5-2013

# SmartPager – An Interactive System for Restaurant-Customer Interactions

Daniel Bauer
dbauer3@utk.edu

Rohan Dave
rdave3@utk.edu

Joe Edwards
jedwards43@utk.edu

Tyler Thurston
tthurst2@utk.edu

# SmartPager Senior Design Report

Rohan Dave, Daniel Bauer, Joe Edwards, Tyler Thurston

## Section I: A Story About Our Product

Mark is the owner of a small local restaurant. While not wildly successful, it has been steadily growing in popularity for some time now. As his restaurant starts to become more popular and more frequented, new problems arise related to the increase in business. Certainly, these are welcome changes, but they are things he has not had to deal with before. Some of them have obvious solutions. For example, if he doesn't have enough wait staff to work all of the tables during the dinner rush, he should hire more wait staff. However, some problems do not have as obvious of a solution. One example of this is the fact that when a small restaurant starts to become more popular, the wait times to eat at that restaurant will go up. This can lead to frustration among the customers and lead to the overall opinion of his establishment becoming more negative.

Mark, being a somewhat savvy business owner, knows that there are some ways to fix his problem with irate customers and long wait times, but most of them would be prohibitively expensive. One way to decrease wait times might be to increase the number of people that could be served at once. However, Mark's restaurant is already accommodating close to its max occupancy level. Trying to fit more customers into his existing building may work, but would probably ultimately end up causing discomfort for the customers being crammed into the existing space and would possible even end up violating fire code safety restrictions.

Well, if fitting more people into the existing space is not an option, Mark has also considered the option of expanding to a larger location or adding on to the existing building. But that comes with even more problems. First off, starting a construction project to expand the current building would lead into a myriad of financial, legal, and logistical problems. First, Mark would have to run around and get all of the required permits and permissions to do construction (assuming that he actually owns the property and is not just renting it). Then, he has to find financing for the construction. While the restaurant is performing well enough, it is not exactly turning a large profit, so a construction project would require taking out a rather large loan. And then, if all of those conditions were met, adding on to the existing building would require closing down the restaurant, meaning money is not being made while interest on the loan continues to build up. And then, after all of that, there is no guarantee that the construction would have any noticeable effect. Of course, Mark could also move to another existing location or build a new location, but both of those have the risk of losing customers. Regular customers know where the restaurant is and are familiar and comfortable with the location. If that were to change suddenly, then they may very well no longer be customers.

With those options ruled out as being infeasible due to a number of hurdles, Mark considers other options. If the problem is customer happiness and attitude while waiting for a table, perhaps there is a more direct way to fix the problem – find a way to keep the customers happy while they are waiting and/or reduce the amount of time they have to wait inside the restaurant. Mark considers first revamping his reservation system. Currently, customers wanting to make a reservation have to call in to the restaurant, wait for a hostess to be available

to talk to them, and then try to work out an available reservation time.  This process is exhausting for the customers and tends to serve as a disincentive for them to give Mark's restaurant their patronage.  Mark has looked into a mobile or web-based solution to the reservation problem, but most of the quotes he has seen have been a tad bit on the exorbitant side for a product whose quality can't be guaranteed.

It seems as though any possible solution to Mark's problem will have a generally high cost with either large drawbacks or a low chance of success.  Mark was wondering if perhaps this was the limit to how successful his business would ever get when he heard word of a new product that might be perfect for his problem – the SmartPager system.

After doing some research, Mark learns that the SmartPager system is a product that can solve his problem in multiple ways while staying affordable.   The system consists of two parts: a mobile-phone application customized and designed with respect to the business which can interact with a server at the restaurant for managing reservations as well as a physical pager for in-store user.  Unlike the traditional pagers that he currently uses, these new SmartPagers would provide customers waiting for a table with an interactive and engaging experience.  The device comes with a touch screen interface that allows the customer to browse the menu and read about the details of each available dish.

Mark is skeptical at first.   He already has a pager system in his restaurant, and the pagers he has now cost him thousands of dollars for just a handful.   Surely these more advanced pagers would be even more expensive, Mark thinks to himself.   He soon learns, however, that this is not the case.   In fact, the pager system costs less than the typical current pager systems.   In addition, the cost of the app development is competitive with comparable existing alternatives and comes with free support as long as the app is in use.

