Project Design: Smart Shelf

SYSC 3010: Computer Systems Development Project

Group: W10

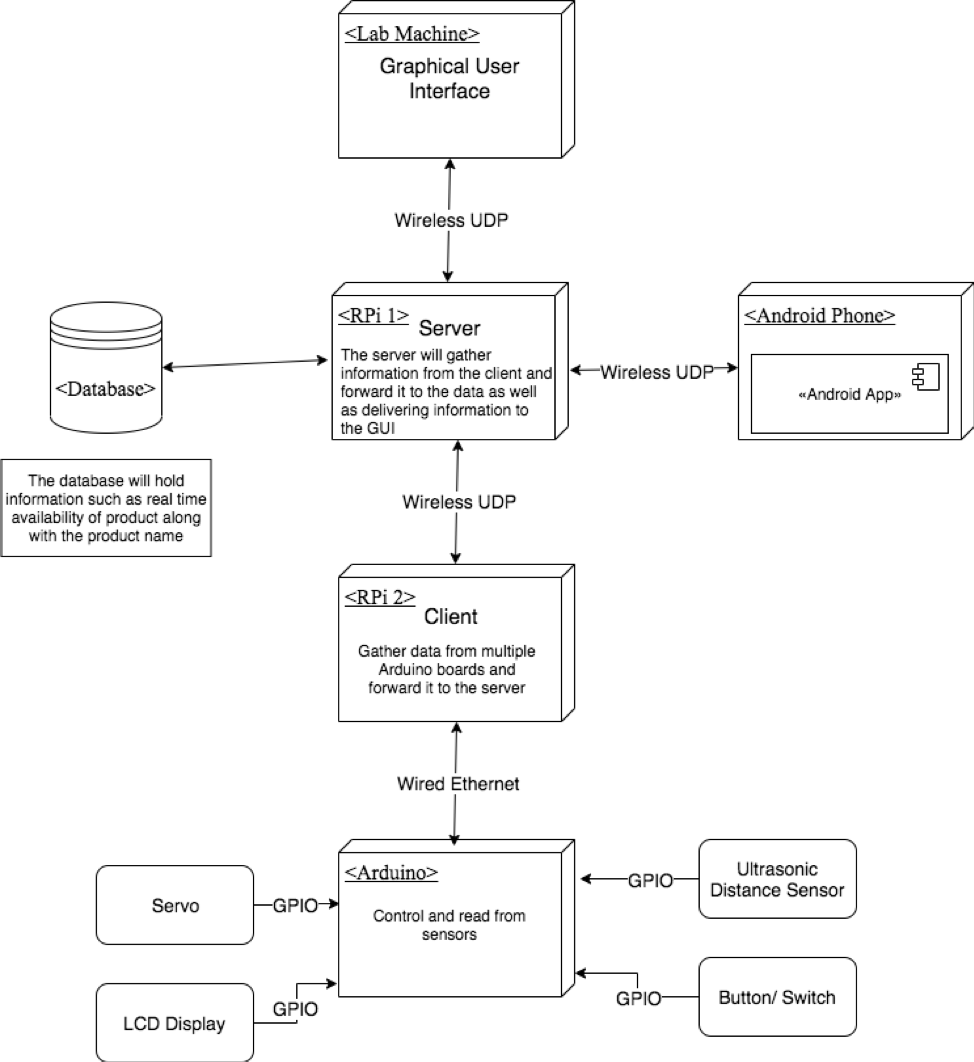
Members: Archit Bhatia, Ross Matthew, Marko Simic, David Casciano

**Problem Statement:**

Keeping shelves stocked with product is a logistics nightmare which can prove costly to retailers. The Smart Shelf detects and notifies when the shelf is empty and how many items are on the shelf. Additionally, the shelf will move all products to the front of the shelf providing better appeal to the customers. This is a common problem in shopping stores as employees don't have information on which shelf is empty and should be restocked. Our goal is to eliminate this complicated and time-consuming process and create a more efficient way for employees to work and customers to shop.

