Willy’s Widget Company, LLC

System Context and Product Description

1. Overview

The system you must design and build is for Willy’s Widget Company (WWC). WWC would like to invest in an *integrated customer order and semi-automated warehouse system*. The system must take orders from customers, fill the orders, and move them to the shipping center for packing and shipping.

2. Customer Interface

There needs to be a ***simple*** customer interface that allows orders to be placed. Assume that customers use their laptop or other mobile device to place orders (for our purposes you only need to pick one). Customers need to specify the widgets they want to buy and the number of each that they want to buy. Orders will be sent to the warehouse where they will be scheduled for service in the order of arrival. To keep things simple, you will not need to worry about taking payment from the customer.

3. The Warehouse

The warehouse consists of a robot that is capable of visiting various inventory stations in the warehouse where order items will be loaded (see the Section 5). An inventory station holds inventory of one or more different types. When the inventory at a particular station is loaded onto the bot, the warehouse worker presses a button at the inventory station indicating that the loading is complete. The robot carrier will then proceed to the next station, and the next, and so on until the order is complete. Assume that no single order will be larger than the robot can hold. When the order is complete, the robot carrier will go to the shipping dock where workers will unload the order. Once the order is unloaded, the workers will press a button that indicates the order is unloaded AND complete (we will assume it ships at this point). The process repeats as long as there are orders to fill.

4. System Monitoring and Management

Warehouse supervisors need to monitor and manage the warehouse system. This application will serve several functions:

- Order Status
- Inventory Monitoring and Management
- System Monitoring

4.1 Order Status

It should be possible to display the orders and their status: pending, in-process, backordered, complete. Each of these status’ is explained below:

- Pending: The order is waiting for resources (the robot carrier) to complete orders.
- In-processes: The order is currently being filled by the system.
- Backordered: There is not enough inventory to complete the order (orders are not serviced if there is not enough inventory).
- Complete: The order has been filled (e g. unloaded at the shipping dock)

The system should provide an option that allows the supervisor to see the items that make up an order. The system should indicate (anyway you like) if any items that make up an order are backordered (e g. no inventory of a particular item available). If an order can't be filled because one or more items are backordered, the system will ignore the order and fulfill other orders until the backordered items are available.

4.2 Inventory Monitoring and Management

Warehouse supervisors need to manage and monitor inventory. A manager must be able to add new inventory for the stations. An inventory entry for any item will include at least: item description, cost, number of items, and station number. You may include any other information you deem necessary. This application will monitor and display the available inventory at each station. Inventory will be automatically debited as items are loaded at each station. Assume that the correct number of items will be loaded onto the robot carrier at each station according to the number of items specified in the order (e g. items in inventory and items at the stations will be consistent).

4.3 Robot Status and Monitoring

Warehouse supervisors need to monitor the progress and status of the robot as it moves through the warehouse. They need to know the status of the robot in terms of: location on the warehouse floor, the inventory on the bot, the stations visited, and the next stop (next station to visit or loading dock). You should display overall status:

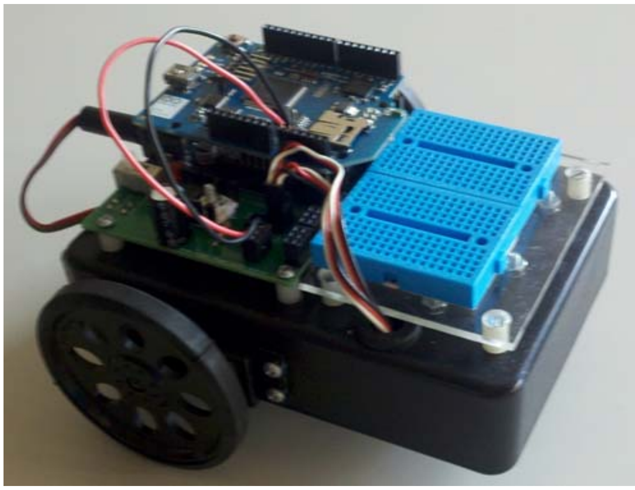
- In-Process: robot is filling an order
- Idle: no orders to fulfill
- Error: problem – include a description of the problem

5. System Hardware

This section describes the hardware configuration of the warehouse and the robot. Note that the hardware for these systems cannot be modified in anyway.

