Math RC Car Final Report

Team 6
Dorian Larios, Hai Joey Tran, Tu-Uyen Phan
December 10th, 2014
Lab Time: Friday 1:10PM – 4:00PM

Introduction

The objective of this project is to develop a way to entertain and engage fifth grade and sixth grade students while also teaching them an important science or math topic. The educational topic would reinforce or complement fifth grade and sixth grade curriculum.

This design had three requirements in conception: teach an educational topic as mentioned previously, use an Arduino microcontroller, and be interactive.

This report will describe the background and rationale for why the Math RC Car was chosen, how the Math RC Car was designed and implemented, and a conclusion on the effectiveness of the created design.

Background

From speaking with the clients, it was known that kids prefer to interact and learn through a hands on system. The kids spoke of enjoying simple educational games such as word bingo or jeopardy. Such games provided the children with the opportunity to interact with fellow peers and learn. Hence, when planning the concept and design of the project, it was important to consider the children's interest in personal interactions with the toy. The toy needed to accomplish the goal of being educational, simple but challenging, while still allowing the children to play with it on a physical level. The idea of the RC Car provided a solution to this challenge of motivating the kids to pick up a new academic concept without losing interest quickly. The car would allow them to manipulate the speed and direction it would travel according to the questions they answered correctly. In addition, the design of the car came with the idea allowing the children to manipulate the steering and speed of the car by controlling it with a smartphone. The clients spoke of technology often and would mention playing games on a smartphones. The concept of the car with the incorporated Bluetooth design would boost the design technologically by matching it to the standard game concepts the kids are used to. Inspiration for the design of the car came from researching the classic RC car toys online, such as Hot Wheels and Air Hogs[1,2]. The two companies are dominant in the RC car industries. By viewing samples of actual RC cars, it was easier to design a shape and image which would strike the children immediately as being an RC car. It also had to be ascetically appealing in the colors used and the size of it. For example, taking from Hot Wheels collection online, the shape of the Porsche car seemed simple enough to construct without sacrificing the "car image" of it. However, Air Hogs also provided inspiration in the interactive design in that it seemed like a tougher car. From the two cars, the final design would include making the car appear neat in structure, but tough in image.

Design Description

The design of the RC car had multiple aspects which were designed separately and then integrated together. These aspects include: Android app design, Arduino circuiting, and physical construction.

Android App Design

The app was coded in Java using the Eclipse IDE equipped with the Android SDK. On initialization, the app takes the user to a menu screen with three different options: “Connect To Car”, “Learning Mode”, and “Reinforcement Mode” (Fig. 1A). If the phone does not have Bluetooth connectivity enabled, then the user will be prompted asking if they want it to be enabled. If confirmed, the app will turn on the phone’s Bluetooth functionality. Then, the user must choose “Connect To Car” and choose the Bluetooth module from the list of paired devices. Once this is done and the phone successfully connects to the Arduino Bluetooth module, the other two menu buttons will be enabled, allowing the user to begin playing the game.

Both learning mode and reinforcement mode have two topics from which the user may choose from: programming and physics. In learning mode, the user is walked through a series of slides which teach the chosen topic. Both the programming and physics lessons have interactable widgets that the user can play with in order to further learn the concept.

In the programming learning mode, the user interface is created with a text box for the slides and instructions, a next button, a back button, an enter button, and an editable text box (Fig, 1B). During the programming lesson, the user is prompted to enter different basic programming commands into the box, such as initializing an object and calling an object’s method. The app is programmed to have the RC car respond to these various methods in order to demonstrate how methods interact with an object. For example, entering “RCCar.drive();” in the text box will make the RC Car drive at a medium speed. The methods are also equipped with different input parameters so the user can see how input parameters can change the behavior of a method. Another example, entering “RCCar.drive(slow);” in the command box will result in the RC Car driving at a slow pace.

In the physics learning mode, the user interface is similar to the programming learning mode interface but with the command box and enter mode replaced with a progress bar and joystick (Fig. 1C). During the physics learning mode, the user is taught about friction and torque. To teach about friction, the user is prompted to try placing differing amounts of weights on the RC Car and seeing how the weights affects its speed. To teach about torque, the maximal amount of speed is limited to differing levels and the progress bar is enabled to reflect the amount of torque being output. With this procedure, the user can see how when the car is allowed to go fast speeds, more torque is needed.

