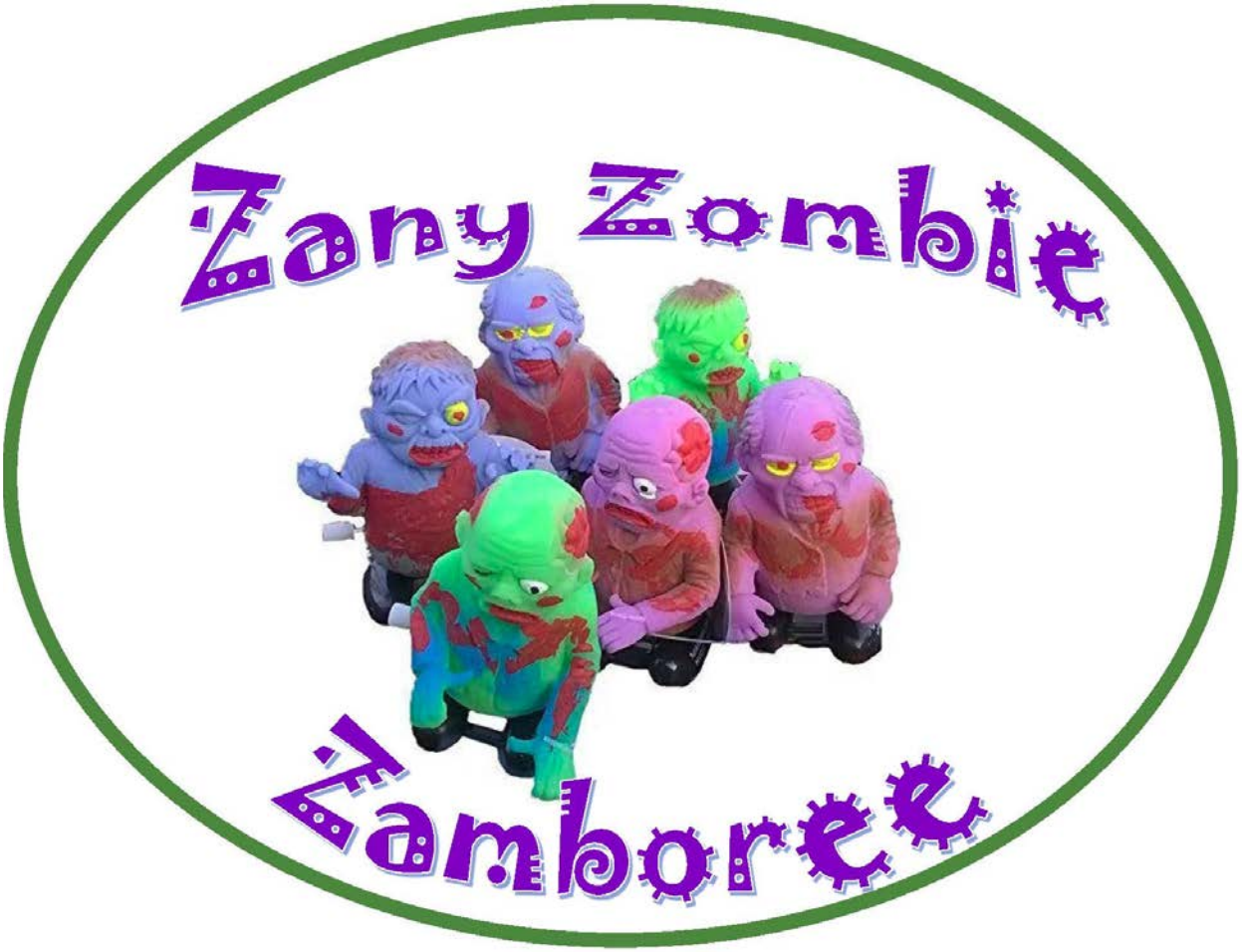

Product Design Specification

Fall 2017

Version 1.0

Final Project
Zany Zombie Zamboree
Product Design Specification

Learning Objective:

Design a mobile robot system to retrieve various zombies and plants and deliver them to holes of various sizes or designated delivery areas.

Learning Outcomes:

At the conclusion of this project, the student team should be able to

- Program a robot to use a gripper or other custom part to retrieve zombies and plants and deposit them in various holes or deliver to designated areas.
- Program a robot to navigate autonomously in a cluttered environment to retrieve and deliver zombies and plants.
- Attach peripherals to a mobile robot and create a program to wirelessly control it.
- Attach sensors to a mobile robot and create a program to detect walls, floors, or other objects in its environment.
- Create properly commented code with modular functions to achieve all of the objectives of the final project.
- Demonstrate the working project and adequately explain its functionality.
- Document the learning objectives achieved through memos, a final report and presentation.