This, Mark thinks to himself, is exactly what I need.   The mobile application solves part of the problem by allowing people to make reservations from their mobile device, meaning they can be handled more efficiently than they would if the customer were to call in.   This means less customers would be waiting around for a table as they would be able to adjust their arrival time appropriately.   The physical SmartPager would solve the other part of the problem by giving the customers something to occupy themselves with while they wait – something that is especially important for families with young children.   Mark quickly contacts the SmartPager team so that he can begin the process of getting his own SmartPager system created and installed.   Satisfied that this problem, at least, could be taken care of in an effective and cost efficient manner, Mark can now spend more time working on other problems, such as finding another way to acquire more property to open another location, continue to grow his business.


## Section II: Motivation and Market Analysis

When it comes to the restaurant industry as a whole there are many driving features that can vary from place to place, but the vast majority needs some form of notification for their customers about when their table is available. The paging systems in place currently get the job done, but the industry can benefit greatly by having this system expanded upon and innovated. "In these industries, the need for instant communication between client and staff is crucial, and

this is why paging systems that are high quality and reliable while also using more innovative technologies to stay abreast the modern world are needed." ("Why Your Restaurant Need Restaurant Pagers System." http://pressreleaser.org/why-your-restaurant-need-restaurant-pagers-system/1405351 (2013) ) Customers appreciate when new innovations can make their dining experience a more pleasurable one. "A comparison of eleven technologies commonly used in restaurants found that restaurant customers find virtual menus with nutritional information to be the most valuable of new technologies, followed closely by online reservations, kiosks for ordering, and pagers for table management." Withiam, Glenn, "Cornell Study Finds Restaurant Goers Like Innovations in Early Dining Stages." (2009) With the ideas of a paging system and virtual menus in mind, one begins to imagine the innovated pager that allows the user to view items while they are waiting to be seated. This is precisely what the industry needs as it can speed up the restaurant itself by seating customers who already know what they want to order. This will effectively eliminate some of the time customers sit viewing menus after already being seated. This speed-up as a whole would result in the restaurant being able to serve people faster and more efficiently during their busiest hours resulting in more revenue. "Technology systems can support restaurant managers' efforts to improve sales and profits..." Kimes, Sheryl E. "The role of technology in restaurant revenue management."*Cornell Hospitality Quarterly* 49.3 (2008): 297-309. From a viewpoint that does not focus on the financial benefits of the system comes the realization that entertaining the customer and keeping them happy directly benefits the restaurant. This could cover a broad range such as praise through social media boosting the restaurant reputation to a happier customer leaving their server a better tip. By providing the customer/user with a paging system that does the job of current systems as well as showing menu items and possible deals, it allows the customer to turn their focus away from their wait time and more towards what they want to eat. "The current method is inefficient because it does not keep guests entertained while they wait for their table nor inform them of available menu items." Rochford, Nicholas, et al. "Advanced Restaurant Pager System." (2011). The system would also allow for a more efficient way to inform customers about particular restaurant specials that are in effect. This alongside with restaurant specific customizations opens up variations of uniqueness that some establishments strive for. The overall result would be a restaurant that would operate more efficiently and give a customer an all around better dining experience through quicker service and more engaging wait times. With all of this in mind, the question arises regarding how receptive the industry would be of the new technology. "The segments of casual service, quick service, and family style, as well as chain versus independent restaurants, use systems differently from each other to meet their specific industry needs." Huber, Marsha M., Murat Hancer, and R. Thomas George. "A comparative examination of information technology usage in the restaurant industry." *Journal of Foodservice Business Research* 13.3 (2010): 268-281. As stated the differing systems that are in effect in the present allow for integration of the new technology easier. The reasoning for this is that restaurants currently using the pager systems would more easily adopt the innovation, while others that are not could see the advancement of the pagers as a good way to progress with their business. By allowing customers to have an opportunity to use new technology when they go to a particular establishment, they tend to realize how convenient and valuable it is. "Restaurateurs need to make sure their customers will accept new

technology when it is introduced, but we found a sort of catch-22," said Verma. "We found that people who use a particular technology find that technology more valuable than those who never tried it. However, to see that benefit the customer first has to try the technology, and many people aren't quick to try new technologies." (Withiam) As quoted, the primary concern when introducing new technology is how receptive the community is, but with the familiarity of touch screen devices that circulate society today the new pager should be very easy to understand. This of course is aided with simple on screen instructions as well as an ease of use for the restaurant to customize and change to their specific needs.