Unlike learning mode, the app’s reinforcement mode has the same general structure for both topics. In reinforcement mode, the user is quizzed about the chosen topic in order to further reinforce

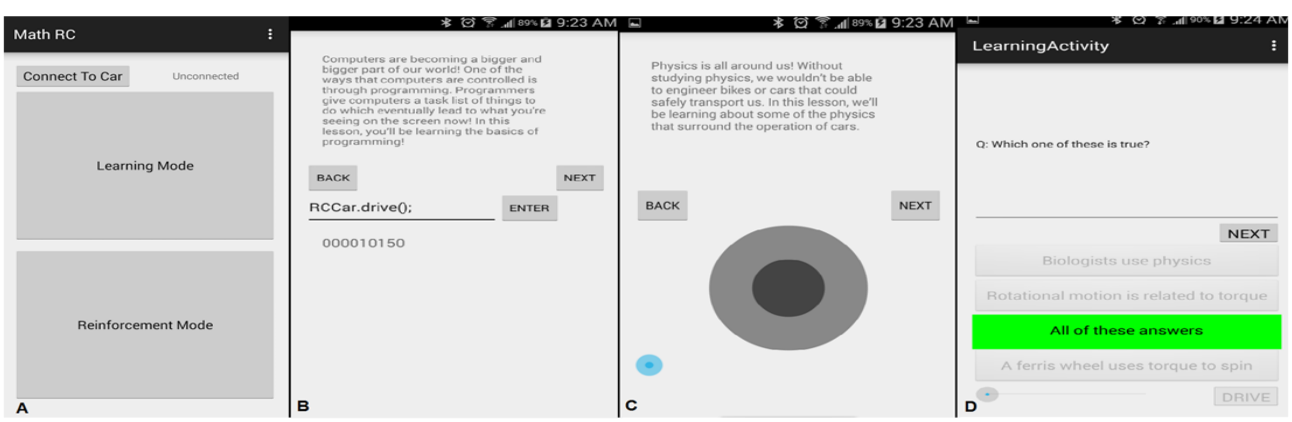


Fig. 1. Screenshots of different activities in the app. **A.** Menu screen **B.** Learning mode for programming **C.** Learning mode for physics **D.** Reinforcement mode quiz

taught concepts (Fig. 1D). If they answer a question correctly, a progress bar is incremented by 10%. If the user reaches 100%, then they will be allowed to begin free driving the RC Car using a joystick. Otherwise, the user will have to complete all of the questions in the quiz bank before being allowed to drive the RC Car.

The joystick widget is bound to a 20 by 20 grid, allowing the user to drag the joystick and reach values between -10 and 10 on the x and y axis. The communication protocol uses a series of nine integers sent at 3 Hz (Fig. 2). The first three integers represent the position of the joystick widget on the x-axis; the second three represent the position of the joystick on the y-axis; and the final three digits represent the max speed the motor is allowed to run on a scale of 1 to 255 (since the motor takes an analog input). In order to keep the message in integer form, a signed bit is used in order to denote sign of the x and y axis. When the Arduino receives the message, it interprets the serial message and multiplies the x and y messages by an arbitrarily chosen factor (in order to make values representative of the possible servo and motor values, ie 10 on the y axis will result in the highest acceleration possible). From there, the servo is turned until it reaches the value of the x message multiplied by the servo value factor and the motor speed is increased or decreased by the motor value factor until it reaches 0, 255, or the given max speed from the serial message.

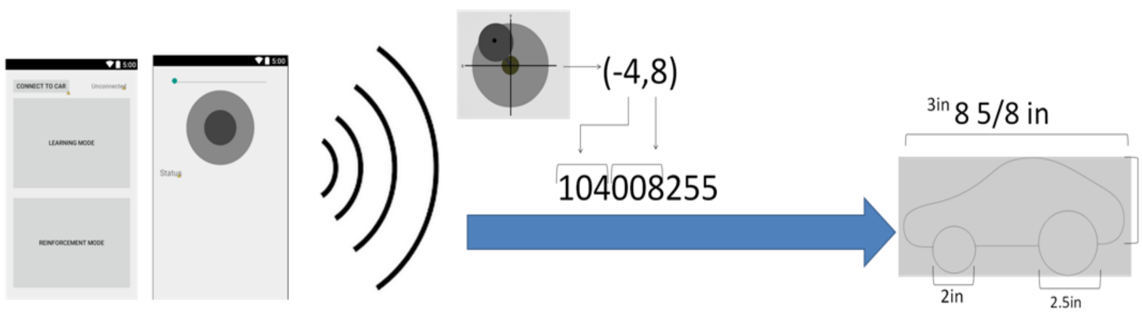


Fig. 2. Conceptual Model of Communication Protocol.

