Technical Report

The Design Process and Testing of an Autonomous Soccer Field Layout System

To:	Claremont AYSO
From:	Elizabeth Poss Kai Kaneshima Felipe Borja
Date:	May 4, 2016

Abstract:

This report outlines the design process used to create an improved soccer field layout system for Ms. Diane Flowers and her organization, AYSO Claremont. The team identified objectives, constraints, and functions to reduce the time spent laying out the field, and improve the accuracy and ease of use of the existing design. The resulting prototype used an Arduino interface to autonomize the process. This design modified the existing painter, providing a low cost alternative that users would be familiar with. While testing was limited by available materials, all available metrics showed that the prototype would successfully be able to lay out and paint soccer fields of different sizes accurately and in a substantially shorter amount of time than before.

Table of Contents

List of Figures

List of Tables

I. Introduction

Laying out and striping soccer fields is an often overlooked process that is vital for effective game play. While straight lines and accurate angles are important for the players, they are also important for the referees, who need to determine when the ball has gone out of play. If a line is not straight or if the field is not dimensioned properly, the job of the referee is much more difficult, and game play is negatively affected.

The current design to paint fields uses a four wheeled cart with metal brackets to hold a can of paint upside down. When a trigger on the handle of the cart is squeezed, the nozzle of the paint can is depressed, making it spray out paint below the cart. The current procedure has several limitations. The wheels of the cart are attached very close to the cart's chassis, which prevents the cart from turning from side to side. While the locked wheels facilitate straight line painting, it is difficult to paint the penalty arcs and center circle, thereby reducing accuracy. Currently a tape measure is used to lay out the field, measuring from one side of the field to the other. The corresponding end points of this line are then marked for later use. Ninety degree angles for the corners of the field and other rectangles are simply 'eyeballed', limiting precision. Furthermore, the amount of time required to layout a field takes approximately three hours and painting over an existing field takes about ninety minutes. This process requires three to four people. Fields need to be repainted every week, costing time and money for paint, because the paint completely fades after two weeks, necessitating a complete relayout. As a result, the fields can only be used for soccer, preventing the space from being utilized by other youth sports. Thus, a product that allows for a faster layout and striping process, would reduce the time demands, costs and allow fields to be repurposed for other activities.

Our liaison, Diane Flowers, presented the above design challenge to engineering students at Harvey Mudd College, and requested that the students improve upon the existing painting process. The students then visited the field and assisted the liaison in painting over an existing field to gain a better understanding of the problems associated with the current process. The initial problem statement follows below:

> *The lining of soccer fields is a challenging task that is often conducted by volunteers. Frequently, the initial layout is challenging to achieve with respect to accurate angles and distances. A solution is required that will make the positioning of the elements of a soccer field easily identified on the field and will lead to immediate or almost immediate marking of the field using conventional painting methods. The solution should identify the points for all field painting. A very attractive solution would be one that is autonomously capable of lining the field with limited human input.*

II. Background Research

The team began by researching how other organizations paint soccer fields. Most professional soccer fields have built in stakes to mark important points. However, professional fields have one standard size, whereas the dimensions of youth soccer fields depend on the ages of the players. In a meeting with our liaison, she emphasized the need for scalability. Thus, these permanent modifications are not a logical choice for the design.

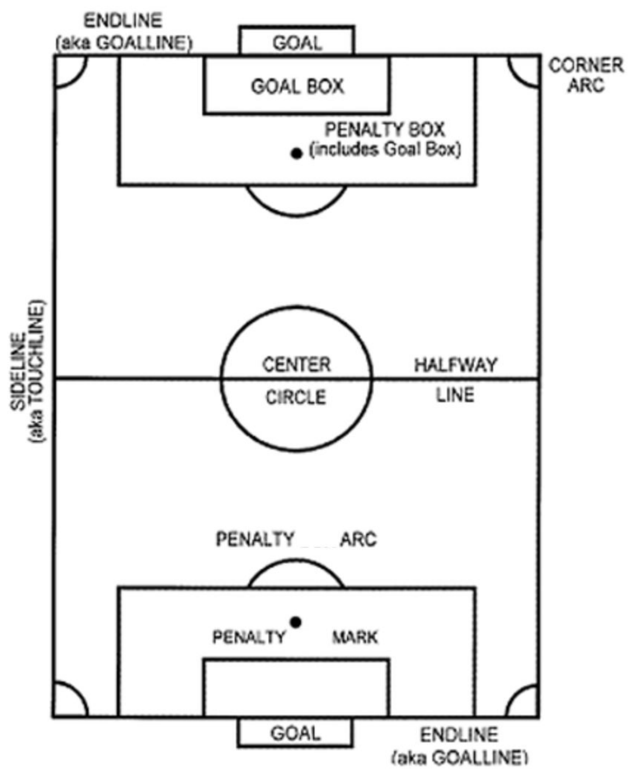


Figure 1. Soccer Field Layout

Soccer Field Kits has developed a design that uses a system of premarked tape measures and a system of stakes to lay out a soccer field (cf. Figure 2). The tape measures would be aligned at a 90 degree angle before using stakes to mark where the halfline, goal box, penalty box, etc., should intersect with the existing line. While this process supposedly takes less than 20 minutes, it would require multiple people to hold the measuring devices and place down the stakes. Furthermore, there was no mention of how the lines would be painted or how straightness could be guaranteed.

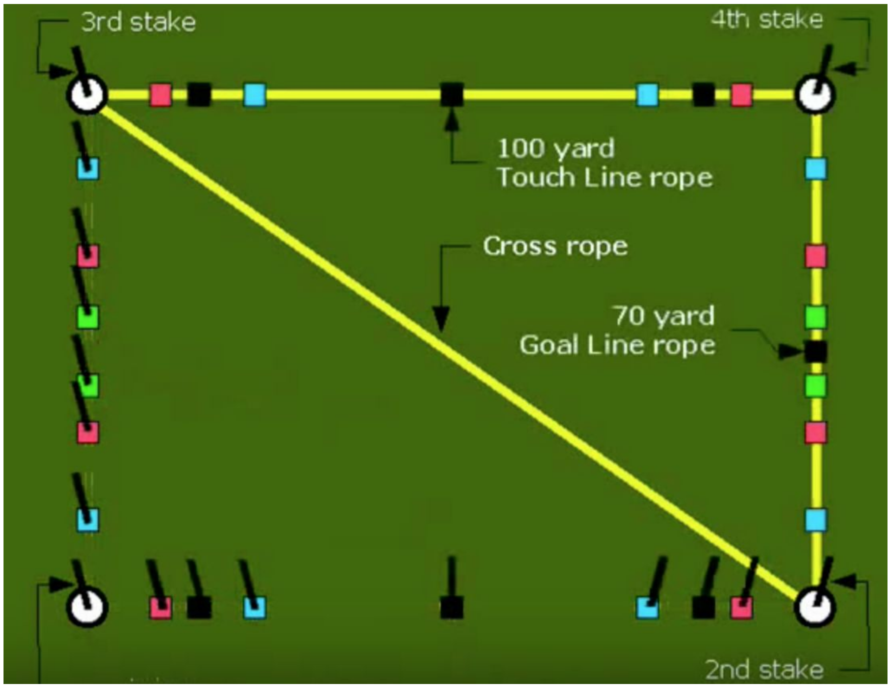


Figure 2. Alternative Layout Method

In addition, neither design is scalable, something that the liaison specifically requested.