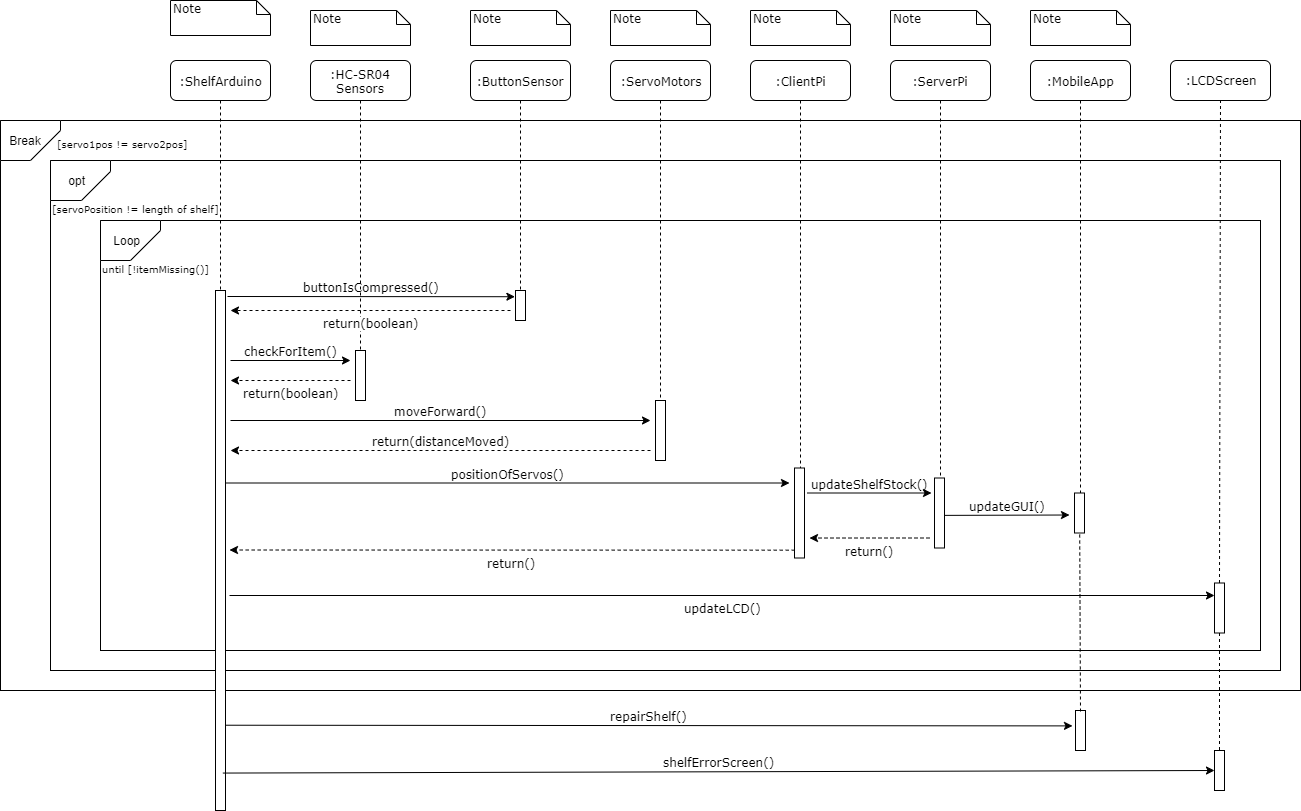
Three main constraints of the Smart Shelf are size, price, and material. Creating a shelf that can easily be integrated into existing stores creates a problem of fitting electronics into the Smart Shelf, as more sensors take up more space. Another constraint is price as it has to be financially feasible for store owners and large-scale business to invest in the Smart Shelf. Again, adding more sensors would prove to be costly. Therefore, aiming for a low cost would mean having to find low cost alternatives for motors and buttons. Lastly, the material of the product is constraint as it needs to be easily producible yet strong. Therefore, 3D printing would not be an option for many of the parts, as it needs to be stronger than the common filaments such as ABS (Acrylonitrile Butadiene Styrene) or PLA (Polylactic Acid).