Arduino Circuit Design

There were three circuits used in this design: a motor circuit, a servo circuit, and a Bluetooth circuit.

The motor circuit was the most complex and had to be designed so that the current could be reversed. This was done using a double relay circuit in which two relays would be connected to the same digital port so they would be switched at the same time (Fig. 3A). When the relay signal was 0, the relays would connect the 9V battery to the motor in one direction. Once the relay signal was switched to 1, the relays would switch and enable an alternate circuit that would result in current flowing through the motor in the opposite direction, thereby reversing the direction the motors spin.

The servo circuit was relatively simple since the servo that came with the Arduino had an integrated transistor, allowing for it to be wired directly to a digital port and power (Fig 3B).

The Bluetooth circuit was also simple. The module was connected to ground and the 3.3V port of the Arduino. The “Tx” and “Rx” pins of the module (which would receive the serial code from the phone) were connected to the 0 and 1 (respectively) digital ports on the Arduino (Fig 3C).

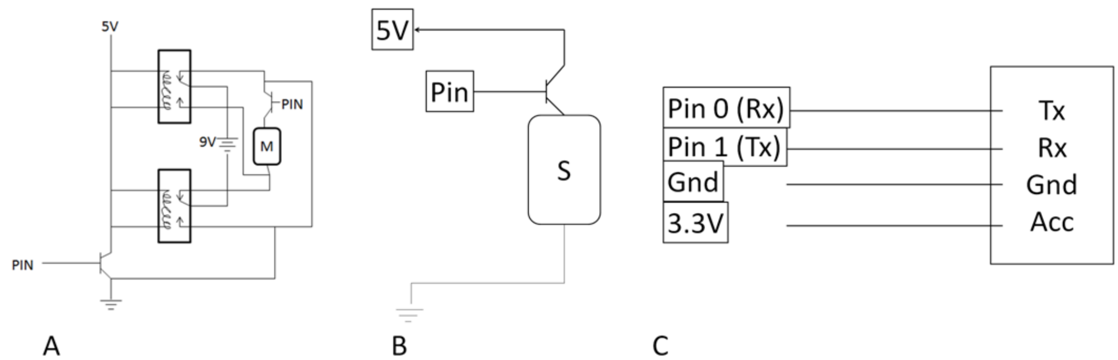


Fig. 3. Circuit diagrams. **A.** Double relay circuit to reverse motor **B.** Servo circuit **C.** Bluetooth Module circuit

Racecar Physical Design

The materials used for the physical design of the car included a wooden dowel, wheel bearings, thin wood, plastic, and two wheel mounted motors.

To create the front axle, wheel bearings were attached to both ends of a wooden dowel. Then, wheels were attached around the wheel bearings. The front wheels were created by laser cutting thin pieces of wood. The dimensions of the center hole were matched to the wheel bearings outer radius so that they could be connected. The Arduino servo was then mounted on top of the axle to allow for turning.

The chassis of the car was created in a similar fashion to the wheels. Using laser cutting, a thin piece of wood was cut to allow room for the Arduino unit, the front wheel and axle, and the motor-wheel units.

The motor wheel units were attached to the chassis with epoxy to keep them firmly in place.

The shell of the car was made using the remaining pieces of thin wood, again using laser cutting. The side cuts of the car were designed using Inkscape and fashioned to allow room for both sets of wheels and with enough room for the front wheels to turn. The top of the car was made from using plastic formed to fit the shape of the rest of the car unit. Refer to Appendix A for detailed orthographic drawing of design.

