microcontroller must not exceed 5V.

**Table 4.1** - Safety Requirement

### 4.2 Sustainability

As our technology is developing rapidly, sustainability of a product has become more important to consumers. For a good product, it should not only have the best technology, but also does not do any harm to our environment. Innovation of technology has improved our living quality greatly, but it is also damaging a lot of natural resources on Earth. To address the environmental concerns and sustainability of our product, LocalHost will follow the guideline of engineering standards by method of sustainable packaging. <sup>[1]</sup>

LocalHost will use biodegradable materials like recycled cardboard and paper to package our product. <sup>[2]</sup>

[Req 4.2.1-c]	Packaging of the device must use recyclable materials.
[Req 4.2.2-c]	The whole packaging of the device should be made at least 60% of recyclable materials.

**Table 4.2 - Sustainability Requirements**



## 5. Engineering Standards

The following table contains the engineering standards that LocalHost should be aiming to meet.

<b>ISO/DIS 31700</b>	Consumer protection — Privacy by design for consumer goods and services <sup>[3]</sup>
<b>CSA C22.1:21</b>	Canadian Electrical Code, Part I (25th Edition), Safety Standard for Electrical Installations <sup>[4]</sup>
<b>CSA ISO/IEC 30137-1:20</b>	Information technology — Use of biometrics in video surveillance systems — Part 1: System design and specification <sup>[5]</sup>
<b>CAN/CSA-C22.2 NO. 60950-1-07 (R2021)</b>	Information Technology Equipment — Safety — Part 1: General Requirements (Bi-National standard, with UL 60950-1) <sup>[6]</sup>
<b>CAN/ULC-S316-14 (R2019)</b>	Standard For Performance Of Video Surveillance Systems <sup>[7]</sup>
<b>ISO 21600:2019</b>	Technical product documentation (TPD)— General requirements of mechanical product digital manuals <sup>[8]</sup>
<b>ISO/TC 122/SC 4</b>	Packaging and the environment <sup>[9]</sup>

**Table 5.1** - Engineering standards

## 6. Conclusion

LocalHost aims to provide an innovative solution to restaurants by incorporating image processing paired with our management software to keep track of table statuses and respond to customer needs. Although there are existing restaurant software management systems on the market, our product aims to combine the use of hardware and software, as well as the use of AI to further improve restaurant management. Our solution aims to help waiters reduce their amount of work through providing real time updates of the table statuses, and help restaurant owners maximize their profits through increasing table turnover rate and decreasing labor cost.

This requirement specification document has included an overview of Localhost from hardware to software aspects, a high-level system overview, and description of the product's general functionality. It also has guidelines of engineering standards along with safety and sustainability requirements in order to make sure LocalHost products are user friendly.

The document also describes the functional requirement specifications of Localhost required to meet with a workable prototype by April 2022.

## 7. Appendix

**Table 7.1** outlines the details of LocalHost’s functionalities that will be presented for the proof-of-concept presentation. The following key problems or features are chosen because image processing with the camera for our device is one of the main features. The next stages require image processing with the camera and our software to work in order to demonstrate that our system and product work fully and reliably.

#	Key Problems/Functions	Description	Expected Results
1	Checking if the cup is empty using image processing algorithm	Detect water level in cups with at least 80% accuracy	If water level is below a certain height, software should display a notification saying “table # may need a refill”
2	Checking if the plate is empty using image processing algorithm	Detect if there’s food in the plate with at least 80% accuracy	If the plate is empty, software should display a notification saying “plates on table # may be taken away”
3	Table status display	Show the table is available/needs to be cleaned/occupied/need attention	Correct status should be displayed on the hardware device and also on the software
4	Camera must not save any footage	The camera should be functional, for data analyzing purpose only	The camera should be working and sending data to the software for image processing purposes only, no data should be saved or uploaded.
5	Software using OpenCV	Our software will be using the OpenCV library	Software should be working properly with the use of OpenCV library for image processing

**Table 7.1** - Proof-of-Concept Test Plan

## 8. References

- [1] “What makes a product sustainable?,” *Earth911*, 02-Aug-2020. [Online]. Available: <https://earth911.com/business-policy/what-makes-a-product-sustainable/#:~:text=For%20a%20product%20to%20be,unlikely%20to%20be%20considered%20sustainable.> [Accessed: 13-Feb-2022].
- [2] A. Krososky, “What are the best sustainable packaging materials?,” *Green Matters*, 21-Jan-2021. [Online]. Available: <https://www.greenmatters.com/p/sustainable-packaging-materials>. [Accessed: 13-Feb-2022].
- [3] “ISO/DIS 31700,” *ISO*, 06-Jan-2022. [Online]. Available: <https://www.iso.org/standard/76772.html>. [Accessed: 13-Feb-2022].
- [4] “Product,” *CSA Group*, 29-Aug-2021. [Online]. Available: <https://www.csagroup.org/store/product/CSA%20C22.1:21/>. [Accessed: 13-Feb-2022].
- [5] “CSA ISO/IEC 30137-1:20,” *Standards Council of Canada - Conseil canadien des normes*. [Online]. Available: <https://www.scc.ca/en/standardsdb/standards/30642>. [Accessed: 13-Feb-2022].
- [6] “Product,” *CSA Group*, 29-Aug-2021. [Online]. Available: <https://www.csagroup.org/store/product/2418577/>. [Accessed: 13-Feb-2022].
- [7] “Can/ULC-S316-14 (R2019),” *Standards Council of Canada - Conseil canadien des normes*. [Online]. Available: <https://www.scc.ca/en/standardsdb/standards/27733>. [Accessed: 13-Feb-2022].
- [8] “ISO 21600:2019,” *ISO*, 22-Feb-2019. [Online]. Available: <https://www.iso.org/standard/71225.html>. [Accessed: 13-Feb-2022].
- [9] “ISO/TC 122/SC 4 - packaging and the environment,” *ISO*, 11-Feb-2022. [Online]. Available: <https://www.iso.org/committee/52082.html>. [Accessed: 13-Feb-2022].