5.1 The Robot

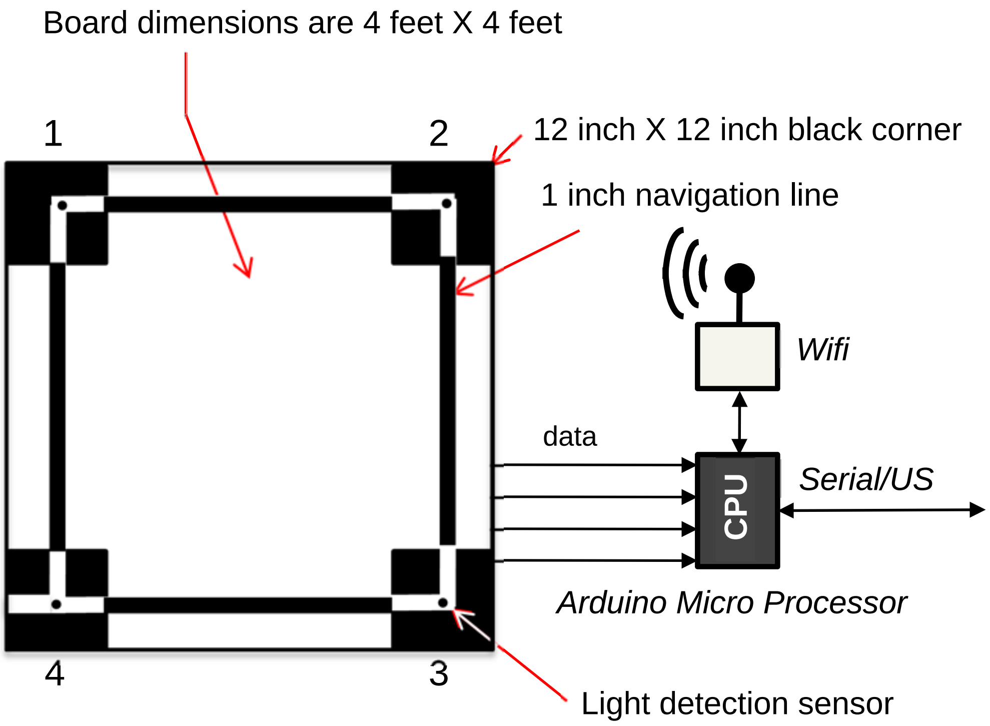
The robot carrier is a two wheeled widget carrier designed for use in the warehouse (described in section 5.2), featuring an Arduino Uno CPU and Wi-Fi card as shown below.



Navigation lines (contrasting black and white) are painted on the warehouse floor so the robot can find its way around. To discern the navigation lines, there are infrared (IR) sensors on the bottom of the robot. You will have to design and develop control software that enables the robot to move around the warehouse from station to station, ending at the shipping station, and repeating the process. Example code has been provided to illustrate how to utilize the robot's various sub-systems. This includes the IR sensors, servos for moving the robot, and the Wi-Fi card to communicate with other applications.

5.2 Warehouse Hardware Configuration

The warehouse is configured with sensors and a control computer. Each inventory station and the loading dock is located in one of the four corners. Corners 1, 2, 3 are inventory stations where inventory will be loaded onto the robot, Corner 4 is the shipping area where the robot will be unloaded.



Corners have black squares and are connected with one another via contrasting navigation lines. In each corner there is a light sensor to detect the presence of the robot over the corner. Each corner has a button located on the side of the game board. This is to indicate that the inventory has been loaded onto the cargo bed of the robot. The sensors and switches are connected to pins on the control computer. The control computer is an Arduino Uno microprocessor. The control computer has Wi-Fi capability to communicate status. The status of the load switches and the photo sensors can be read with the Arduino.

Your System Monitoring and Management system needs to communicate with the warehouse/robot to coordinate the movement of the robot from station to station to fill orders. Note that you can choose how to communicate with the warehouse and robot to coordinate activities. There are several options. The warehouse control computer (Arduino) can be connected to an external computer (e g. laptop) via USB (appears as a com port) or via the Wi-Fi card on the control computer. Note that the robot also has a Wi-Fi card as well and it is possible to communicate with robot via the control computer or directly from a laptop. So an important design decision will be to determine how all these parts will communicate with the System Monitoring and management system.