Design Trade-Offs

The original design for the Math RC Car used the motor that came with the Arduino package. Unfortunately, after trial and error with pulley systems and laser cut wheels, the motor proved too difficult to interface with shop-made wheels and pulley systems. For this reason, it was decided that a premounted wheel-motor unit would prove better than fashioning the Arduino motor into a pulley or gear system with shop-made wheels. Furthermore, the premounted wheel-motor units had higher torque capabilities than the included Arduino motor which would allow for a faster race car.

Evidence and Rationale for Design

The Math RC Car was chosen based on evidence from a variety of sources, including primary sources from interviews with clients.

During the conceptualization and ideation phase of this project, a visit was paid to the ASU Preparatory Academy. At this visit, particular attention was paid to answering two different questions: what were sixth grade age students currently learning, and what toys or activities interested and engaged sixth grade students.

During said visit, time was spent asking the clients explicit and implicit questions in order to detail what exactly could grab and hold their attention. When asked explicitly what kind of toys they would be interested in, they answered with a myriad of ideas. These ideas include: a rubix cube that lights up, something remote controlled, a portable copier, a time machine ,something smartphone oriented, and 3 dimensional novels. Although some of these ideas extend beyond the realm of feasibility, they give an idea on some of the interests of current generation kids. From this interview, it was decided that something remote controlled that could be controlled with a phone would be able to engage the students.

Another source of rationale comes from a study that analyzed the key principles involved with effective teaching environments. In the study, Judith Boettcher breaks the learning experience into four primary components: the student, the teacher, the goal knowledge, and the environment encompassing the learning experience. Boettcher finds that the best learning experiences are the ones with "multiple channels of communication, engagement, and collaboration" (Bettcher 2007) [3]. Environments that meet that criterion are the most well equipped to teach a wide range of users and different learning needs. The Math RC Car meets this criterion by offering a channel to teach, reinforce, and engage the students, which will encourage the students to learn and practice essential math and science topics while also giving them a reprieve from the standard learning environment.

An important aspect of the project was that it be educational. To make sure that the topics taught by the toy were applicable or complementary to relevant curriculum, research was also done on the sixth grade curriculum. This research was done through questions during the interview with the clients as well as common curriculum for the state of California. In the curriculum, it was found that kids were beginning to learn basic fundamental concepts to science, such as the scientific theory. The learning mode of the app is based on this concept and uses it to teach programming and physics. In the programming and physics lessons, the user is taught concepts through methods where they are instructed to try different options so that they can formulate hypotheses in their head and test them with the widgets given to them. Furthermore, programming is nowhere to be found on curriculums for middle school age students, which is why programming is included in the app. Programming is inherently a problem solving topic, which further reinforces use of scientific method and the understanding of programming at a young age to reinforce a strong foundation in an era where computer science is an ever-growing field.

Criteria Used to Choose a Design

In order to methodically compare different conceptual designs, an analytical hierarchy process (AHP) matrix was created. In the matrix is defined the most important criteria to the success of the toy. They are ranked against each other in order of importance. These criteria include (in order of priority): educational, engaging, user friendly, durable, affordable, practical, and compact. The exact weight of each of these criteria was determined by comparing each to one another. If one criteria was deemed more important than another then it would be given a number from 2-9 depending on how much more important it was. Conversely, the less important criteria was given the inverse of this number. After all criteria were compared, their scores were added up and divided by the total score accrued by all criteria. This percent was then used as the weight of the criteria.

After calculating the weights, each design was given a rating factor between one and nine for each criteria with a nine indicating that the design greatly satisfied that criterion and one indicating it poorly satisfied the criterion. The rating factor was determined by considering prices of materials, researched pedagogical methods, and information from the interview with the clients. For example, the RC car was given a rating factor of seven because the materials for the design were relatively cheap while the Math Math Evolution design was given a factor of four because accelerometers can be expensive.

Each rating factor was multiplied by the weight of its respective criteria. Each designs' weighted rating factor was then added together and compared. The RC car received the greatest total score followed by Educational Risk and then Math Math Evolution.

In the end, the RC Car was the best candidate because it met the given project weighted criteria the best. It was the most affordable and feasible design with the highest estimated educational value. Furthermore, it was the most engaging and user friendly candidate.