Limitations of the Smart Shelf are the servo motors and the weight of the product. As mentioned, keeping the shelf compact is a priority. While keeping the design compact, the team compromised on the size and power of the servo motor. By doing this the backplate will have less driving force when moving the product, limiting the size and weight of the product. Secondly, having too heavy of a product will compress the button placed on the backplate, this causing a false positive that there are no gaps in the products. Creating a suitable weight range of a product will limit the versatility of the Smart Shelf but would improve the accuracy of the readings. If the team where to choose a button with a stronger spring it may lead to a false negative of lighter products.

**Deployment Architecture Diagram**

**Figure 1:** Deployment architecture diagram shows how each component is linked to the arduino and pi.

**Communication Protocol Diagram**



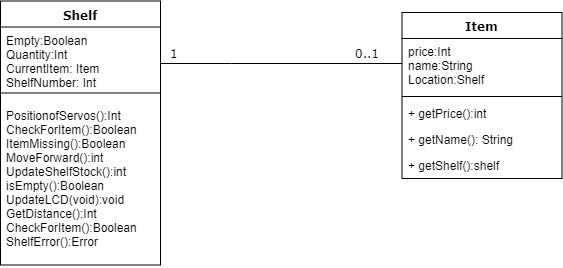
**Figure 2:** This figure shows the communications between all of the objects within the shelf and the devices attached to it. Error Protocols will be displayed on the LCD Screen to notify customer and employees that there is a malfunction in the hardware or software.

**Communication Protocol Table**

|  |  |  |
| --- | --- | --- |
| Sender | Receiver | Packet Format |
| ShelfArduino | ButtonSensor | Check for value of button |
| ButtonSensor | ShelfArduino | [Value of true/false] |
| ShelfArduino | HC-SR04Sensors | Check if HC-SR04Sensors have a value |
| HC-SR04Sensors | ShelfArduino | [Value of true/false] |
| ShelfArduino | ServoMotors | [Distance to move] |
| ServoMotors | ShelfArduino | [Distance moved] |
| ShelfArduino | ClientPi | [Current position of servo] |
| ClientPi | ServerPi | [Position of servo] |
| ServerPi | MobileApp | [Shelf quantity] |
| ServerPi | ClientPi | [return] |
| ClientPi | ShelfArduino | [return] |
| ShelfArduino | LCDScreen | [screen with updated stock] |
| ShelfArduino | MobileApp | [Repair/error message] |
| ShelfArduino | LCDScreen | [Update screen to Repair/error screen] |

**Figure 3:** This table gives a brief description of the nature of the communications which are displayed in the figure above.Sample error scenarios include the servos are malfunctioning by not moving correctly and dropped UDP packet. All errors will be displayed on the LCD Screen so the customer and employees know that a shelf is not working accordingly.

**Database Schema**

****

**Figure 4:** This figure describes the functionalities of the database and data that will be stored within the database.

**Hardware Diagram:**

The Smart Shelf consists of eight components:

· Hypersonic Distance Sensor

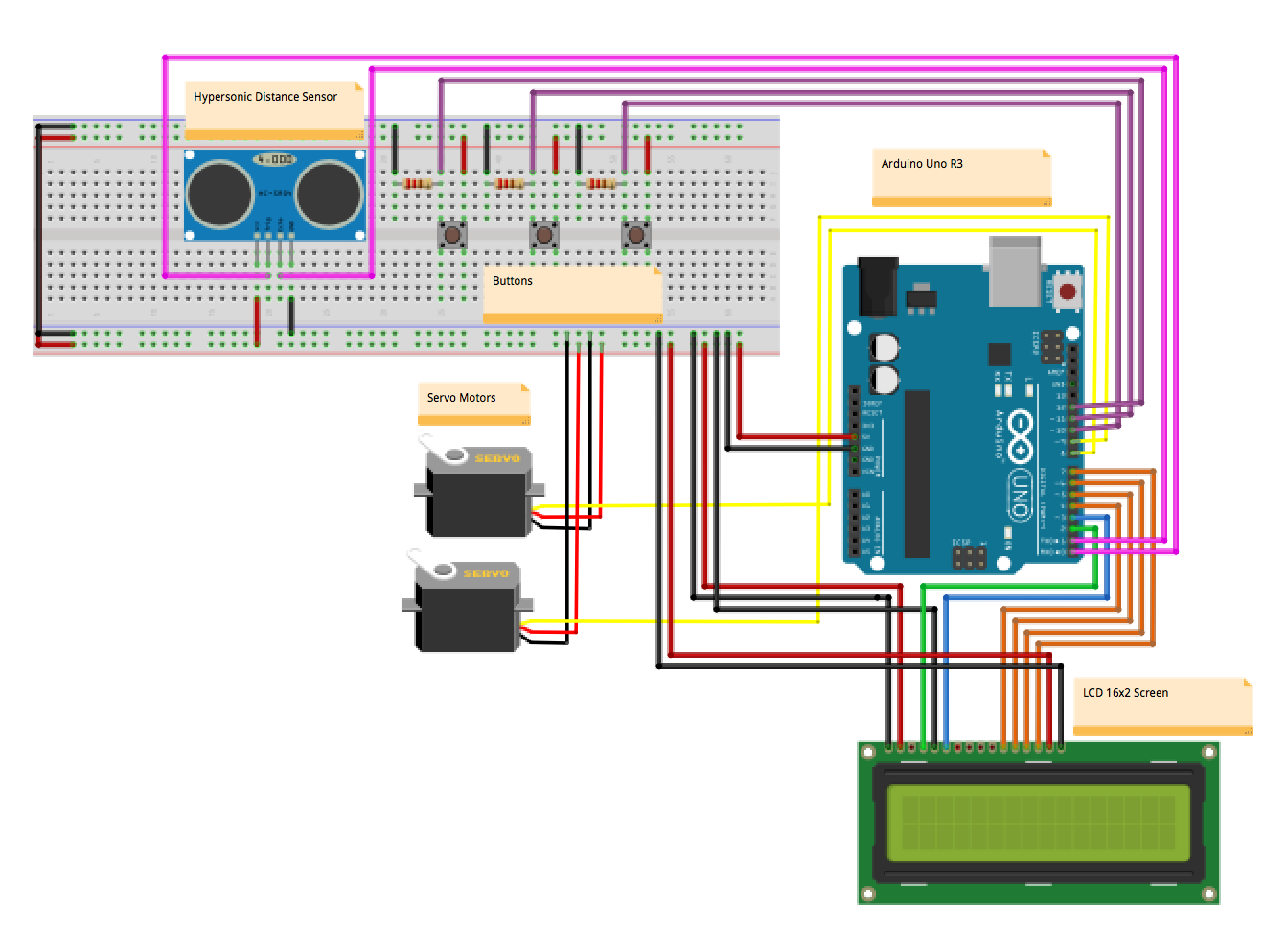
· 3x Buttons

· 2x Servo Motors

· Arduino Uno R3

· LCD Screen

The components will be connected to the Arduino by GPIO, as seen below in Figure 1



**Figure 5**. Visual of how the components will be wired to the Arduino

The Arduino will be used to control each component and read from specific components. Listed below are the different components, the different task performed, and specifications.

**Servo Motors**

The servo motor is a 5 volt DC motor which drives a gear box. The motor will turn to drive the backplate forwards and backwards. The servo can be controlled with the Arduino libraries and boards. The key parameters to abstract from the datasheet are;

