Deployment and Maintenance Plan:

To get LaserPi to the market is no simple task, at least not on the surface level. Since this project has a hardware component, software component, and a full web stack there are a lot of bases to cover before our end users would have a complete product. With careful planning and much thought behind the concepts, PySweeper has developed a viable deployment plan that would have LaserPi on shelves and scalable for years to come. I will go over all foreseeable costs needed to put LaserPi to market including all costs, reasons behind the choice of product/service, and options for future scalability if applicable.

The obvious place to start is with the hardware component of LaserPi. While embedded software is necessary to play LaserPi, without hardware there is no LaserPi! The interactive hardware is a defining component that the end user will demand is reliable, ready to use, and fun. A Raspberry Pi Zero W from microcenter currently costs $5 for an individual board, however when purchasing more than 1 the price rises drastically as the Zero model Raspberry Pi’s are intended for individual user applications and not business applications. Assuming Raspberry Pi doesn’t change their Zero model strategy in the future, we would need to change over to the Raspberry Pi Compute Module 3 Lite. These boards are easily available from Newark1 in quantities of over 100 for $23.75 each. This is, of course, assuming we do not convert anything to use cheaper tech. From a business stand point, $23.75 per board for a laser tag application is unreasonable. Thus, if possible from a software standpoint, we would ditch Python in order to go with Lua for use on a much more reasonably priced NodeMCU (Lua board) at $3.55 per board (50 – 199 piece order)2. We also would need to port our code over from Python to Lua, likely using python-lua translator32. However, likely this will have some cost to it as well even though Lua officially states it is free to use for businesses at no cost. Using statements ratio, Python accomplishes 6 lines of code with what C does in 1 (we are using C as the information is available and it is comparable to LUA).33 This means we have to pay 6 times what we did for the Python code in terms of statement ratio. How to figure out the cost of the original Python is tricky, the code took about 8 hours of total time with debugging to develop and at an arbitrary pay rate of $30 per hour this would cost about $240. 6 times this cost is $1440 and that seems high but porting code is a difficult task. No one wants to port code unless they need to. This however will accomplish the task. With the brain of the hardware covered, we move onto the HAT (protoboard), IR sensor (sender and receiver), press buttons, LEDs, capacitors, resistors, and solder. The protoboard will cost $3.50 (over 100) a piece3, the IR sensor and receiver will cost $3.49 for a set of 25 or about $0.14 per set4, for the switch the cost is about $0.23 per piece (over 100)5, LEDs will cost about $0.21 for a red and $0.21 for a green (over 100 each)6,7, capacitors will cost $0.13 per piece (over 100)8, there are 3 different types of resistors needed the 10 kΩ will cost $0.02 per piece (over 100)9 while the 220 Ω will cost $0.04 per piece10 (over 100) and the 330 Ω will cost $0.01 per piece11, and the solder is $30.08 per pound12. Notice that many of these parts are extremely inexpensive and considering how little solder is actually needed for the board, 1 pound may solder well over 1000 boards and thus the price becomes negligible on a per unit basis. The total cost per unit for these components then comes to about $4.49. With the NodeMCU board the total cost then comes to $8.04 per unit to this point. Now the user needs portability, and thus each unit must have a battery pack. To add a battery pack isn’t as simple as going and getting a cheap pack. The battery should be rechargeable, last awhile, and not blow up. Since the hardware is all about user experience as well, we chose to indicate to the user when the battery is running low. Clearly this isn’t cheap but we don’t want a lawsuit and we need reliability, so the battery pack will be a highly rated pack from Miuzei that has a UPS (won’t experience interrupts), an onboard LED battery level indicator, works for 3800mAh (about 9 hours constant use), and is expandable (can add additional battery packs and a fan). The Miuzei Battery Pack expansion board costs $18.99 each13 but with over 100 orders with a deal for future order we would hope to negotiate the price down to below $15.00. The total running cost is now $22.54. Only one more aspect to the hardware must be added, the look and feel. Let’s be honest, everything up until now simply needs to work and the user won’t care otherwise but the outside of the LaserPi gun must be clean and crisp. Which is why we will 3d print guns. The design can be printed for $918.75 for 100 units or about $9.19 per unit14,15. This completes the hardware portion of deploying LaserPi with a total cost of $31.73.