Criteria	Affordable	User Friendly	Engaging	Durable	Educational	Practical	Compact	**Total**	**Weight (%)**
Affordable	1	1	1/4	2	1/6	1/3	3	7.75	10.1
User Friendly	1	1	½	3	1/5	1	3	9.7	12.6
Engaging	4	2	1	2	1	2	4	16	20.9
Durable	½	1/3	½	1	¼	3	3	8.58334	11.2
Educational	6	5	1	4	1	3	4	24	31.3
Practical	3	1	½	1/3	1/3	1	1	7.1667	9.34
Compact	1/3	1/3	1/4	1/3	1/4	1	1	3.5	4.56
Total	15.833	10.667	4	12.667	3.2	11.334	19	76.7	

Table 1. Analytic Hierarchy Process Table. Lists criteria of project on top row and left-most column. Each cell describes the comparative importance of the row criterion compared to the column criterion on a

Considered Alternative Designs

Three different designs were considered before deciding on the Math RC Car. The other two designs included Math Math Evolution and Educational RISK.

The Math Math Evolution was designed as an educational spinoff to the popular arcade game Dance Dance Revolution. In Math Math Evolution, a Bluetooth enabled Arduino would sense the position of a user's legs through ankle-attached accelerometers. The user would then be prompted with a math question through either a phone or a TV screen. To answer the question, the user would have to select one of the multiple choice answers using their leg as a signal. For example, to answer "A" to a question, the user would have to step their foot forward. After getting a question right, the user would then be prompted with another question. In order to answer the second question, they would have to remember and chain the answer from the first question. Continuing the example, if a user got a second question right and "B" corresponds to a step to the right, the user would have to step forward and then right in order to answer question two. The goal of this game would be to get a high score and see how many answers they would be able to chain, resulting in a dance like movement. The hope of this design was to motivate the kids to answer questions through a fun and physically engaging game.

The other alternative design consisted of an educational spinoff of the popular board game RISK. It entailed a world map studded with LEDs at major towns and cities. Users would take turns choosing a city and would be prompted with a question about it using the Arduino LCD screen. If the player successfully answers the question, the city would light up with whatever color is designated for them. Whichever player has the most cities at the end of the game would win. This would engage the students through competition and would teach them about geography and historical facts.

Although these designs were feasible and somewhat met the criteria for the project, the Math RC car was deemed a better candidate and hence chosen.

Criteria	Weights (%)	RC		Math Revolution		Educational RISK	
		Rating Factor	Weighted R.F.	Rating Factor	Weighted R.F.	Rating Factor	Weighted R.F.
Affordable	10.1	7	70.7	4	40.4	6	60.6
User Friendly	12.6	9	113.4	4	50.4	5	63
Engaging	20.9	9	188.1	7	146.3	5	104.5
Durable	11.2	6	67.2	4	44.8	5	56
Educational	31.3	8	250.4	6	187.8	6	187.8
Practical	9.34	6	56.04	4	37.36	5	46.7
Compact	4.56	7	31.92	2	9.12	3	13.68
Total	100	777.76		516.18		532.28	

Table 2. Design Matrix. Each design was assigned a rating factor between 1 and 9 (9 meaning that the design satisfied the criterion very well). The rating factor was then multiplied by the criteria weights calculated from the AHP table. The weighted rating factors were then summed and listed at the bottom.

Design Implementation

Description and Pictures of Final Design

The final design reflected what was described in the section above. The chassis, shell, and front wheels were created using laser cut thin wood. Two wheel bearings were attached to the ends of the wooden dowel as an axle and the wheels bearings were placed inside the laser cut thin wood. The back wheels were attached using epoxy, and the Arduino and breadboard were placed snugly into the chassis.

Fig. 4. Pictures of Various Angles of Final Design

Tools and Methodology Used for Final Design

The side designs of the car were made using Inkscape which is a vector graphics program. Once the design was finalized, it was printed on regular paper and traced over onto the wood. Using the machinery available in woodshop, the side of the car was cut and sanded down accordingly. The band saw was utilized extensively in order to shape the hood part of the sides and for cutting out the round area for the wheels. Wood was chosen as the best material to use because it was durable and could be easily altered at any time during the design process. The wheels were made using the laser cutter because it needed to be perfectly round in order for the car to be able to smoothly roll. The wheels were cut from a type of wood found in the recycling bins at Tech shop. The base of the car was also built from wood and using the drill bit, the axle attached to the front wheels were carefully connected to the front of the base. The hood of the car was made from plastic forming the shape of the car. The design utilized for the side of the car was used to make several thick copies which were glued and sanded down together to provide the shape of the hood needed to form plastic over it. The ascetics of the car was created with vector graphics using the measurements of the final design. Once the graphics were created or converted to bitmap copies, the vinyl cutting machine was used to cut the images out.