To conclude, this innovation in the pager technology of the present is of great importance to the restaurant industry. The new technology will allow for a speed up in service by seating customers who have already had a period of time to view the menu from the pager, the customers themselves will be much less focused on their wait time and more entertained using the new pager device to view menu items and restaurant deals, and with less focus on waiting for food or a table the customer will be much happier. While not every establishment has a need for the pager systems that are currently in place today, these new improvements and innovations will give owners a chance to streamline their business while showing their customers the importance and convenience of the new system. Overall, the reason why this technology is needed is to keep the customer happy while providing them with an overall faster and better quality service.

## Section III: Measurements of Similar Products

Currently there exists no system available to the market that provides a combination physical pager and phone application for reservation making; however, this does not imply that there are not similar products that provide part of what our entire system can provide. The following paragraphs will detail how other systems provide a similar functionality without fully satisfying the problem that our SmartPager system solves.

The backbone of the phone app that is used in the SmartPager system is the handling of received messages from the user's phone. There exists a patent detailing a system for transmitting data messages created on computer systems to phones.[1] This system provides a way for a computer to communicate with a phone utilizing the mobile phone network to transmit information. The data from the computer is routed into the mobile network via a communication server. The server packages the information into a discrete packet that can be sent over the mobile network and received by the phone. The key difference between the system described above and the system implemented in our SmartPager system is that our system reverses the process. The phone constructs a packet which is packaged for use over the internet, rather than the phone network, the packet is transmitted to the restaurant server by traveling over the mobile network. By reversing the process, the result is that our app is able to reply to the phone with an authentication message in order to demonstrate the message was successfully received. The other system does not allow for acknowledgment of a received message without having to specify what type of phone is able to complete the packaging of an acknowledgement packet.

Another crucial component of the SmartPager system is the use of the physical pager

with the touchscreen.  A similar system can be found in the patent for "On premises restaurant communication system and method" which details how to handle communication systems in a restaurant environment.[2] The system that the patent details involves the customers receiving a personal identification number and using that to identify where the customers are seated and want by making them enter it at their table and when ordering. This systems allows for the server to know where their order to go and who it is for. It differs greatly; however, from how the pagers in the SmartPager system actually operate. Rather than have the customer keep track of their number and be required to enter it when seated, the pager will be able to take the order from the customer while they are waiting. Then after the customer is seated the pager can be given to the server, who extracts the order from the pager, and returned to the front desk. This dissipation of labor from the user to the restaurant will allow for a simpler experience for the customer, and can even help cut down on wait time for servers.

One of the unique aspects of the actual pager is the ability to input an order with it. There is a patent detailing a similar system called, "system and method for accepting customer orders".[3] In this patent, a system involving a centralized kiosk is implemented utilizing a touch screen or some type of input device. The customer can go the kiosk and type in their order and pay for it in one single location. The kiosk then distributes a number to identify the customer and notify them when their food is ready over an intercom system. The kiosk would be able to accept many different types of payments. The similarities in the SmartPager system and the system previously described lie in the SmartPager allowing the customer to fill out an order form. In essence, each individual pager would act as a kiosk in this scheme. The major difference being that the SmartPager pagers would only allow for the customer to say what they want to order and not actually submit it to the system. In an ideal environment for the SmartPager system, the customer would be in a location which the restaurant would know about which allows for the pager not to have to be able to distinguish customers with a number.

The system which most closely replicates the experience that the SmartPager system will provide is detailed in the patent titled, "Personalized and dynamic restaurant interfaces and systems".[4] The system described involves an apparatus which allows for the user to input their order either at the restaurant or not at the restaurant using some generic device, calling in the order and going through an automatic ordering menu, etc. The system will then automatically handle the order and obtain payment from the customer before placing the order. This system duplicates many of the uses of the SmartPager system such as automatic ordering and not requiring the customer to place the order at the restaurant or deal with a server or host/hostess over the phone. The major difference in the system lies in their primary goal. The SmartPager system is designed to minimize wait time at a restaurant for a table and provide some use from the standard pager rather than just as a pager. The above system is designed primarily for a fast food establishment to take orders from the customer without any human involvement. The result of this difference is that the SmartPager system involves more personal interaction with restaurant staff while the system described above automates the entire process, which would ultimately detract from the level of service and the experience one would obtain at a "sit down restaurant".