Epic Sports Soccer sells a Ameri-Stripe Aerosol Marking Stick, referenced from here on as a hand painter (cf. Figure 3). The design holds a can of paint, and uses the same trigger method as the cart to spray out paint. The handpainter can be moved freely, which would be very useful for painting curved lines. However, this means that this device could not be used to paint straight lines.

Figure 3. Hand Painter

III. Methodology

A. Constraints

Before deciding on possible objectives or necessary functions, the team decided to identify the constraints governing the project, which came directly from the liaison following an initial meeting. Constraints are limitations put on the design space that must be upheld regardless of the situation. If a constraint is not met, the product has failed. Below are the constraints associated with this project.

- *The time to lay out the field must be under 3 hours:* It currently takes 3 hours to lay out a new field. This process simply takes too long and the product created must reduce the amount of time needed.
- *The time to stripe the field must take under 90 minutes:* The liaison repaints the field every week in order to make the lines more visible. As a result, the liaison does not need to lay out the field again and simply needs to paint over the existing lines. However, the error involved in the repainting process is quite high, as it is hard to ensure that the paint is falling directly on top of the previous lines.
- *No more than three to four operators:* The layout process currently requires three to four volunteers to manually lay out the field, as a tape measure is used to measure out each dimension of the soccer field. Because multiple fields must be painted, needing multiple people to paint a single field is inefficient. The design is required to reduce the number of people necessary to layout a single field.

- *Very low angle tolerance:* There is no standard for creating the right angles for the field. Currently, the user just ‘eyeballs’ it. The design must create very precise angles in order to ensure the layout of an accurately measured field. Tests showed that an error of about .5 degrees was acceptable.
- *No prior user training necessary:* The liaison specified that volunteers usually paint the soccer field. Thus the product must be designed for users with no previous knowledge. If a volunteer has no background painting a soccer field, the product must be able to guide them.
- *Only uses paint for drawing lines:* The liaison specified that only paint can be used for striping the fields, as chalk would kill the grass. Furthermore, the paint is distributed by the same company that the liaison buys other materials from, so the product must work with the existing type of paint can.
- *The product must cost under $100.00:* The current field striper costs approximately $100.00. The design should cost less than this, or function with the existing painter. Because the liaison works for a volunteer based company, the product must be affordable.
- *Line width must be between two to four inches:* The lines painted by the field must meet the standard width of two to four inches, which allows for the best visibility by the soccer players and referees.
- *No permanent modifications can be made to the borrowed field painter:* The liaison allowed the team to borrow the current field painter for testing. The team was not allowed to make any permanent physical modifications to the cart.

B. Objectives

After determining the constraints, the team formulated the objectives for the product. Objectives are goals that the product is desired to meet, but not necessary for functionality. Below is a pairwise comparison chart, which compares objectives to each other and then awards points to the more important objective. A total score for each objective was then calculated; the objective with the highest score is the most important.

Table 1. Piecewise Comparison Chart

	Be faster than conventional methods	Be reusable	Be sustainable	Be inexpensive	Be precise	Be scalable	Be easy to use	Be portable	**Total:**
Be faster than conventional methods	-	1	1	0.5	0	0.5	1	1	**5**
Be reusable	0	-	1	0	0	0	0	1	**2**
Be sustainable	0	0	-	0	0	0	0	0	**0**
Be inexpensive	0.5	1	1	-	0	0	1	1	**4.5**
Be precise	1	1	1	1	-	1	1	1	**7**
Be scalable	0.5	1	1	1	0	-	1	1	**5.5**
Be easy to use	0	1	1	0	0	0	-	1	**3**
Be portable	0	0	1	0	0	0	0	-	**1**

Below are the objectives ranked in order of importance.

1. *Be precise:* The liaison put tremendous emphasis on precision, as the field should be a perfect rectangle that contains perfectly made circles, and goal and penalty boxes that are placed at their precise locations.

2. *Be scalable:* The product should be able to lay out and paint soccer fields of different dimensions. It should be able to accommodate standard field sizes, as well as scale the field dimensions for areas of nonstandard size.
3. *Be faster than conventional methods:* The product should considerably decrease the amount of time required to stripe and lay out a soccer field.
4. *Be inexpensive:* The product should not cost a lot to increase accessibility to volunteer-based soccer leagues.
5. *Be easy to use:* The product should be fairly easy to use, as volunteers with no prior experience often paint these fields.
6. *Be reusable:* The product should be able to be used many times without the need for replacement, thus lowering the long term cost.
7. *Be portable:* The product should be fairly portable so that it can be used easily used by only one person. In addition, it must be easily stored.
8. *Be sustainable:* The product should retain functionality for a reasonably long period of time.

IV. Conceptual Design

Once the design space was fully defined, the team started searching for the best design alternative. This process began by generating a large number of possible ways to fulfill the desired objectives and then narrowing these options until the best one had been chosen.

A. Functions

A function is an action that the product performs. The functions are represented as verb noun pairings.

- *Scale field area*
- *Mark important field points*
- *Paint accurately over laid out lines*
- *Creates exact 90^o angles*
- *Lay out straight lines*
- *Operate on multiple terrains* (Because soccer games are played on grass and occasionally dirt, the product should be able to operate on multiple terrains.)

B. Means

Next, the team explored possible means of fulfilling these functions. The team brainstormed a large number of possible means with each corresponding to one of the functions the product was meant to perform. These ideas were displayed in a Morph Chart, a structure which allows us to group possible means by the functions they fulfill.

Table 2. Morph Chart

Scale field area	Computer program	Preset field types	User input	Physical scale	Phone application
Mark important field points	Tape measure	GPS marker	Paint	Stakes	Laser pointers
Allow users to accurately paint over laid out lines	Locked wheels	Metal marker, paint thing next to chassis	Hand painter attached to cord + stake for circles	Angled wheels on painting cart	Guiding piece of wood the width of the line
Creates exact 90º angles,	Angled guides	Intersection of two lines	Measuring angles	Tape measures	Laser guides
Lay out straight lines	Laser lever	Straight edge	Arduino		
Operate on multiple terrains	Current wheel system	More visible paint	Rugged wheels		
Produce Accurate Circles	Hand painter + string	String + stake system	Angled wheels		

C. Design Alternatives

Now, with a variety of means of fulfilling our design, the team attempted to evaluate these means for compatibility with each other. This approach briefly set aside the efficiency or success of the individual means, in favor of finding defined groups of means that would work together effectively to fulfill our entire design. This allowed us to come up with a number of design alternatives that could be evaluated for complete fulfillment of constraints, and then for overall efficiency.

The design alternatives separated into two different design paths. The first was an exercise in optimizing existing methods. The user would employ a system of stakes, tape measures, and wood markers to lay out the field, then use the existing striping method to paint the field. The system of stakes and markers would be modified to allow for scalability, and actions would be optimized to reduce time and number of users necessary. The second path was a greater deviation from existing methods. The painting and layout process would be combined into one, and the painter would be modified so that it could measure its position on the field. This would be achieved by attaching a wheel encoder to the painter, allowing us to measure distance traveled. An Arduino microcontroller would provide a user interface that would direct the user in the steps of laying out the field. A third possibility briefly explored was the use of laser levels to generate the straight lines of the field. This design would function much like the first path, but using laser levels instead of string/tape measures to generate straight lines.