Budget and Materials

The materials used included (not including the Arduino package): wood, an Arduino Bluetooth module, wooden dowels, wheel bearings, plastic, and motor mounted

Table 3. Project Budget and Materials

Item	Unit Price	Quantity	Total Price
Wood		1	$10.00
Arduino Bluetooth Module	$6.43	1	$6.43
Wooden Dowel	$1.00	2	$2.00
Wheel Bearings	$3.00	2	$6.00
Plastic		1	$6.00
Motor Mounted Wheels	$4.25	2	$8.50
	Total Budget		$38.93

Table 3. Budget and Materials.

wheels. The total price summed up to $38.93.

Testing Procedures

The design necessitated an iterative testing procedure in which different parts of the prototype were tested separately. Testing procedure included testing of the app with Bluetooth connectivity, testing of circuitry, and testing of physical design.

The app was tested using a Samsung Galaxy S4. As the app was developed, it was constantly tested whenever a new fragment (or screen) was developed. The fragment would be tested for functionality. For example, when the menu screen fragment was created, each button was tested to see if it would successfully transition the app to the next desired state.

The circuit was tested using Arduino code to signal different values to the components. If the components would run as expected with the Arduino test code, then it would be deemed functional. To test the circuits when completely integrated with the Arduino, a code was written to operate each component to see if the circuitry was assembled correctly.

The physical design of the race car was tested by adding component by component and doing analysis to see if the car would form with the other necessary parts. For example, the chassis would be cut out with the laser cutter and then the Arduino and breadboard would be placed on top of it to see if could adequately fit given the size and shape of the chassis.

Results from Demonstration with Clients

The final design was taken to a toy expo where the clients were able to play with the toy and were asked for an evaluation of the design. Each client was asked to give a rating between one and five for the following criteria: instructional, fun, innovation, craftsmanship, and ease of use. The data was then collected and statistically analyzed. The mean score for the criteria instructional, fun, innovation, craftsmanship, and ease of use were 4.5, 4.5, 4.5, 4.4, and 4.1, respectively. The standard deviation of the ratings were: 0.71, 0.71, 0.65, 0.84, and 1.08, respectively.

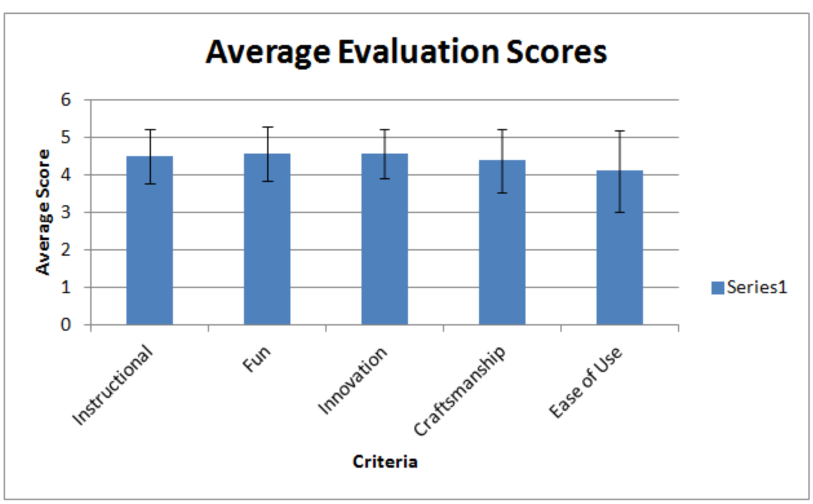


Fig. 5. Average Criteria Score with Standard Deviation. n = 48.