With the hardware component of LaserPi planned for deployment, attention can shift to getting the web interface up and running. This is the next main component of the user interaction with LaserPi. A user will have the ability to keep track of all records in a global database but in order to start a game that will store their records they must have a database for their individual system. The web interface can easily be the same across all these servers and a single domain can be split into subdomains for each set of LaserPi’s purchased. Subdomains can be automated to save man power and a single MySQL server can be separated into separate databases for each LaserPi. Let’s start with the MySQL server. After much research on costs and efficiency, the best options for commercial MySQL databases are all on Amazon Web Services (AWS). AWS offers a pay by the hour plan based on the size or users can pay upfront for a lower rate. We can change the plan on AWS and thus our model is flexible and scalable to serve our needs. LaserPi settled on the t3.small MySQL database plan that has a cost of $5.73 per month (3 year upfront payment, operating 24/7) making the total yearly cost $68.7616,17. Based on our calculations, this would provide us with enough for 209,715 LaserPi databases. This is based upon our own database using less than 10 kb of storage and the t3.small MySQL database providing us 2 gigs of memory. Again, this is AWS database plan is changeable but if more databases are needed we would, of course, have to pay more. This price is very low compared to the prices Microsoft, Oracle, and even the often considered inexpensive MySQL.com databases (which costs minimum of $2000 per year18). AWS offers affordable scaling that should fit LaserPi’s database needs for the foreseeable future. Again, at $68.76 a year this is affordable but not negligible in cost. Luckily, web hosting for our domain is very inexpensive in comparison. AWS offers hosting through their service called Lightsail, however this option is more costly than others explored. Preferably, we would keep all web stack applications to AWS but cost is always an issue thus LaserPi will be hosted by HostGator. The Hatchling Plan offers a single domain with unmetered bandwidth and unlimited email addresses. Of course, this supports unlimited subdomains, MySQL database interaction, PHP, HTML5, Javascript, and of course 24/7/365 support. With these services we can ensure a very high reliability for our end users with no fuss. The total cost of web hosting is $2.75 per month or $33.00 per year19. This hosting plan offers affordability and reliability but also very much scalability. This concludes the web portion of LaserPi with a total yearly cost of $101.76 per year.

Finally, the last deployment cost will be purely promotional. LaserPi will host our own Embedded Systems Convention here in Lawrence at the University. This will operate as a Raspberry Jam would, an independently organized event such as a workshop, show-and-tell, speech, symposium, etc based around Embedded Systems. LaserPi will host a convention to not just promote LaserPi but to share ideas around all embedded system based projects. Now, hosting the event is free and this is true but there are still costs for a venue, costs to set up that venue with tables/chairs, and of course extras such as food and beverage. While exploring options for hosting our own convention, it didn’t take long to come to the conclusion that KU is the perfect location. Amazingly, students can rent any room, outdoor space, or café available for free. That’s right, I could actually rent the entire Debruce Center (this includes the courtside café, mezzanine, and rest of the center along with the plaza) right now and host an air guitar competition for $0.00. While this is absolutely mind blowing, we will assume that for deployment we are no longer students but are alumni and thus associated with the University. The Jayhawk room in the student union is 2453 Square Feet and can hold 40 tables conference style. This is a very large venue and only costs $130 to rent20,21, we will add in $100 to get tables/media rentals for everyone involved, and $500 for food/beverage and advertising in the form of small snacks, water, soda, as well as email, internet, and poster promotions. We can do a lot of the setup ourselves and have volunteer’s handle what we cannot, this brings a total cost to $730 for the event. With this event to promote LaserPi and the embedded systems community on a whole, we will be well on our way to a full scale deployment. Note that I did not include the potential for us to recoup costs by charging for booth space. That is a possibility but would be calculated in to expenses for deployment purposes. While hosting our own convention is very viable and would go far to aid our cause, we understand the need to attend a more established convention as well. There are seemingly endless options for conventions to go to but we will attend Embedded Systems Conference (ESC) at several locations as this appears to be a very well established convention organization. ESC Silicon Valley will cost $59 per square foot of booth space, we will use a standard 4 foot by 2 foot table so we need 8 square feet. This will leave us with a total cost of $472 to attend ESC Silicon Valley23. The others are ESC Minneapolis and ESC Boston. ESC Boston will cost $54.25 per square foot for a cost of $434 and ESC Minneapolis actually takes you to ESC Boston page so we’ll assume it’s the same cost24. This give us a total of $1340 for setting up booth space at the conventions. Unfortunately it will cost money to our items, mainly the table, shipped. Uship estimates about $220 to ship to San Jose(Silicon Valley)25, $165 for Boston26, and $133 for Minneapolis27. This a total of $518. Now we have to consider flights as well. I found round trop flight prices for the days of each expo as such: $326 per person to San Fransisco28, $218 per person to Boston29, and $197 to Minneapolis30. We will likely send two people so the cost will add up to $1482. Hotels will run about $200 per night on average across all three cities and a rental car will be $60 per day as per skyscanner31. We have to stay 6 nights total, so this will add up to $1200 + $360 for a total of $1560 for hotels and rental cars. Conventions will provide food but let us add in $100 per day assuming there will be some food costs even if the conventions provide food, as well as some other expenses that might come. In total to attend these three conventions: $1340 for convention costs + $1482 for flights + $1560 for lodging and rental cars + $600 for food and extras = $4982. This actually isn’t too bad considering the promotion we could get for the product. Under $5000 for 3 cities in 3 unique areas of the USA is reasonable. This gives a total cost of $730 to host our own convention + $4982 to go to other conventions for a total of $5712 to promote the product for deployment.