Parts:

- Arduino Shield Bot with Gripper
- Arduino Uno Microcontroller
- 2 infrared sensors
- 2 sonar sensors
- 2 object detection sensors
- 1 breadboard
- SparkFun Inventor Kit
- 1 battery charger
- 2 nRF24L01 Wireless Transceivers
- 1 joystick
- 1 laptop keyboard

- 1 remote Control & infrared receiver
- 1 Wiimote & Dongle (optional, limited availability)
- 1 PlayStation Controller & Bluetooth Dongle (optional, limited availability)
- 1 CMUcam5 Pixy Camera (optional, limited availability)
- Team Custom Parts ($10 or less)

Preparation:

In order to prepare for the final project, review the following links as well as any supplemental materials provided for this course including the mini-projects and midterm projects. The team should also conduct any online research necessary to help complete the final project. Make sure to take good notes and keep good records of any resources used as these will be part of the final project report literature review.

- Arduino Website www.arduino.cc
- Shield Bot http://learn.parallax.com/ShieldRobot
- IR Remote https://www.arduino.cc/en/Tutorial/RobotRemoteControl
- Sonar Sensor https://www.arduino.cc/en/Tutorial/Ping

  http://playground.arduino.cc/Code/NewPing

  http://www.instructables.com/id/Easy-ultrasonic-4-pin-sensor-monitoring-hc-sr04/
- Color Sensor https://www.sparkfun.com/products/12787

  https://github.com/sparkfun/APDS-9960_RGB_and_Gesture_Sensor (Sparkfun Library)

  https://learn.adafruit.com/adafruit-color-sensors/ (includes library and tutorial)
- Sharp IR Sensor http://playground.arduino.cc/Main/SharpIR
- Linearize Sharp IR Sensor https://acroname.com/articles/linearizing-sharp-ranger-data, http://projectsfromtech.blogspot.com/2014/10/linearizing-sharp-ir-ranger-2yoa21-with.html
- PlayStation Controller Library http://playground.arduino.cc/Main/PSXLibrary

  http://playground.arduino.cc/Main/AnalogPSXLibrary

  https://www.youtube.com/watch?feature=player_embedded&v=xlupRVF_6W8
- CMUcam5 Pixy Camera

  http://cmucam.org/projects/cmucam5/wiki/Hooking_up_Pixy_to_a_Microcontroller_(like_an_Arduino)

Procedure:

Phase I – Remote Control Demonstration

- Attach at least 2 of the following wireless communication devices to the robot (PlayStation Controller & Bluetooth Dongle, Joystick & 2 transceivers, laptop keyboard & 2 transceivers, Remote Control & Infrared Receiver, Wiimote & Dongle).
- Use each device setup to wirelessly drive the robot, retrieve and deposit zombies and plants in the various holes or designated storage area.
- Use a decision matrix to determine which method the team will use to remotely control the robot for the final competition. Some examples of criteria are programming ease, communication lag, range, flexibility, ease of use, accuracy, degrees of freedom, etc. The decision matrix must be included in the project final report.
- A program flowchart, state diagram or pseudocode for the remote control mode must be included in the final report.

Phase II – Autonomous Control Demonstration

- Attach at least 2 of the following sensors to the robot (infrared, sonar, object detection sensor, CMUcam5 Pixy Camera). You should try a variety of mounting locations and orientations.
- Perform tests on the sensor data to relate distance information to analog data read by the program. Use an equation to linearize the data and use the expression in your code. You must include an Excel table and graph of the sensor distance and analog data in the appendix of the final report
- Use the sensors to perform tests on whether the robot can accurately detect objects, holes, and walls based upon distance or size. Create a table to compare and contrast performance based upon sensor and zombie or plant type along with identified criteria. Some examples of criteria are programming ease, flexibility, repeatability, accuracy, and knowledge.
- Use a decision matrix to determine which position, orientation, and sensor combination the team will use for the robot's autonomous control.
- Write a program so that the robot operates for 30 seconds in autonomous mode to gather and deliver plants and zombies to holes and a designated storage area. You must use the sensor data as part of the program.
- A program flowchart, state diagram or pseudocode for the autonomous control mode must be included in the final report.