Competition and Profitability Analysis

Competitive toys that offer the same general product (an RC car controllable by Android app) range in price from $30.00 to as high as $170.00 [4-8]. Based on this and for use in profitability analysis, it will be assumed that this product can be priced at retail for the a little under the average price at $80.00. With this price, the product will have a profit margin of approximately $42.00 per unit. Profits can be further supplemented through the sale of additional test banks and lessons through the Android app store. In this business model, additional subject lessons and quizzes for the app could be sold at $0.50 to a $1.00, earning a pure profit since there are no explicit costs with the creation of additional software. This business model has great potential because it would be placed in a niche market for teachers that want to teach their students different topics or reinforce lessons during breaks. Refer to Appendix B for detailed profitability analysis calculations.

Conclusion

Summary

In this project, a Math RC car was designed to give middle school age students a break from the traditional learning environment while also reinforcing topics they have just. The design used an Android app, Bluetooth communication, and a physical car designed in a wood and metal shop. The demonstration of the design yielded positive reviews from the intended audience.

Learned Lessons

The design and implementation of this project shined light on common problems faced during the design process.

Time and responsibility management was one of the greatest problems faced during this project. Timeliness is key in creating a polished final product. In retrospect, this project should have had a schedule which would focus on completing a final design far before the toy demonstration day. This would allow plenty of time for revisions to the product if any flaws may come up last minute. Furthermore, division of responsibilities is a concept that would expedite the process of design. Every team member has a set of specialized skills and this knowledge should be utilized in order to ensure the processes are done most efficiently and with the highest quality. If this project was done again, responsibilities would be better divided amongst the team members based on preassessed skills and talents.

Recommendations for Full-Scale Production

The biggest key component for moving this product to full scale production would be ensuring the efficiency of the manufacturing process. The Math RC Car utilizes simple materials that all easily lend to manipulation and assembly. Full-scale production would need to find a bulk supplier for wood, wooden dowels, plastic, wooden dowels, and wheel bearings. Once these suppliers were found, manufacturing would require simple processes which could be easily facilitated using machinery to laser cut wood and assembly parts.

Past manufacturing and assembly, full-scale product would have a software maintenance aspect. This aspect would be easily handeable, requiring elementary knowledge of Android development. Further, past polishing the Android app, coding lessons and additional quizzes would increase the profitability of the product by allowing for further flexibility for educators.

References

[1] “Hotwheels Car Collector” Internet: http://www.hotwheels.com/en-us/collection/CFH86_Porsche-911-GT3-RS. [Nov. 20, 2014]

[2] “Air Hogs” Internet: http://www.airhogs.com/site/product/hyperactives-5. [Nov. 20, 2014]

[3] J. V. Boettcher. (2007) *Ten Core Principles for Designing Effective Learning Environments: Insights from Brain Research and Pedagogical Theory* [Online]. Available: http://www.environmentalconclusions.com/resources/Ten_Core_Principles_for_Designing_Effective_Learning_Environments-__Insights_from_Brain_Research_and_Pedagogical_Theory.pdf

[4] “VMW X6 App-Controlled Car” Internet: http://www.kohls.com/product/prd-1538693/bmw-x6-app-controlled-car.jsp?ci_mcc=ci&srccode=cii_17588969&cpncode=38-11339215-2&CID=shopping15. [Dec. 10, 2014]

[5] “Lamborghini S680 App-Controlled Car” Internet: http://www.kohls.com/product/prd-1540087/lamborghini-s680-app-controlled-car.jsp?ci_mcc=ci&srccode=cii_17588969&cpncode=38-11339251-2&CID=shopping15. [Dec. 10,2014]