D. Choice of final design

These design alternatives were first evaluated to ensure that they could fulfill all constraints. During this evaluation the team chose to exclude the laser level based design, realizing that we would not be able to buy the necessary equipment with the existing budget. The team used a Best of Class Chart (cf. Table 3) to compare the effectiveness of the two remaining designs. Each design's ability to fulfill the objectives was ranked (lower rank corresponds to better fulfilment of objective), and ultimately the Arduino assisted painting method was found to be more effective. Overall, the team supported this design because it represented a greater improvement upon the field layout process, rather than just optimizing an existing process. This design was also chosen because it combined the field layout and painting process. This marked a significant improvement for the objective our client ranked most highly- layout time. Even if we failed to decrease the layout time, combining the two processes would decrease the total time needed to lay out and paint a field. With the design chosen, the team began the process of designing a functional prototype.

Table 3. Best of Class Chart

	Design 1	Design 2
	Wheel encoder	**Stakes + tape measure**
	arduino display	**wood marker**
	led markers/display	**paint over string/tape measure**
Be faster than conventional methods	1	2
Be reusable	2	1
Be sustainable	0.5	0.5
Be inexpensive	1	2
Be precise	1	2
Be scalable	0.5	0.5
Be easy to use	1	2
Be portable	1	2
	8	**12**

V. Designing and Prototyping

A. Hand Painter/Circle Apparatus

The team began prototyping by creating the independent devices necessary for drawing the circular segments of the soccer fields. The wheels of the painter are fixed and are thus inadequate for drawing the curved segments of the field. The best course of action was to produce a separate tool that facilitated the painting of the center circle, penalty arcs, and corner arcs.

The hand painter was inspired by a compass (drawing tool) and by models seen during background research. The full hand painter apparatus consisted of a purchased hand painter to draw the circle, and a metal stake used to mark the center of the circle. The two were joined by a length of string extending from a freely-rotating spool on the stake to a fitted metal connector on the hand painter. The string was wound around the spool, with every yard marked by a black flap of tape for ease of use. An alligator clip was used to hold the selected flap of tape onto the spool, letting the user draw a circle of fixed radius. The spool and metal connector are at the same height compared to the field, ensuring a radius of the exact length requested.

The hand painter is operated by a single person during the field painting process. During the overall painting process, the user would be instructed to draw a circle or arc of a given radius around a marked center point, using the hand painter connected to a stake using rope.

Specifically, the stake would be fixed in the ground at the center of the circle, and the user would unravel a length of string equal to the desired radius. The string would then be clamped in place and the operator would walk until the string was taut. Then, using the hand painter, they could paint in a circle until the desired shape had been fully painted. The spool on the stake would rotated freely with the operator's walk, to prevent tangling that could alter the radius.

Once the circle was completed, the operator could roll the string back into the spool with ease and continue painting the straight lines and angles of the soccer field until another circular segment needed to be drawn. The combination hand painter and stake is sufficiently compact to be carried in the paint cart's storage basket without hindering its movement.

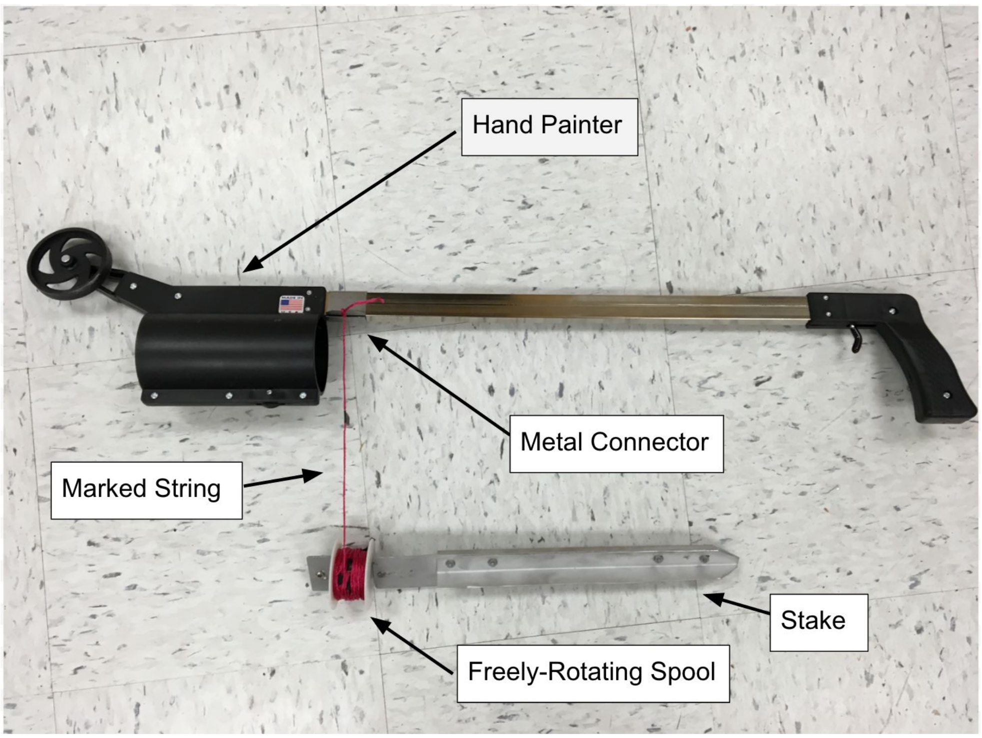


Figure 4. Hand Painter Apparatus and Functions

B. Wheel Encoder Apparatus/Sensor

The team choose to modify the painter with a wheel encoder. A wheel encoder is a sensor attached to a wheel that measures the number of wheel rotations. Because the circumference of the wheel is known, the distance that the painter has traveled can then be calculated. However, the addition of a wheel encoder can be achieved in many ways. Common methods include an optical wheel encoder, in which a sensor measures light and dark patches on the wheel, or a limit switch, in which a part of the wheel presses a switch every time it rotates past the sensor.

Given these options, the team had to choose a sensor that would act as a wheel encoder and could attach to the painter without extensive modification. Because most optical wheel encoders must be attached to the axle of the wheel, the team was forced to eliminate this option. The fixed nature of the painter's wheels meant that there was not enough space on the axle for an optical wheel encoder. In addition, the possibility of dirt and grass collecting on the wheels of the painter would introduce more sources of error. The team then considered using a limit switch - a sensor most commonly designed as a piece of metal that senses when it has been depressed. However, most limit switches have a limited range of motion, i.e., they cannot be moved both forwards and backwards. This range of motion would be necessary for a painter, as it is moved forward and backwards during normal use. This leaves the option of a rotary limit switch, a limit switch with a much larger degree of possible movement. This sensor would reduce error by making the units of rotation very clearly defined, with fewer sources of error. However, only a few of the available rotary limit switches within our budget allowed for a full 180º of motion. While a viable, inexpensive rotary limit switch was found, it would have required permanent modifications to the painter to mount and test. Because permanent physical modifications were not allowed for the prototype, the team choose to use a different approach for the prototype.