6. Sample Code Archive

An extensive archive of example code has been provided that illustrates the following:

- how to move the robot
- how to read the IR sensors
- how to use the Arduino Wi-Fi cards as clients and servers
- Java code illustrating communication as client and server with Arduino Wi-Fi cards
- Java code illustrating how to read data from the Arduino control systems USB port
- how to read the status of the light sensors and station switches

All of these examples assume that you are running the Windows 7 OS or better and Arduino 1.0.4 (note the latest version 1.0.5 is incompatible with the Wi-Fi cards).

7. Deliverables

Your team should produce a comprehensive architecture design, presentation, and system demonstration:

7.1 Design Document

Your team will produce a design document that addresses the following:

- Description of the project context.
- Brief description of the architectural drivers. For quality attributes, focus on the high priority quality attributes. Describe them using clear quality attribute scenarios using one of the formats presented in class. Prioritize your quality attributes. The architectural drivers should be used as a basis to design the architecture, reason about alternatives, and evaluate the fitness of the architecture.
- Description of the design in terms of views from the various perspectives discussed in class. Be sure to include sufficient detail in order to analyze the extent to which the key drivers are supported or inhibited.
- Include an analysis of the architecture that discusses rationale, key tradeoffs, and argue why your design is fit for purpose with respect to the architectural drivers.

You are free to structure your design document anyway you like, but you should follow the documentation principles presented in class and in the readings.

7.1.1 Future Needs

In addition to the requirements listed in this project description, you should also address the following future needs of the stakeholders (note that you do not have to implement these requirements):

- How would you have to change your system to support fulfilling a single order that was larger than the robot could hold?
- How would your design support scaling out to include more inventory stations within the same warehouse? What would have to change (other than the physical warehouse) to support this requirement?
- How would your design support a distributed inventory management? Imagine that an order is received at one location, but the inventory is located at, and

shipped from multiple identical warehouses. What would have to change (other than the addition of multiple warehouse) to support this requirement?

- How would your design support two robots in the same warehouse? What would have to change (other than the addition of another robot to the warehouse) to support this requirement?
- You must address these questions in your design document and final presentation with clear realistic answers based on your design.

7.2 Team Presentation

Your team will provide a 30 minute presentation on the last day of class that summarizes the following points:

Programmatic aspects…

- team organization and rationale for organization
- schedule and schedule performance
- presentation of time logs

Overview of the architectural design…

- context and the high priority architectural drivers
- your technical decisions and approaches
- the design, your rationale, and tradeoffs – this should include a discussion of how the system meets the key quality attributes that you will not be able to demonstrate

Lessons learned…

- describe what when well and what did not go so well organizationally, programmatically, technically, architecturally
- what would you do differently in the future
- feel free to analyze all aspects of the project

8. Demonstration Guidelines

You will also be required to demonstrate your system. Specific test cases will be provided later in the program. At a minimum you will have to demonstrate:

- order taking
- filling an order
- inventory management and status monitoring
- order status monitoring
- robot monitoring

If you robot gets stuck, you may intervene provided you adhere to the following guidelines:

1. the monitoring systems must indicate that an error has occurred requiring human intervention
2. you may not touch the robot or relocate robot
3. you are able to correct the problem in 3 minutes

Each team will have up to three attempts to satisfy all of the key test cases. If your robot gets lost, communication is lost, no error is reported when there are errors, or your robot is in danger of falling off the table, the test will end.

9. Assumptions and Hints

Some basic assumptions and hints:

- we will use Arduino IDE and libraries 1.0.4, Arduino IDE and libraries 1.0.5 has firmware incompatibilities with our Wi-Fi cards
- keep all user interfaces simple
- you may use any third party software you like, but you must justify your choice of technology
- assume that anything can fail… your design should account for failure and recovery to the greatest extent possible
- keep your robot’s batteries fresh
- all of the fundamental code pieces you need for the warehouse and robot is provided as an example in the archive – you will need to design the system, you should not be fumbling with low level technologies
- you may use a database if you choose (e g. MySQL) but it is not required
- Netbeans is a nice simple tool (https://netbeans.org/) for developing user interfaces very quickly
- you will use your own laptops for the robot and inventory monitoring systems and order taking systems