1. Voltage Rating: 5V DC
2. Torque Rated (oz-in/ mNm): 22.22/156.9
3. Size: 23mm x 11mm

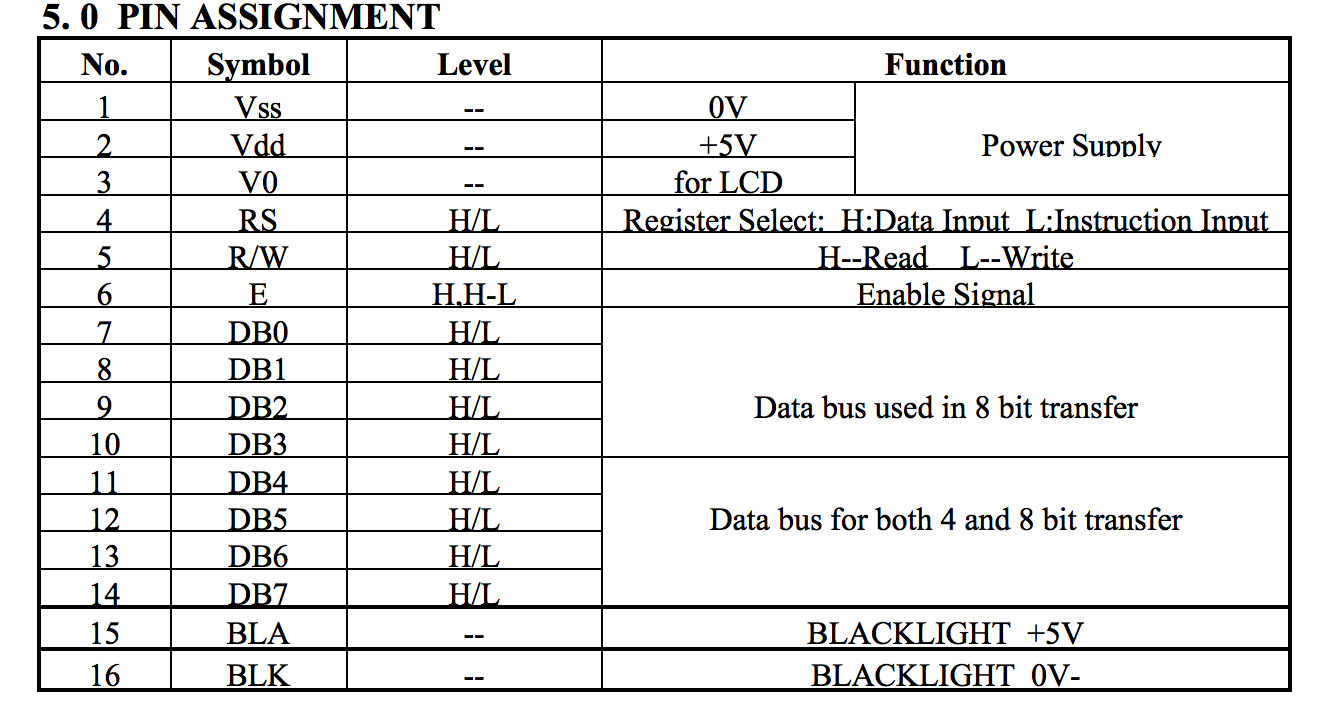
**Buttons**

The buttons are momentary switches which will be able to notify the arduino when they have been pressed. The button will be operating at 5V and will be connected to a 220 ohm resistor

**Liquid Crystal Display (LCD)**

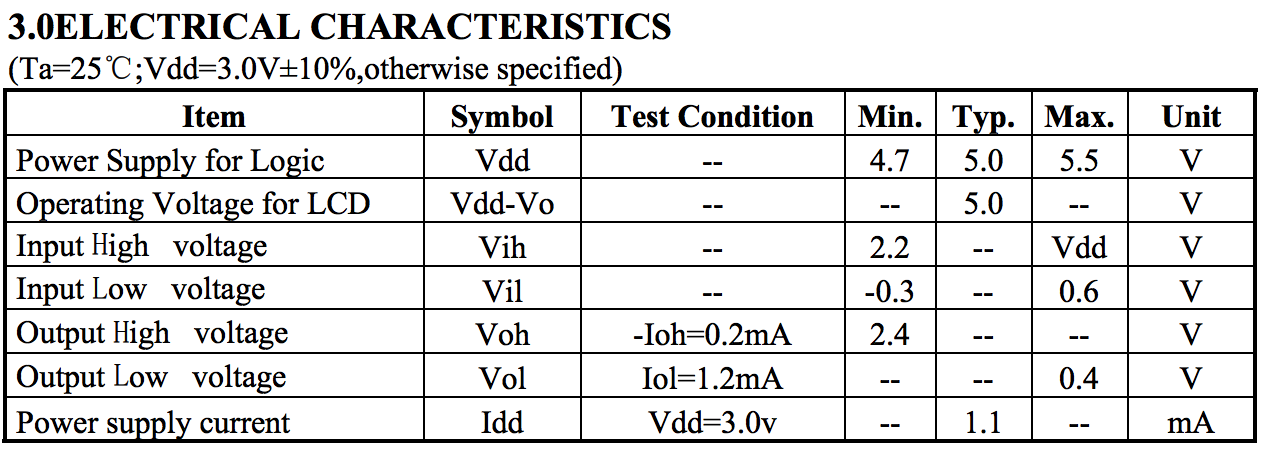
The lcd display that the shelf will use is the Arduino 1602 LCD display. Key abstractions from the data sheet are:

1. The pin placement



**Figure 6:** The pinout for the Arduino 1602 Display

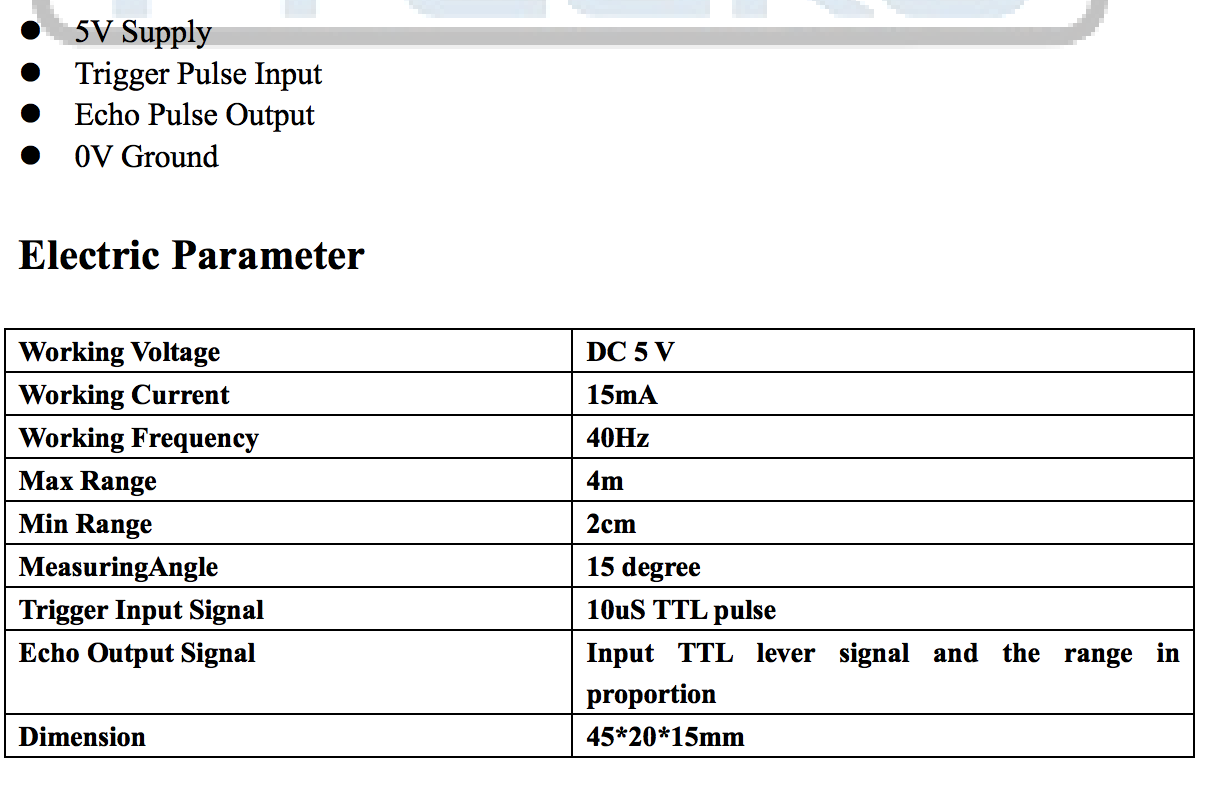
1. Electrical Characteristics Sheet



**Figure 7:** Electrical Characteristics Sheet

**Hypersonic Distance Sensor**

The hypersonic distance sensor being used is the HC-SR04 sensor from Arduino. This uses hypersonic distancing to sense the distance between the edge and the product. Key abstractions from the datasheet are:



**Figure 8:** Datasheet for the HC-SR04

**Software Diagram**