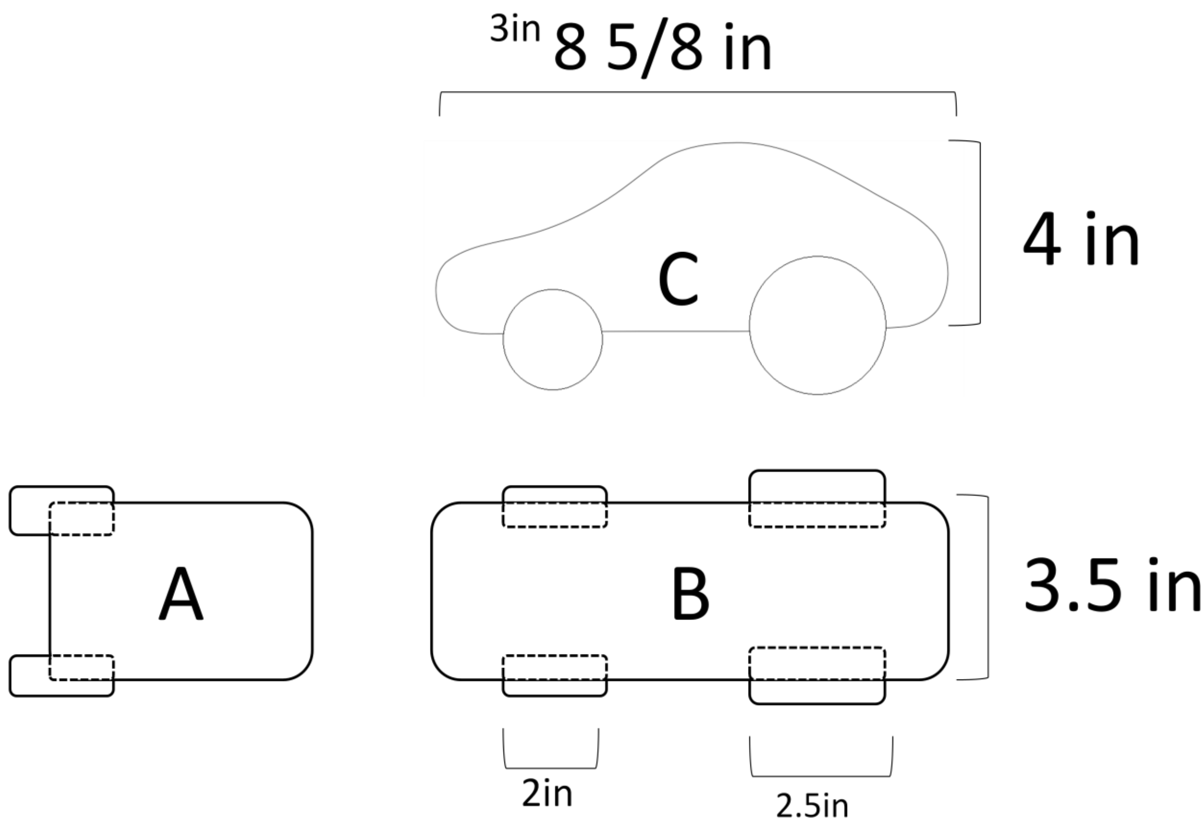
[6] “Remote Control Smart Link Ferrari Enzo” Internet: http://www.brookstone.com/remote-control-smart-link-ferrari-enzo-1-50?bkeid=compare%7cmercent%7cgooglebaseads%7csearch&mr:trackingCode=7B46DB2B-DD55-E311-A1FB-001B21A69EB0&mr:referralID=NA&mr:device=c&mr:adType=pla&mr:ad=62995871964&mr:keyword=&mr:match=&mr:filter=51320962143&gclid=CjwKEAiA-5-kBRDylPG5096R8mASJABqEdm4FIG-brJs7CfS4D1ORQOAkRvXl9KY9Sxwqih_0GlRwRoCTiLw_wcB. [Dec. 10, 2014]

[7] “Wifi Remote-Controlled Car – iOS/Android Video Audio App” Internet: http://iyouwebuy.com/wifi-remote-controlled-car-ios-android-video-audio-app?utm_source=google_shopping&gclid=CjwKEAiA-5-kBRDylPG5096R8mASJABqEdm4obAt-MDEf33I9DUz7dOqVPtrRHepATtGjP7r51hRWhoCHE7w_wcB. [Dec. 10, 2014]

[8] “Remote Control Speed Rader Car for iOS” Internet: http://www.ebay.com/itm/like/371102056910?lpid=82. [Dec. 10, 2014]

Appendices

Appendix A. Orthographic Drawing of Initial Design



A. Frontal view. B. Top view C. Side view

Appendix B. Cost Analysis Calculations

Revenue per Unit = $80.00
Cost per Unit = $38.93
Profit Per Unit = Revenue per Unit – Cost per Unit = $80 – $38.93 = $41.07
Profit Per Thousand = Profit Per Unit x 1000 = $41.07 * 1000 = $41070.00
Profit Per Thousand Apps Sold = Price of App x 1000 = $1.00 x 1000 = $1000.00