With this deployment plan, we have covered enough bases to get off the ground and get a product into user’s hands for several years. To port the code from python to LUA, the cost is $1440. The yearly cost for web services is $101.76. The cost for promotion/conventions is $5712. Assuming we produce 400 LaserPi’s for launch hardware will cost $12,692 to be produced and shipped. Thus, our total costs would come to $19,945.76. For a small business, under $20,000 to deploy a product is very reasonable. In fact most tech based products on KickStarter get funded while asking for $20,000 to $99,99922. So in total, we are keeping costs low and making deployment very viable. However, 400 units is rather low for an entire year. Our profit margin is low under this strategy so we will likely need to produce many more units. Producing 1000 units would allow us to market towards businesses and individuals as well. If there is serious interest from places like Dave and Busters or Round 1 then we may need to produce more than even 1000 but until we are established let’s say we need 1000 LaserPi units. This would cost $31,730 and bring total costs to $38,983.76. This isn’t great compared to the last number but again under $40,000 isn’t as bad as it could be. There are likely a few discounts that we may be able to procure when purchasing very large quantities of several products from a distributor but we didn’t count those costs as likely they would be offset by unforeseen costs that would add up as well. However, this plan is very thought out and hit on almost everything we could think of. Maybe a deployment will be in our future and hopefully this document can aid us if that time should come.

When deciding how to best maintain LaserPi several key strategies came to mind. Since the hardware is such a crucial component we will need to ensure we have a maintenance/repair package available to our customers for a monthly or annual fee, especially those in the business sector. Also, we will need an extensive troubleshooting documentation as well as an employee to handle any issues a customer may face. In order to keep our website zippy, especially when traffic on our domain increases, we will provide options for subscription to the web service. With these three maintenance pillars, we have a plan in place. Website, Repair, Troubleshoot, or WRTs for short, will be the defacto first year company motto.

The website/domain/servers have already been discussed on a yearly basis. The yearly cost for all web services is $101.76. This is great for first year and likely for the future 5 to 10 years. Again, we will have space for over 200,000 databases which will be more than enough. The more important part here will be hiring developers to maintain all of this. We will plan to hire two developers, one for the development on the Pi/Lua board code and one for the web interface primarily however they will share duties. Each will get a salary of $65,000 for a total of $130,000 per year (I didn’t source it but this is based on Kansas City area starting salaries. As costs increase in the area, we will of course provide a form of health insurance. This cost comes from a stipend of $300 per month for the employee to spend on healthcare as they see fit (source: This is actually how team member Jeff’s last job worked). This roughly concludes the web maintenance portion with a total cost of $137,301.76.

With any product that relies on hardware repairs will always be a major portion. This is a tricky area to tackle since we must weigh the cost of actually doing a repair vs outright replacement. The cost of repair will vary by component and what’s wrong with the component. If we have to hire or contract out repairs the repairs then we have to pay for that cost as well. The more time goes on, the more repairing small components due to some small manufacturing/user error and the time spent on figuring out what is faulty doesn’t make any sense. More likely, we will return to manufacturer or sell off faulty components and simply offer replacement parts. We have no way of currently projecting how often people will need replacement parts. Research has not revealed any results for average fail time so I will attempt to find a fail time for Raspberry Pi’s. Research is hearsay and should thus be taken with a grain of salt but a stack exchange forum reveals that Pi’s have a 1% failure rate after ALL the years that Pi’s have been on the market34. That’s a tremendously business friendly rate of failure so we will assume at least 1% of LaserPi units will fail per year. With 1000 units produced this means at least 10 will fail. This will cost $310.73 to produce 10 brand new LaserPi units, as listed above each unit costs $31.73. This, for now, concludes the repair portion of the maintenance plan.