The backbone of the SmartPager app is the ability to transmit the data to the computer server at the restaurant. The patent titled, "Pagers over a wireless network" details a scheme in

which many different pagers, each with some identifier such MAC address or product number, are able to receive a page over a wireless network.[5] The advantage of this system is its ability to transmit the data quickly and from a much further distance than a convention signal pager would be able to achieve. The difference is that the SmartPager app utilizes the mobile wireless to get the information to the server rather than a wireless network. This means that the user can potential use the app at any time they have signal for their phone, rather than be dependent on when they have access to wireless internet.

**Section IV: Measurements of Prototype and Final Product**

As a product development process proceeds it is necessary to continually measure and record characteristics and data regarding the different prototypes and iterations of the design. These measurements are an invaluable tool in determining whether a design decision was beneficial to the design or whether it had a detrimental effect. The data also has the added benefit of being able to compare the designed product against a similar competitive product in order to demonstrate the designs superior aspects and characteristics. Over the next few paragraphs the physical measurements of the SmartPager System will be discussed along with the implications of that data and what effects it had on the design process.

The easiest aspect of the SmartPager system to measure would be the pager handed out to guests at restaurants. The size of the pager is approximately 5 inches by 5.5 inches by 1 inch as compared to the Jtech restaurant pager which is only 4 inches by 4 inches by 0.75 inches. This decrease in size would suggest that the Jtech pager is superior to the SmartPager pager; however, there are some crucial details that this physical measurement leaves out. The Jtech pager is only a pager thus it can only vibrate to inform the customer their table is ready. The SmartPager pager has a screen and timer thus it is able to not only inform a customer when their table is ready, but also to show them items on the menus, inform them of specials, what ingredients each item has, how much each item costs, and etc. These added uses for the SmartPager create a useful, interactive device which is able to multitask and sustain the customers attention to keep them focused on the food that is coming. The Jtech pager only serves the purpose of telling a patron their table is ready, and possibly remind them that they are waiting for food when they could eat somewhere else without a wait. With this added benefit it is clear that the SmartPager is superior to the JTech and really it should be seen as a win for the SmartPager pager to maintain competitive size with a rival product while offering significantly more utility.

The previous paragraph helps to show why even though the SmartPager may not be as small or portable or efficient as a competitors product it is able to provide significantly more use while still staying competitive in those previous categories. The design measurements carried out throughout the semester are difficult to quantify effectively. The issue arises with how to compare a partial creation with a final product. The first prototype we had developed was a paper pager meant to be a model of the actual pager. The size of the first prototype was significantly smaller than what the final design consisted of mainly because we had no idea what size the screen or microcontroller would be in the end. The functionality of the devices were quite similar,

able to view different menu options, however our initial prototype allowed for the viewing of ordering in the queue while the final design was not able to accommodate that functionality due to a lack of interface with the SmartPager server systems. This issue arose due to the desire to remain competitive in terms of cost to the design as adding an Wi-Fi module would have driven the price up by 60$ which we felt was too much considering the our rough estimate at the time had the final pager costing $120 while the Jtech pager only costs about $60 per pager. We felt that even the added functionality of the pager was not able to justify the cost of the addition and thus the integration with the server system was ultimately scrapped.

The next iteration of our design from the paper model, was the actual final product. This was due to a shipping error which resulted in our micro-controller reaching us in the 1st week of April. A consequence of this was that many of the design decisions we had made for the paper model of the pager had to be scrapped as we realized the utility of our resources. As we began to realize what we were able to do and not do it became meaningless to try and design a case around an incomplete pager. The reason for this is that even though we have wanted to get the pager to be hand-held, the necessary power supply to power the microcontroller, the touchscreen, and the vibrating motor became too difficult to predict. As our design decisions changed regarding how what the pager should and should not be able do, we made the size of the pager a small issue in our mind. This was due, as discussed in the second paragraph, to the fact we believed we could justify any negative aspects of physical design, size, battery life, and appearance and spin them into positives due to the touchscreen. The touchscreen honestly transforms the pager in such a way that it really is not even comparable to the existing products. An equivalent example would be comparing an iPhone to a flip cellphone from the late 2000's. Sure the iPhone might be bigger, heavier, and have a poorer battery life, but the number of functions it was able to do ultimately transformed phones and the reality was the iPhone was too dissimilar to a cell phone and received the name smart phone. I think this example readily illustrates the fact that the SmartPager pager is so far ahead of the competition it cannot be compared in the existing measurements.