For the prototype, the team chose to engineer an optical wheel encoder using a reflectance sensor. The reflectance sensor measures the reflectance of a surface, a value affected by the color of a surface, and the distance between the sensor and the surface. The team mounted temporary black color markers on the existing spokes of the wheel for the reflectance sensor to detect. This allowed the sensor to measure the number of rotations of the wheel. The team used 10 markers that were evenly spaced around the wheel at each of the existing spokes, to reduce the error if the sensor failed to detect one of the markers.

The sensor used was a QRD1114 Digikey Reflectance Sensor. The circuit used was sourced from the sensor array used in a previous engineering class (E11), because the sensor array contained a backup photoreflectance sensor. Although the team did not use this sensor, it allowed for the possibility of mounting a small LED somewhere on the wheel, and measuring rotation by tracking the number of flashes of light measured.

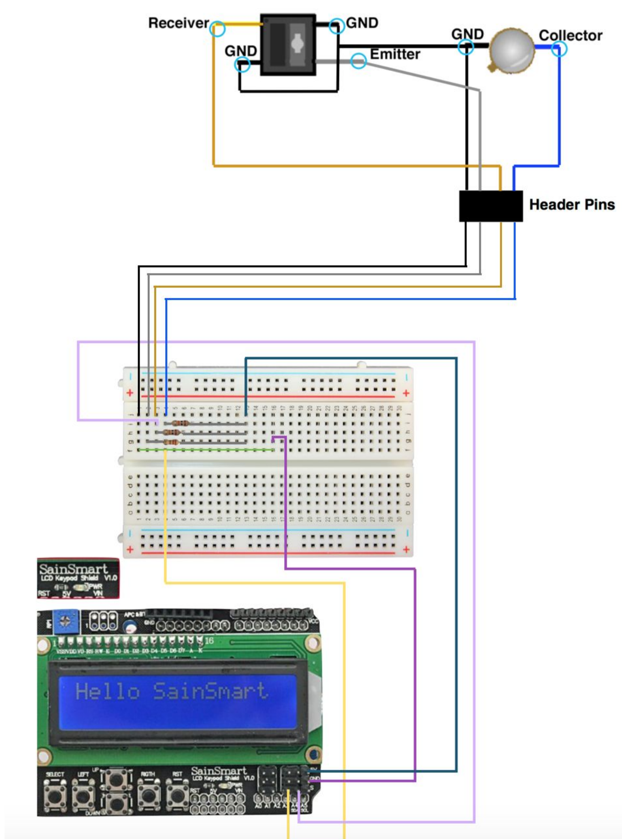


Figure 5. Arduino, Display and Sensor Apparatus

C. User Interface

Our design used an Arduino Uno microcontroller to calculate field dimensions and formulate instructions for the user. These instructions were displayed using a SainSmart 1602 Liquid Crystal Display. This display had two rows that could display 16 characters, and up/down/left/right, select, and restart buttons. This allowed the Arduino to display instructions to the user, and for the user give inputs using the buttons. The sensor array was attached to the Arduino, which was mounted to the LCD screen. This combination was powered using a rechargeable battery pack, allowing the user to recharge the interface between uses. The interface is controlled by the programmable Arduino, which was preloaded with our field layout program.

D. Coding

The layout of the field was directed by the Arduino, using a field layout program that the team designed to minimize the time spent on the painting/layout process. First, the user is prompted to choose from the standard field sizes: U8, U10, U12, U14, U16, and U19. Because our client wanted a method to lay out non-standard sized fields, the program includes an "other dimensions" option. This allows the user to input their own field dimensions. Because of the limited input buttons, the user inputs only the length or width of the field by choosing each individual digit by using up and down buttons to scroll through the numbers 0 to 9, and confirming each digit with the select button.

The field size is the only real input the user needs to supply- the program immediately begins instructing the user in their movements. This appears to the user as a series of instructions- to move to the lower left corner of the field, to align the painter, and start painting. All instructions followed the same format. For painter realignments that needed to be performed by the user, the program included a “press select to confirm” function that would pause the program until the user has confirmed that they have completed the necessary steps. For straight lines, the program knows the length of the line that needs to be painted, and displays to the user how far they have traveled, and the final length of the line. This updates as the sensor reads the rotations of the wheel, so the user can see in real time how much of the line they have painted. In addition, when the user is painting straight lines, the user can press the left button to make the program stop accepting inputs from the sensor. This means that the user can move the painter without their movements being counted as part of the length of the line. This is necessary in situations when the user needs to replace empty paint cans or if the user needs to realign the painter.

With these generic instructions, the user moves around the field, first painting the perimeter, then the left penalty and goal box, the centerline, and then finally the right field and goal box. The penalty arcs are painted after the penalty and goal boxes, with the user instructed to leave the stake as a marker and come back shortly after to paint the circle once the painter has been moved away. The same process is used for the center circle, with the circle painted after the centerline.

