**C458 Group Project Proposal**

Joseph Olin  
Michael Roark  
Branden Wagner

**BarBot17**

**Abstract**

**Introduction**

Attending customers at a counter is a repetitive task. First, the waiter/waitress must ask the customer if they desire a consumable. Then, the waiter must retrieve the consumable from where it is stored. Once the consumable is retrieved, the waiter makes a pass over the length of the counter, giving customers the order they desired. One notable model for attending customers can be found at Brazeiros in Louisville, Kentucky. At Brazeiros, customers are given a color-coded coaster, with one side being red and other side being green. Based on the color of the coaster the customer shows, the customer may or may not be continually served freshly cooked meat. The Bar bot our project proposes to implement will take the repetitive task of attending customers at a counter and make that task the responsibility of an intelligent robot rather than a human. The idea is that Bar bot will load itself via an arm attached to the top of the bot. Customers will each have a double-sided card with a certain color (e.g. red) on one side and a different color (e.g. green) on the other side. Similar to how other restaurants such as Brazeiros operate, one color will indicate the customer wants an order and the other color will indicate the customer is content and does not desire an order. The robot will make cycles through the counter, ignoring unoccupied spaces and customers who use the color-coded card to indicate they don’t desire an order. The Bar bot’s goal will be to deliver items only to customers who desire an order as quickly as possible via the attached arm. We will simulate the environment of a bar with a narrow platform on which we will place the delivery area.

**System Details**

**Environment**

There are 3 main parts of the environment: the storage/loading area, the bot traversal area, and the customer area.

The storage/loading area is to be at the end of the counter. There, a hopper or some similar storage container will contain the consumables (e.g. mints). The bot will approach the hopper at the end of a bar run and the hopper will be opened, allowing the consumables to empty into the storage basket on the bot.

The bot traversal area is the space on the counter where the bot moves. The space stretches from one end of the counter to the other end. The bot will move forward from the hopper along this traversal space to attend each customer and, when the bot reaches the end of the counter, it moves backward to the hopper to start another run.

The customer area is the area closer to the edge of the counter where customers will display their color-coded card to the bot, and where the bot will also place the customer’s order. To start with, the number of customers allowed at the counter will be static. However, as the project progresses, delivery memory will be implemented so the bot knows who all its delivered orders to in a run.

**Sensors**

The sensors likely to be needed for the bot to perform the aforementioned tasks are: two line follower sensors, an ultrasonic sensor, a color sensor, and possibly some pressure sensors for the arm (if none come with it).

The line follower sensors would be needed for 1) notifying the bot when it is near the edge of the counter and 2) possibly for notifying the bot when it has reached the loading area (where a piece of tape would mark where it needs to stop).

The ultrasonic sensor might be used in place of the second line follower sensor to notify the bot when it has reached the hopper.

The color sensor would be used to notify the bot what color card the customer is displaying, thus notifying the bot if it needs to give the customer an order.

The pressure sensors would be needed to notify the bot how much pressure the arm is using in gripping a consumable.

**Performance Matrices**

Two primary performance metrics we intend to use in this project are: the number of orders the bot is supposed to fulfill (based upon the number of green colored cards displayed by customers), and the time the bot takes in fulfilling those orders. For the first metric, a count can be taken on the number of green colored cards displayed on a run, which can then be compared with the number of consumables the bot handed out. For the second metric, a timer can be implemented either within the code of the bot or externally via a physical timer, which will track the time taken to make a delivery run.

**System Functionalities**

* Get bot to move forward along the counter
* Get bot to move backward along the counter
* Get bot to stop at loading area
* Get hopper to load bot
* Get arm to grab and hold on to a consumable
* Get arm to unload a consumable
* Get bot to recognize color cards
* Get bot to unload a consumable at a green card
* Get bot to pass a customer at a red card

**Timeline**