The other aspect of the SmartPager system is the app and server side system which manages information. To our knowledge, there is no existing competitive product that provides both an app and reservation service that is integrated with a queuing system to manage reservations. The final app was able to be installed quickly onto an android device and interface effectively with the server software.  The time taken to place a reservation from the app was a matter of seconds (on the scale of 2 to 4 seconds), although it was somewhat higher when there were network problems.  However, as this is not an issue with the application itself, the time is not reflective of the quality of our product.  As for measurements of the server side application, there is not much to be taken under consideration.  The server software is limited only in its ability to store reservation by the limitations of the MySQL database.  From what we have found, there is not set limit to database size.  This would indicate that the limit is more based on the amount of storage available which can be augmented easily.  Of course, having too many rows in a MySQL table would likely result in more problems in terms of run time efficiency.  Pulling data from the MySQL database would become more and more expensive and would require more RAM to store the information in the program.  So although there is not set limit, there would likely be a limit to the amount of data that could be stored based on how the amount of data

impacts the performance of the server application. Additionally, the server can handle as many simultaneous connections as the number of threads that the server can spawn, allowing for maximum responsiveness to clients using the application.

## Section V: Design Decisions

Over the course of our senior design project, we obviously had to make a number of important design decisions. Often, these decisions would have a fairly strong impact on the final product we would produce. Here, I hope to lay out a few of the most important decisions, our thought process behind them, and how they ended up working out.

One of the first design choices we were faced with was what microcontroller we were going to use in the SmartPager. This was easily the most important design choice we had to make in terms of our physical pager as any other aspects of our design for the SmartPager will be directly affected by our choice of microcontroller board. Our choice ended up being to use an Arduino Mega 2560 board. Our main motivation behind this choice was the number of pins available on the Arduino Mega as well as the ready availability of compatible shields for Arduino devices. The choice ended up working out fairly well. The number of pins available on the Mega allowed us to use the TFT touchscreen shield without having to worry about running out of pins. This means that we have the potential to add more shields (such as a wireless radio shield) to the SmartPager without having to worry about running out of pins. Unfortunately, the ARM processor on the Mega has a slower clock than the processor on the regular Arduino board. This means that parts of our code (in particular, the functions that load and display bitmap image files) run fairly slowly. The impact on loading time is not enough to have a strong negative impact on the final product, but it is enough to be useful. The other big result of using the Arduino Mega relates to the inclusion of wireless capability. We definitely would have liked to include wireless capability as it would allow us to have the SmartPager be fully integrated with the server software. However, the Arduino wireless shields were somewhat cost prohibitive (as we wanted our price to remain competitive with existing products) and would have put more of a toll on the battery life. As such, our SmartPager is currently unable to talk to the server software and exists as an isolated part of the system. Obviously, this is not optimal as we would like all of the parts of our product to be connected in some way.

From where we stand now, if we were to make a new version of the product, we would likely use a Raspberry Pi for the microcontroller instead. Using a Raspberry Pi would have solved many of our problems for us: we could use any touchscreen with HDMI input, we would be able to use a USB wireless dongle to add Wi-Fi capability, the processing power of the Pi should be fast enough to handle displaying images without a noticeable delay, and the fact that the Pi runs a variant of the Debian operating system would provide for a much more familiar and comfortable coding environment. Although the Arduino Mega worked well and was good enough for this iteration of our product, given more time, we would definitely go for the Raspberry Pi instead as the biggest possible downsides would be higher energy use which should be able to be solved with high-capacity, rechargeable lithium-ion batteries.

In terms of design choices for the client application and the server software, there were a

few options, although none were as essential as the SmartPager microcontroller decision. One of the first decisions was what mobile phone platform we wanted to target. Our choice ended up being Android. This was mostly due to the fact that the computer scientists in our group were already familiar with the Android platform and API. It would, of course, be possible to add an iOS application.

As for the server software, there were several smaller decisions throughout the process that affected the final product. The first decision was what programming to write the server software in. We chose Java for this because it would allow the server code to be very familiar to the application code (since Android runs a Java virtual machine). In addition, Java code is fairly portable across a variety of platforms as long as the platform has an implementation of the JVM. The server code could be written in basically any language that can talk to a MySQL server and read and write to and from socket streams. This is because all of the communication between the server and the app is basically done using text streams. So the server code could be re-written in C/C++ and as long as it can talk to the database and send/receive the same messages as the Java version of the code.