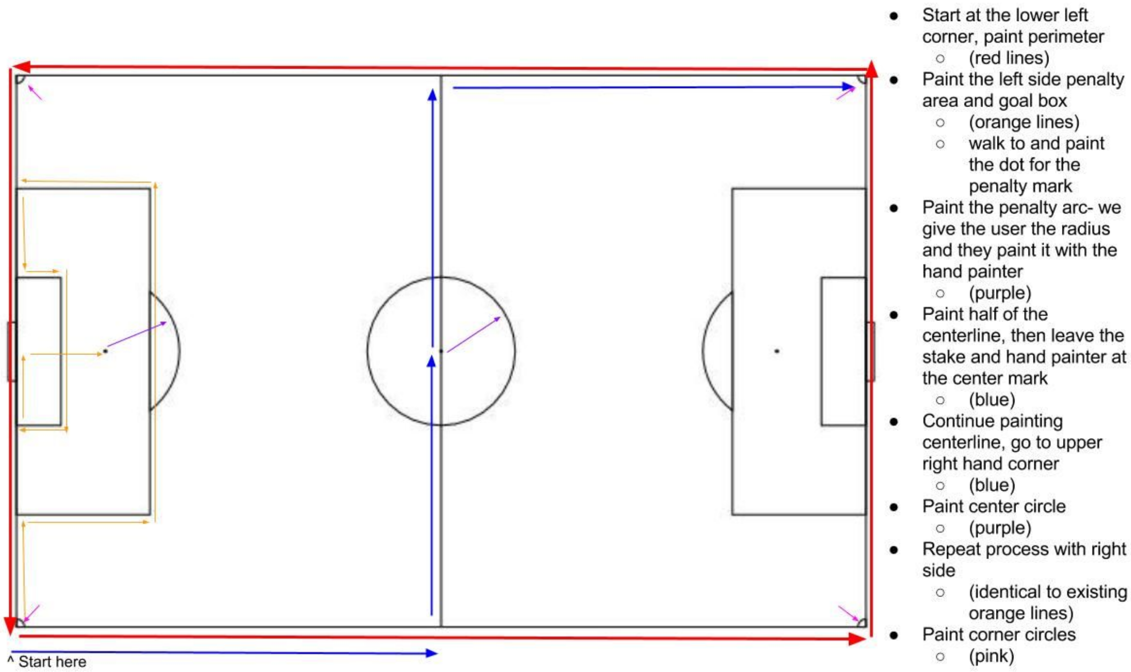


Figure 6. Field Layout Order

E. Physical Modifications on Painter

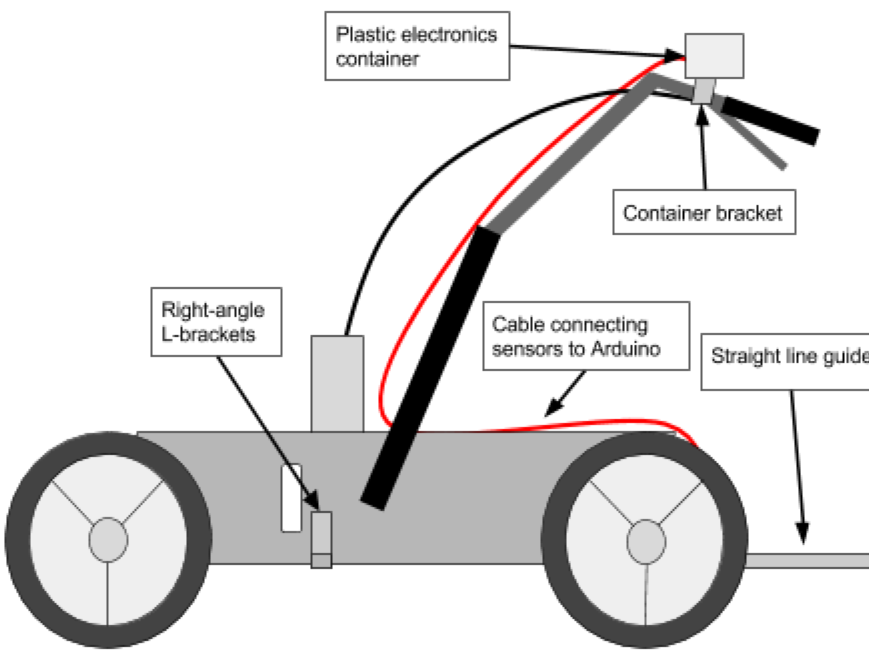


Figure 7. Painter and Physical Modifications

The first removable physical modification made to the painter was the attachment of a pair of large L-brackets to the side of the painter, lined up with the point at which the paint is sprayed. These brackets extend 8.25 inches in either direction, forming a straight line with the paint spray point directly between them. These brackets were intended to facilitate the creation of 90 degree angles. After the cart painted a line, the cart's chassis would be aligned so that one of the L-brackets (on the appropriate side) would cover part of the existing painted line. The next line produced would then be perpendicular to the existing line, within an allowable tolerance of .5 degrees.

In order to ensure that the lines produced by the painter were straight, a straight line guide was added to the back of the cart. A long, straight piece of metal was extended from the back of the cart so the operator could check its position against that of the line. This metal extension was fashioned from a light piece of metal to prevent bending. It was positioned in such a way that it would not restrict the operator's movement. The straight line guide was 18.5 inches in length when fully extended, but only 14.75 inches in length when rotated inwards. A bent piece of metal was attached to the rotating joint of the line guide to ensure a constant angle of 180 degrees.

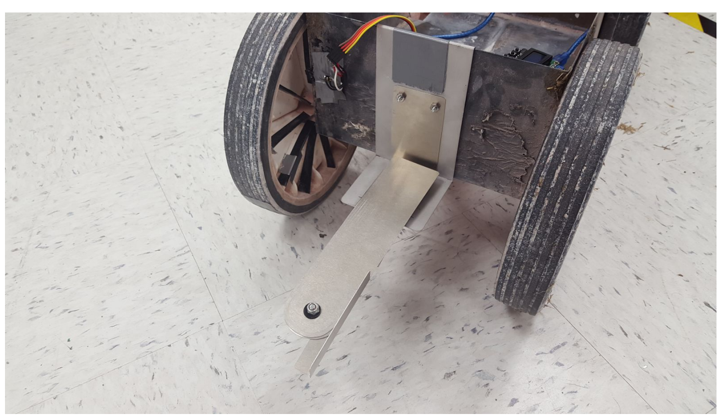


Figure 8. Straight Line Guide (Retracted)

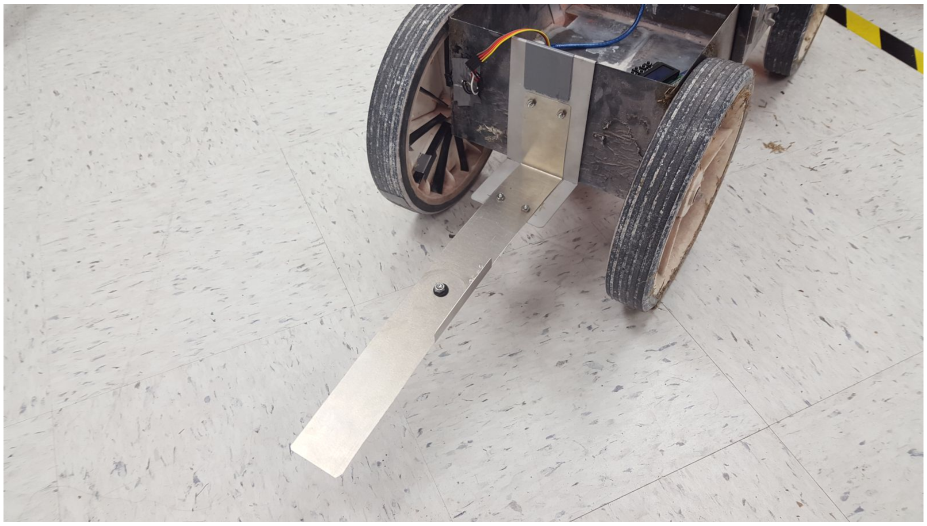


Figure 9. Straight Line Guide (Extended)

Next, the Arduino, display, and battery pack were mounted in an acrylic box attached to the handle of the painter. This container also served to protect the electronics from environmental exposures, but was equipped with access points for the Arduino serial port and the battery charging cord. While original plan was to 3D print the container, the 3D printers were experiencing technical difficulties. Instead the box was created by gluing together acrylic plates.