Submission Requirements:

Demonstration

You will demonstrate the final project in parts with specific milestones. You will show your working project to the professor or lab assistant by the due dates. The phase I, II, and III demonstration rubrics are available on Moodle.

Code

Upload your properly documented code to the Moodle DropBox by the due date. Properly documented code has a header comment with the name of the file, name of the author/team, date created and a description of how the code works. The header comment should also have a description of any key functions or variables in the program. There should be no magic numbers or numeric values not related to a constant or variable to make it clear what it does. There should also be in line comments to describe all major functions as well as lines of code as needed. In addition, as your code becomes larger and more complicated, you should create functions with logical names that are called from the main function (`void loop()`, `void setup()`) to make the program readable and easier to follow. Someone unfamiliar with your project should be able to read your code and understand the functionality based upon your formatting and comments.

Memo

Create memos and submit to the Moodle DropBox and Google shared team folder which meet the following requirements:

- ✓ Format
  - Begins with Date, To , From, Subject
  - Header includes team name, member names, faculty name, date and subject
  - Memo addressed to the course faculty member
  - Font no larger than 12 point font
  - Spacing no larger than double space
  - Written as a paragraph for the purpose and overview
  - Key decisions or progress updates can be a bulleted list
  - No longer than three pages of text
- ✓ Writing
  - Memo is organized in a logical order
  - Writing is direct, concise and to the point
  - Written in first person from team members
  - Written in past tense
  - Correct grammar, no spelling or punctuation errors
- ✓ Content
  - Starts with a statement of purpose
  - Summarizes key problems and accomplishments for the prior week
  - Provide major action items and list of issues or work to be done the next week
  - Discusses the strategy or pseudocode for implementing the project
  - Current progress on the project
  - Current contribution and time completed on the project by each team member
  - Discusses the tests and methods performed
  - Clear statement of conclusions

Final Report

Create a final report on the final project that meets the following requirements:

- The document should have default Word settings with respect to font and margins
- All pages should be numbered
- All headings must be numbered, left-justified, bolded, and capitalized at the beginning of the section.
- All figures must have a number and title (caption) underneath (images of circuits, breadboards)
- All tables must have a number and title (caption) above it
- The cover page should have title, partner names, course number and title, date
- The report should be organized and free of spelling, grammatical, and punctuation errors.
- All graphics should be detailed and high quality
- The report should have the following sections:
    a. Executive Summary
    b. Introduction
    c. Requirements
    d. Research/Literature Review
    e. Design Method
    f. Verification and Results
    g. Conclusions
    h. Recommendations
    i. References
    j. Appendices (Code, Flowcharts)

Final Presentation

Give a final presentation on the project that meets the following requirements:

- Team Introduction with name and major of each member
- All members have a professional, business casual appearance
- Title slide has title of presentation with member names
- Quality graphics with minimum animations/transitions used in the slides
- Presentation has a problem statement and project overview
- Presentation has a clear method, verification, results, conclusions, and recommendations
- All members participate in the presentation
- The presentation is practiced with minimal reading of note cards or slides
- Team members show knowledge of the project
- Team recommends a final project for future courses
- Team handles questions well
- Team members engaged and asking questions when other teams give final presentation

Grading Criteria:

The project score will be scored based upon the following rubric:

Score	**100%**	**60%**	**20%**	**0%**
Demonstration	All parts of the project function and can be completely explained	Most parts of the project function and can be mostly explained	Very few parts of the project function and the functionality could not be explained	No demo completed
Code	The code includes header and in line comments with clear variables and modular functions as needed	The code is missing some necessary comments, variables and is not clear or modular as needed	The code has very few comments, variables, functions or modularity	No code submitted
Memo	Follows all of the guidelines with respect to the writing, content, and format	Follows most of the guidelines with respect to the writing, content, and format	Follows few of the guidelines with respect to the writing, content, and format	No memo submitted
Report	Follows all of the required guidelines for the final report	Follows most of the required guidelines for the final report	Follows few of the required guidelines for the final report	No final report submitted
Presentation	Follows all of the required guidelines for the final presentation	Follows most of the required guidelines for the final presentation	Follows few of the required guidelines for the final presentation	No final presentation given

Please refer to the course schedule for the due date for all assignments. Assignments not submitted on time will incur a 20% late penalty per day. Assignments more than one day late will not be accepted. You cannot receive an incomplete on the final project and failure to complete this assignment may result in failure in the course.