Related to the server side software, we also had to make a decision about how to store the data being collected (reservations, orders, and menu items). We decided to use a MySQL database as it would allow an easy to use and widely available database implementation that can communicate with most widespread programming languages. Changing the database being used would be possible but would require changing some of the server code. Due to how the server software is designed, all of the functions to interact with the database are contained inside their own class. Rewriting the functions in this class should be the majority of the work required to change the database implementation used to store the data.

Another design decision that was hinted at earlier has to do with how the server software and user application would talk to each other. In the end, we decided that the best way would also be the simplest. We have the server and client software communicating using simple text streams. This allows for the data being sent back and forth to have a wide range of values. This also makes it easy to send error messages that might occur on the server software to be sent back to the client software and displayed to the user. Although a text stream does require more bandwidth in terms of sending messages back and forth, it allows for the code that interprets and sends data to be simpler on both the client and the server ends.

This is obviously not an exhaustive list of the design decisions that we made, but it illustrates most of the major design decisions we made. It also attempts to look at our reasoning behind the decisions. For the most part, our design decisions worked out fairly well and had more of a positive impact than negative on our final product. There is always room for further improvement, but considering our constraints and goals in terms of what we wanted to maximize in our product, our design decisions were for the most part beneficial.

# Section VI:
# User Guide

# Contents

**Quick Start Guide:**

In order to properly setup and use the Smart Pager paging system, it is recommended that you read the following quick start instructions.

To set up the paging system:

• Carefully pull up the top part of the case to expose the inner-workings of the Smart Pager. If you are unsure which side is the top part of the case to lift, it will be the same side in which the touchscreen can be seen.
• Once open, simply connect two 9-volt batteries to the battery connectors inside the casing. There is no need for caution in regards to mixing positive and negative ends as the 9-volt connectors only connect the correct way.
• Now that the batteries are connected flip the internal switch and watch as the touchscreen lights up and the pager becomes ready for use. Now simply close the casing back up by lowering the top part of the case so that it fits snugly back in place.
• With the batteries plugged in and the case closed up your Smart Pager is now ready for use! When the need arises to replace the batteries for your Smart Pager simply refer back to these instructions.


**How to use the Smart Pager:**

• Now that the pager is ready for use, it is time to explore the features that the pager showcases. The first screen that is seen when the pager is powered on will be the timer. This feature will allow you to set a wait time for the customer. By doing this you will be able to eliminate any confusion of when the customers table is ready due to the internal motor that will cause the pager to buzz and vibrate.
• In order to set the time simply hit the (-) to decrease the number of minutes or hit the (+) to increase the number of minutes.
• Once the desired number is shown and visible on screen tap the accept button.
• Now that the time is set the pager will display on-screen instructions for the customer so that they know exactly how to use the pager to view menu items.
• After a very short time the instruction screen will disperse and the menu items will be in full view. In order to change a particular item on screen simply touch the right or left side of the screen.
• To view more information about a particular menu item simply touch the center of the screen and the Smart Pager will display the item name, price, and ingredients.
• To go back to view more items touch the "Back" button.


**Now that you have read these instructions on how to setup and use the Smart Pager, you are fully and completely informed on how this device functions and how to use it.**

**With this knowledge all that is left to do is put the pager in the hands of the customer.**

### Handling

Handle the Smart Pager with care as it contains sensitive electronic components inside and is made from plastic.  The Smart Pager can be damaged if punctured, set ablaze, crushed, dropped, or come into contact with liquids.  If the Smart Pager appears to be damaged in any way do not give it to a customer and do not attempt to self repair the pager.  The Smart Pager case is resilient, but if there is concern for maintaining the case, use a cover.

### Repairing

Do not attempt to repair damages to the Smart Pager yourself.  If you disassemble the Smart Pager it could cause injury and could potentially damage the electronic components that are part of the Pager.  If the Smart Pager is damaged or malfunctions, contact an authorized Smart Pager representative for more information on how to get the pager repaired or replaced.

### Batteries

Only used the approved 9-volt batteries to power your Smart Pager.  Failure to use the appropriate batteries could potentially damage your device or cause it to not function properly.

### Operating and Storage Temperature

The Smart Pager is designed so that it will function properly in ambient room temperatures.  Due to this, do not store the Smart Pager in extreme temperatures such as 0 or 100+ Fahrenheit.

### Cleaning

When cleaning and maintaining the Smart Pager, do not use abrasive cleaners or abrasive cloth.  Instead use a soft cloth that can be damp if needed.  Do not submerge the pager in water or use running water to clean it as it will damage the electronics.  Do not put the Smart Pager in the washing machine or dishwasher.