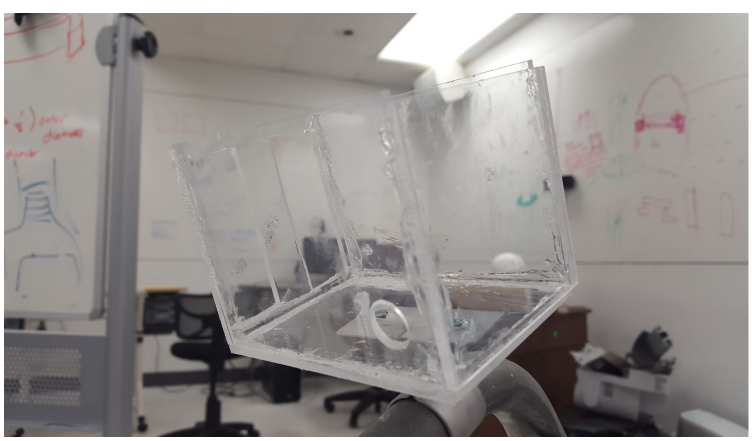


Figure 10. Electronics Box (Attached to Painter)

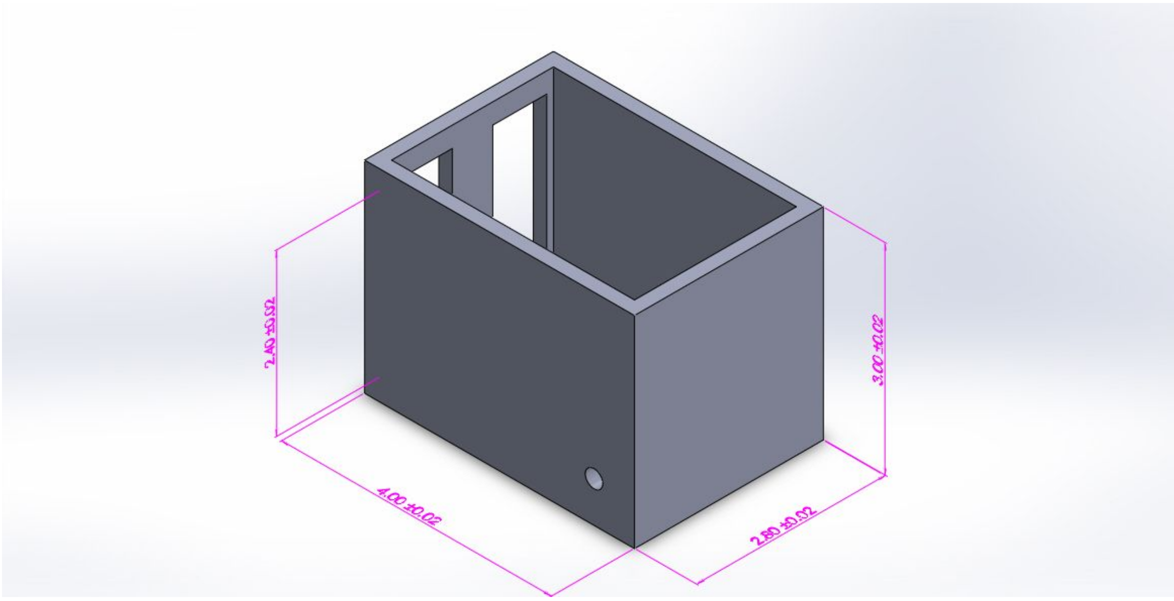


Figure 11. Electronics Box (Diagram)

A sheet metal omega-shaped mount was designed to hold the box in place on the handle. The diameter of the mount needed to be precise, because the safety of the electrical components in the box would depend on the tightness of the mount around the handle. Holes in the mount would allow for the acrylic box to be mounted using screws.

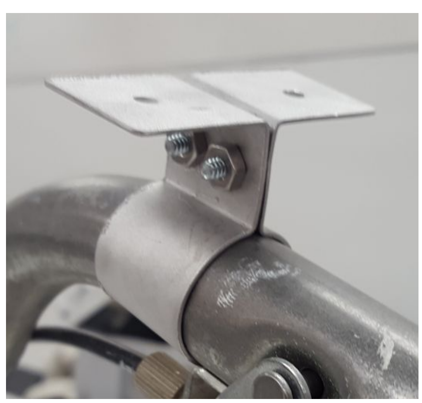


Figure 12. Mounted Omega Bracket

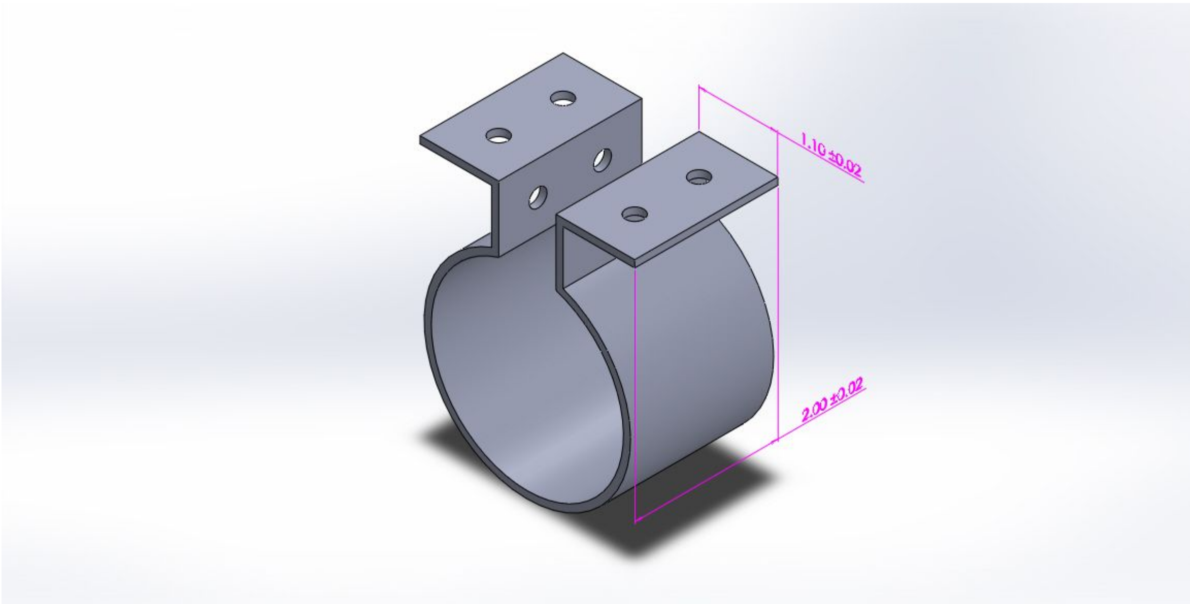


Figure 13. Omega Bracket (Diagram)

The final physical addition to the painter was the inclusion of a sensor mount that hangs from the bottom of the chassis and protrudes from a hole in the chassis' storage basket. It is intended to hold the reflectance sensor close to the wheel so it could read the markers attached to the spokes of the wheel. The sensor mount was painted black in order to prevent any interference with the sensor, which detected the difference between light and dark colors. However, the sensor mount was not used in our final prototype because it held the sensor too close to the ground. This would be problematic if mud or dirt covered the sensor. Furthermore, this positioning would make it much harder to access the sensor and clean it. The sensor was instead attached to the back of the chassis, which was higher above the ground. Another advantage of this placement was that it granted protection from the paint spray, eliminating the need to create a wind shield.

VI. Testing and Results of Final Prototype

The team ran a series of tests on the accuracy of the painting process. These tests were limited by several factors, most significantly, the amount of spray paint available. Time and budget constraints prevented the team from purchasing the quantity of paint necessary to paint an entire field. Instead the team tested the layout order and measurements by following the instructions from the display and drawing out the distances on a piece of paper. This created an approximately scaled drawing of the full field, and allowed the team to confirm that the layout of the field in the Arduino was accurate. Next, the team drew a series of short straight lines with the painter to confirm that the wheels were locked enough to ensure straight lines. These tests provided an estimated average angle displacement for straight lines, allowing the team to be check that when the display directed the user to paint a straight line, the painted line would actually be straight. These tests gave an average angle error of .0167 degrees (standard deviation .006). This is virtually negligible. Observation during testing confirmed some waviness within the line, but the endpoint of the line is accurate, so this will not matter in the overall construction of the field. This, too, can be improved with practice on the part of the user.