The final component for cost will come in the form of troubleshooting. This is a large cost as essentially we will need to spend time and money creating troubleshooting protocols/Standard Operating Procedures for employees to follow. Of course, LaserPi would at least need one member to handle the troubleshooting department duties. While the duties are simple: returning emails, answering phone calls, and other clerical duties, the person behind these duties must be able to answer the questions presented to them. These questions could be extremely specific or technical and thus couldn’t have been predicted ahead of time. What I’m getting at here is that the troubleshooting employee for LaserPi will have to be another technical person and thus will cost more money than a typical per hour phone employee. A technical support engineer in the Lawrence, Kansas area gets paid somewhere around $37,000 to $73,000 depending upon the extent of the position35. At LaserPi, we will offer on the lower end, $40,000 per year to start. We aren’t heartless so of course this includes insurance again, which will bring about another cost of $3,600 per year. This is normal for a startup but with time and increased responsibility the pay could increase. Final costs for troubleshooting then would be $43,600 where the cost of creating troubleshooting protocols and SOPs is calculated into the employee salary.

While paying for things is important, LaserPi is also be concerned with the bottom line. This is why here I will outline some of the potential methods of recouping our costs. For web services, we will have two tiers of packages: a consumer level package or a business level package. The consumer level package will consist of a small database allocation and subdomain at a reasonable cost of $1 per month. The business level package will consist of a larger database allocation, a subdomain, and the ability to customize their stats and look and feel (prebuilt packages from our dev and ability to request additions) for a heftier price of $100 per month. The price is based upon a few factors. Business’ will charge a lot to play the game of Laser Tag and thus will reap in profits, they will need more support than a message saying “We’ll get back to you as soon as possible”, and they will likely cost more than anticipated as the database will be constantly queried for potentially 80 hours per week +. For the repair/replacement, all LaserPi’s will come with a 30 day warrantee but we will offer again a consumer level package and a business level package. The consumer and business packages will be much the same, a conditional 30 day warrantee. However, since the consumer will accrue much less return volume than a business would, a consumer level repair/replacement package will only cost $30 per year while the business package will cost $3000. This is calculated based upon the expected overall sales in one year for hardware. The consumer cost is just under 1% of the cost of 1000 LaserPi units, and the business cost is just under 10%. This is, of course, assuming we sell 1000 LaserPi units but also represents the floor for our warrantee package. Should we happen to sell more than 1000 units we will use the 1% for consumer and 10% for business calculation based on the expected sales for the year or quarter, rolling average36 (based upon a six-sigma idea for warrantees). This is a more sustainable business model than simply having the cost fixed. If larger businesses with multiple locations invest heavily, we can incentivize them to purchase repair packages by offering a discount, say 1% off for every 100 units purchased down to a minimum of 5% of expected sales. Ideally, we would be right around the draw even point for the cost of hardware vs the cost of repairs and/or replacements. In the area of troubleshooting, our hands are tied. We must have a product that works at all times, our company reputation is on the line in this sense. Thus, troubleshooting must come free for the consumer. For the business end, we expect much more frequency and demand on the troubleshooting end. While we will still offer this service for free to those with a repair package, we will not offer it for free to those who choose to opt out of that package. Again, this is normal as larger companies tend to have their own repair personnel. In this event off no repair package, troubleshooting will be done on a pay by time basis, $50 per hour. It’s a large price to pay but if they don’t have a repair package, a business is likely to handle troubleshooting themselves as well.

With WRTs, website repair troubleshoot, costs and some money coming in to offset these costs, LaserPi hopes that they have a through, sustainable, and extensible business model for maintaining the LaserPi product and brand. As we go forward through the year, pending success, we may be able to add to our personnel and product line. Currently however, we are looking at a cost of at least $181,212.76. A very large cost to run a business for year at about $500 a day. Considering there are other aspects, such as facility costs and taxes, I would expect this number to be even larger. Thus, until we are able to establish the company we would be in the red for quite sometime. Hopefully, overtime we build a solid reputation and find clever ways to lower costs while raising profits. In the meantime, we have outlined our yearly maintenance plan as meticulously as possible.

Product Sources:

1. <https://www.newark.com/raspberry-pi/rpi-compute3-lt/raspberry-pi-compute-module-3/dp/02AC9992>
2. <https://www.electrodragon.com/product/nodemcu-lua-amica-r2-esp8266-wifi-board/>
3. <https://www.newark.com/microstack/microstack-protoboard/prototyping-board-raspberry-pi/dp/83X9200?ost=raspberry+pi+protoboard&krypto=E7%2FMWXE%2BRWlcIQz3%2FZCGX15e0HYEM8%2FNydStp3sIOD66EXOFgGfhRt5by94v4zRcZrrWxiduKyGa3boCjN6YFw%3D%3D&ddkey=https%3Aen-US%2FElement14_US%2Fsearch>
4. <https://usa.banggood.com/50pcs-5mm-940nm-IR-Infrared-Diode-Launch-Emitter-Receive-Receiver-LED-p-1205409.html?rmmds=search&cur_warehouse=CN>
5. <https://www.newark.com/omron-electronic-components/b3f-1000/switch-tactile-spst-no-50ma-though/dp/36K7138?st=button>
6. <https://www.newark.com/vcc-visual-communications-company/4304h1/led-red-t-1-3-4-5mm-6-3mcd-650nm/dp/93K6988?st=LEDs>
7. <https://www.newark.com/vcc-visual-communications-company/4304h5/led-green-t-1-3-4-5mm-10mcd-563nm/dp/93K6989?st=LEDs>
8. <https://www.newark.com/avx/sr211c104kaa/cap-mlcc-x7r-0-1uf-100v-radial/dp/95W2705>
9. <https://www.newark.com/multicomp/mccfr0w4j0103a50/carbon-film-resistor-10kohm-250mw/dp/58K5002?MER=bn_browse_1TP_MostPopular_1>
10. <https://www.newark.com/multicomp/mcf-0-5w-220r/carbon-film-resistor-220-ohm-500mw/dp/38K0281>
11. <https://www.newark.com/multicomp/mcmf0w8ff3300a20/metal-film-resistor-330-ohm-125mw/dp/58K3926>
12. <https://www.newark.com/kester-solder/14-5050-0125/solder-wire-50-50-sn-pb-214-c/dp/00Z529?MER=bn_para_1TP_LastViewed_1>
13. <http://www.miuzeipro.com/product/miuzei-for-raspberry-pi-3-3-model-b-battery-pack-expansion-board-power-supply-with-usb-cable-2-layer-acrylic-case-for-raspberry-pi-3-b-2-model-b-mc12-1/>
14. <https://www.thingiverse.com/thing:2908555>
15. 3dPrintPrice.PNG (Documentation folder)
16. <https://aws.amazon.com/rds/pricing/>
17. AWSDBPrice.PNG (Documentation folder)
18. <https://www.mysql.com/products/>
19. <https://www.hostgator.com/web-hosting?utm_source=google&utm_medium=brandsearch&kclickid=f0e7980f-e183-409f-9e98-e2aecc5d4581&kenshoo_ida=Host%20Gator%20IDA&adid=340457678313&utm_term=host%20gator&matchtype=e&addisttype=g&campaign=1748611327&adgroup=69935979793&gclid=Cj0KCQjw4-XlBRDuARIsAK96p3A9C0_fNtw7Pl3G6pDuge5Jgn6qWSDkr7jXyyAGeYW1oDisV6hQL-IaAuJqEALw_wcB>
20. <https://union.ku.edu/rates>
21. <http://kumuprdemswb.home.ku.edu/emswebapp/LocationDetails.aspx?data=7cZwpiiplNuE8sYZ9UAYwBVoBG5zq5MxsXe3g2DTp9Y%3d>
22. <https://www.kickstarter.com/help/stats>
23. <http://escsiliconvalley.com/become-an-exhibitor>
24. <https://esc-boston.com/become-an-exhibitor-esc>
25. <https://www.uship.com/ship/household-goods/furniture/pricing/4iwQa6iLTfpoqkHLHh22wSUUpuDhKJ3Y5TZZ3WXYUFCZfqaLxp9HaU5QMd1flukM>
26. <https://www.uship.com/ship/household-goods/furniture/pricing/4iwQa6iLTfpoqkHLHh22wSUUpuDhKJ3Y5TZZ3WXYUFCZfqaLxp9HaU5QMd1flukM>
27. <https://www.uship.com/ship/household-goods/furniture/pricing/4iwQa6iLTfpoqkHLHh22wSUUpuDhKJ3Y5TZZ3WXYUFCZfqaLxp9HaU5QMd1flukM>
28. FlightPriceSF.PNG
29. FlightPriceBOS.PNG
30. FlightPriceMin.PNG
31. <https://www.skyscanner.com/car-rental/results/95673577/95673577/2019-04-26T10:00/2019-04-28T10:00/30>
32. https://github.com/NeonMercury/python-lua
33. <https://en.wikipedia.org/wiki/Comparison_of_programming_languages#Expressiveness>
34. <https://raspberrypi.stackexchange.com/questions/38321/raspberry-pi-lifespan-reliability>
35. <https://www.linkedin.com/salary/technical-support-engineer-salaries-in-lawrence-kansas-area>
36. https://www.isixsigma.com/topic/warranty-calculation/