The team then tested the error of the right angle method. This was first done by testing the accuracy delivered from simply eyeballing a right angle from the side of the cart. Four trials gave an average angle displacement of 0.4 degrees (standard deviation .216), a very small margin that .would result in a maximum two feet displacement over a 100 yard line. While not optimal, this level of error would result in an acceptable field. Unfortunately, further tests could not be conducted because of lack of additional paint. However, the team is confident that the addition of the right angle guide would further reduce the error by a significant margin. From this inference, the team concluded that the final angle error would be low enough to result in an accurate field.

Table 4. Straight Line Angle Error

Error (in)	Total Length (in)	Angle Error
3.00	354	0.0085
4.87	213	0.0229
3.00	289	0.0158
2.00	102	0.0196
	Average	0.0167
	Standard Deviation	0.0062

Table 5. Right Angle Error Test Results

Right Angle Results
90.5
90.4
90.1
90.6
Average: 90.4
Standard Deviation: 0.21

Table 6. Success of Objectives

Objectives	Achievement
Be faster than conventional methods	From 3 or 4 hours to 1.5 hours with our model
Be reusable	✔
Be sustainable	✔
Be inexpensive	$94.30 (and painter)
Be precise	Yes (see testing)
Be scalable	“Other dimensions” feature
Be easy to use	Intuitive user interface
Be portable	Yes- attached to portable painter

Table 7. Success of Constraints

Constraints	Achievement
Under 3 hours for layout, (no more than 1.5 hours for striping)	✔
No more than 3 or 4 operators	1 operator
Low angle tolerance (very precise angles)	Yes (see testing)
No prior user training necessary	✔
Only uses paint for drawing lines (no chalk)	✔
Cost ~100 dollars, no more than the cost of a striper	✔
Line width must be between 2 - 4 inches	✔
No permanent modifications can be made to the currently field striper used	✔

VII. Recommendations

While our prototype design is fairly accurate in its rotation counting, there is some error, as a reflectance sensor simply detects a color reflected back to it. If the wheel of the cart were to become very dirty, it is likely that the sensor would not be able to pick up any of the black lines necessary for counting rotations. Furthermore, if any dirt or particulate matter covered the sensor, it would not be able to accurately detect the number of rotations. As previously mentioned, the team recommends replacing the reflectance sensor with a WLCA12-2 90 Degree Rotary Roller Lever SPDT Momentary Limit Switch. This rotary limit switch would use the existing spokes of the wheels to count the number of rotations. This sensor would have a much better defined method of measurement, giving it a much lower possibility for error.

There are some improvements that can be made to the coding portion of the design as well. Ideally, the team would have included a function that could subtract distance traveled, if the volunteer accidentally moved the cart backwards. However, this improvement could not have been made with the reflectance sensor. The addition of the rotary limit switch would allow the sensor to detect forward and backward motion, letting the user change the measured distance accordingly. We also could have further optimized the user interface. Additions like a numeric keypad for inputting field dimensions, or scrolling text instructions would make the user interface more friendly, but due to time constraints, the team chose not to pursue them during prototyping in favor of producing a fully functional design.

VIII. Conclusion

Overall, the tested prototype performed the necessary functions, met all the constraints, and achieved all the objectives. The design was chosen from several different alternatives based on the best of class and pairwise comparison charts. The final prototype costs approximately $94.30 to produce, with the hand painter using most of the budget due to high shipping cost. While the team could have tried to make a hand painter on their own, time constraints made buying a the pre-existing design the most feasible option. During the design process, the team decided to add on several physical modifications to the cart in order to allow for line straightness and right angle precision. These were achieved by attaching an extended piece of metal to the back of the cart and L-shaped brackets to the sides of the cart. The prototype will significantly reduce the amount of time needed to lay out and paint a field, and needs only one operator. While we cannot predict the exact time it would take to lay out and paint a field, we predict it would be similar to the time it previously took to only paint the field, roughly ninety minutes. Furthermore, our method is more autonomized, and can paint all types of standard field size, as well as fields of intermediate size. The addition of an Arduino takes any duty of calculation away from the user, with their only necessity to input the length and width of the desired field. This method also produces a field of necessary accuracy for practical use. While extensive testing was not possible, the team is confident that the added modifications will greatly reduce error and successfully produce a useable field.

IX. Acknowledgements

The team would like to thank Professor Srebotnjak for her invaluable advice as our advisor. Alex Alves played a major role in helping the team set up their Arduino and sensors, and Evan Kahn provided advice for machining and coding. We extend our thanks to Patrick McKeen for his recommendations and ideas throughout the design process. We also acknowledge the Harvey Mudd College Department of Engineering for funding our project. Finally, we would like to thank our liaison, Ms. Diane Flowers, for letting the team use her painter, and for doing everything she could to facilitate our work.

X. References

"Ameri-Stripe Aerosol Marking Stick." Soccer.EpicSports. Epic Sports, n.d. Web. 1 May 2016

"Arduino LCD KeyPad Shield (SKU: DFR0009)." *DFRobot Wiki*. DFRobot Wiki, n.d. Web. 10 Mar. 2016.

BrightCore. "How to Layout a Soccer Field Quickly." *YouTube*. YouTube, 07 Sept. 2010. Web. 10 May 2016.

Clark, Christopher. "Autonomous Passenger Vehicles." (n.d.): n. pag. E11 Mudduino Reference. Harvey Mudd College. Web. 1 May 2016

.

"Lucidchart." *Lucidchart*. N.p., n.d. Web. 01 May 2016.

MarkersInc. "Field Marking Equipment - Simple, Easy, Fast Field Lining System." *YouTube*. YouTube, 03 Oct. 2011. Web. 10 Mar. 2016.

"SainSmart LCD Keypad Shield For Arduino Duemilanove UNO MEGA2560 MEGA1280." SainSmart 1602 LCD Keypad Shield for Arduino Duemilanove UNO MEGA2560 MEGA1280 3D Printing, Arduino, Robotics. N.p., n.d. Web. 01 May 2016.

"Soccer Field Dimensions and Layout Tool for All Ages." *Trumark Athletics Field Markers*. Trumark Athletics, n.d. Web. 10 Mar. 2016.

"Soccer Field Marking Paint | Striping Lining Painting Soccer Lines." *US Specialty Coatings*. US Specialty Coatings, 2007. Web. 15 Mar. 2016.

Truscomfg. "New Way To Do Hash Marks And Border Your Football Field, Trueline Stripers New Models." *YouTube*. YouTube, 08 July 2010. Web. 10 Mar. 2016.

"WLCA12-2 90 Degree Rotary Roller Lever SPDT Momentary Limit Switch." : Amazon.co.uk: DIY & Tools. N.p., n.d. Web. 01 May 2016.

"2013 Version Arduino UNO R3 ATmega328P Development Module 2013 Version with Free USB Cable." Http://www.gearbest.com. N.p., n.d. Web. 01 May 2016.

XI. Appendices

10.1 Revision of the initial problem statement

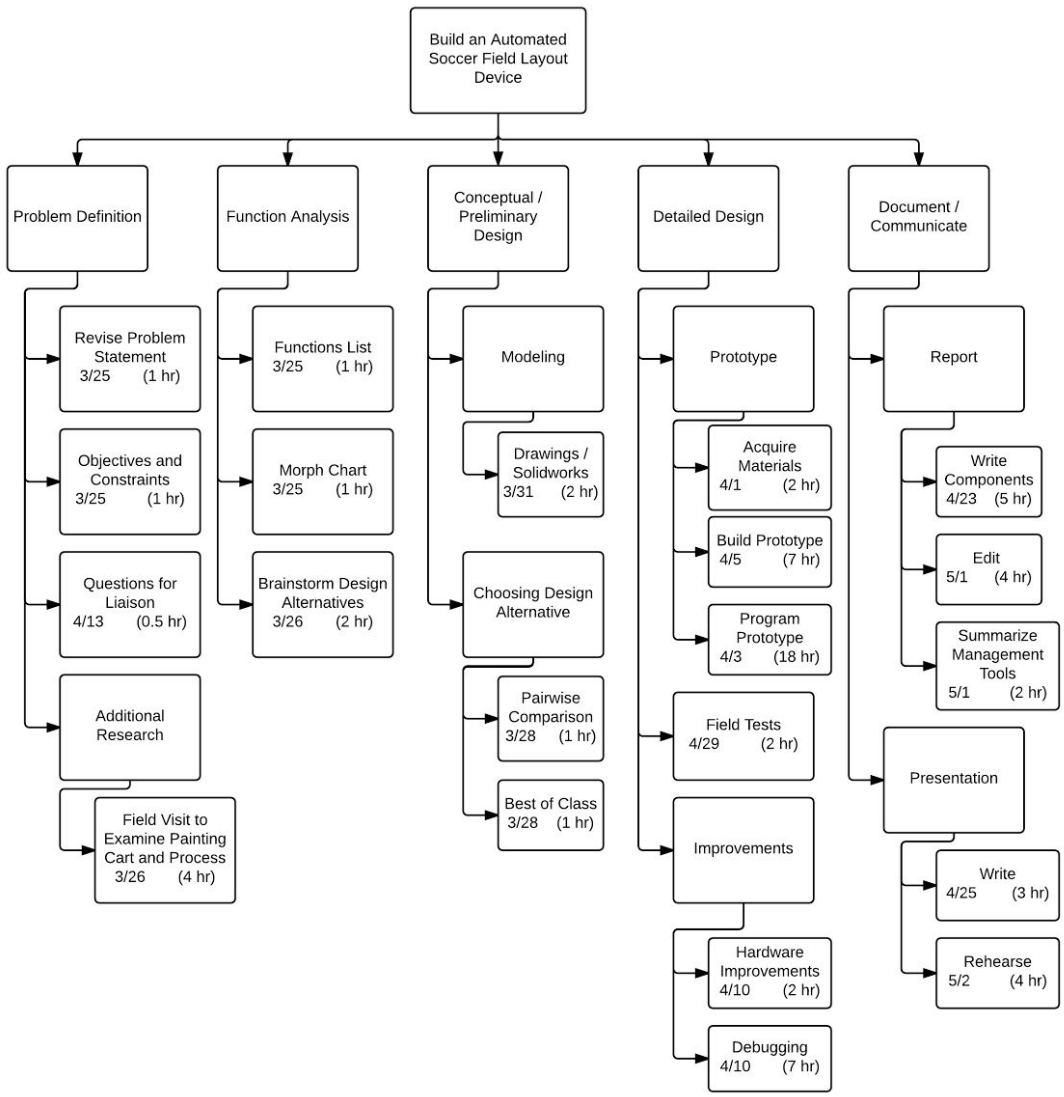
The team decided to revise the liaisons initial problem statement in order to remove any sort of error, bias, implied solutions and to clarify any ambiguity. Below is the revised problem statement.

Design a simple product that allows for the rapid and precise layout of different sized soccer fields using limited human input and traditional painting methods.

10.2 Project Management

The team used a work breakdown structure to organize the design process, which is shown below.

Figure 14. Work Breakdown Structure



10.3 Physical Modification and Installation Guide

The physical modifications are comprised of the following 5 pieces:

- Clear plastic electronics holder box
- Metal electronics omega-shaped bracket, composed of a circular segment with two perpendicular offshoots
- Folding rear line-guide
- Right-angle guides (2)

In order to attach the physical modifications to the cart, the following fasteners are required:

- 6-32 x ½ bolts (8)
- 6-32 nuts (8)

For attaching the right-angle guides and the rear line-guide, a hand drill with a 5/32’’ bit will be required.

Attaching the Electronics Box to the Handle

1. Attach the omega-shaped bracket onto the handle of the cart with the offshoots facing upwards.
2. Squeeze the neck of the bracket together.
3. Run two 6-32 x ½ bolts through the aligned holes in the neck and thread two 6-32 nuts on the other side. Tighten the nuts firmly.
4. Align the slot at the bottom of the electronics box with the holes in the top of the offshoot.
5. Run two 6-32 x ½ bolts through the slot in the electronics box and the holes in the offshoots of the bracket. Thread two 6-32 nuts onto the other end of the bolts and tighten them firmly.
6. You can now place the electronics inside the attached box.

Attaching the Folding Rear Line-Guide to the Back of the Cart

1. Rest the U-shaped part of the rear line guide on the back panel of the cart so that each edge of the rear line-guide is approximately 2.75’’ from its respective side of the cart.
2. There are 2 holes on the bottom of the rear-line guide. Line the drill up with those holes and extend them through the back of the cart.
3. Run two 6-32 x ½ bolts through the holes in the cart you created and their corresponding holes in the rear-line guide. Thread two 6-32 nuts onto the other ends of the bolts and tighten them firmly.

Attaching the Right-Angle Guides to the Side of the Cart

1. On the left side of the cart, drill a ⅛’’ hole 2.75’’ from the front wheel and ½’’ from the bottom of the cart so that the long segment of the right-angle guide is nearly flush with the bottom of the cart.
2. Insert a 6-32 x ½ bolt through the hole in the cart and the corresponding hole on the right-angle guide, then thread a 6-32 nut on the protruding bolt.
3. Tighten the nut, ensuring that the long edge of the right-angle guide is the one closest to the bottom of the cart.
4. Repeat steps 1-3 on the right side of the cart.

10.4 Bill of Materials

Table 8. Bill of Materials

Material	**Supplier**	**Quantity**	**Cost (USD)**
LCD Keypad	SainSmart	1	$13.99
Arduino microcontroller	Arduino	1	$9.99
Hand painter	Epic Sports	1	$25.39
NiMH 7.2 V Battery	BatterySpace	1	$15.95
Tamiya Battery Charger	BatterySpace	1	$18.95
2.1 mm power cable	Digikey	1	$3.33
400-point breadboard	Pololu	1	$3.38
QRD1114 Reflectance Sensor	Digikey	1	$1.31
BPW77NA Phototransistor	Digikey	1	$1.91
Resistors	Digikey	3	$0.10
Total Cost	-	-	$94.30

10.5 Arduino Code

See attached email for the full code.