### Disposal

The Smart Pager may be disposed of according to proper laws and regulations of the buyers local area.  Do not simply throw the pager in normal waste, follow proper procedure and protocol.

### Important warnings

•      Do not attempt to eat this product as there is not a single edible part that is used in the construction of this device.  Failure to comply with this warning could lead to endangerment of your health.  If any part of this device is eaten contact a hospital.

•      Do not attempt to stick this product in your eye socket.  There is not a single part of this product that is meant for this purpose.

•      Do not sit on this product.  The Smart Pager case is sturdy for normal use, but is not designed to support a persons full weight or to be used as a seat.

•      The Smart Pager is in no way designed to be used as a medical device, therefore do not

attempt to use this device to deliver medicine to a dying patient.

•       Do not drop this product from an elevated height as the impact could potentially damage the case and internal electrical components.

•       The Smart Pager is in no way designed to be used as a flotation device, and as such should not be used in this particular manner.

•       Do not attempt to use this product for any act that it is not explicitly designed to do.  The approved way of use for this product is as a timed, interactive, touchscreen-equipped paging system for waiting customers at a business.

1. Preamble: This Agreement, signed on Apr 26, 2013 [hereinafter: Effective Date] governs the relationship between Client, a private person, (hereinafter: Licensee) and Smart Pager inc, a duly registered company in TN, United States whose principal place of business is Knoxville, TN, United States (Hereinafter: Licensor). This Agreement sets the terms, rights, restrictions and obligations on using [Smart Pager] (hereinafter: The Software) created and owned by Licensor, as detailed herein

2. License Grant: Licensor hereby grants Licensee a Personal, Non-assignable & non-transferable, Commercial, Royalty free, Without the rights to create derivative works, Non-exclusive license, all with accordance with the terms set forth and other legal restrictions set forth in 3rd party software used while running Software.

     2.1. Limited: Licensee may use Software for the purpose of:

        2.1.1 Running Software on Licensee's Website[s] and Server[s];

        2.1.2 Allowing 3rd Parties to run Software on Licensee's Website[s] and Server[s];

        2.1.3 Publishing Software's output to Licensee and 3rd Parties;

        2.1.4 Distribute verbatim copies of Software's output (including compiled binaries);

        2.1.5 Modify Software to suit Licensee's needs and specifications.

2.2 **Binary Restricted:** Licensee may sub-license Software as a part of a larger work containing more than Software, distributed solely in Object or Binary form under a personal, non-sub-licensable, limited license. Such redistribution shall be limited to {apps} codebases.

2.3 Non Assignable & Non-Transferable: Licensee may not assign or transfer his rights and duties under this license.

2.4 Commercial, Royalty Free: Licensee may use Software for any purpose, including paid-services, without any royalties

2.5 Licensor shall retain full title in Trademarks, and any trademarks and trade names contained, including Software's names, logos, and all other intellectual property. Unless specifically stated in this license, no license shall be made to use, associate or affiliate Software with Licensee in any manner. Licensee may not use Software's name, trade name, trademarks or logo when distributing derivative works of software to 3rd parties.

2.6 If Software is covered by Patents, Licensor grants Licensee a non-exclusive, worldwide, royalty-free patent license under Licensor's essential patent claims, to make, use, sell, offer for

sale, import and otherwise run, modify and propagate the contents of Software.

3. Term & Termination: The Term of this license shall be until terminated. Licensor may terminate this Agreement, including Licensee's license in the case where Licensee :

      3.1 became insolvent or otherwise entered into any liquidation process; or

      3.2 exported The Software to any jurisdiction where licensor may not enforce his rights under   this agreements in; or

      3.3 Licensee was in breach of any of this license's terms and conditions and such breach        was not cured, immediately upon notification; or

      3.4 Licensee in breach of any of the terms of clause 2 to this license; or

      3.5 Licensee otherwise entered into any arrangement which caused Licensor to be unable   to enforce his rights under this License.

4. Payment: In consideration of the License granted under clause 2, Licensee shall pay Licensor a {fee}, via PayPal or any other mean which Licensor may deem adequate, and through Binpress' clearing system. Failure to perform payment shall construe as material breach of this Agreement.

5. Upgrades, Updates and Fixes: Licensor may provide Licensee, from time to time, with Upgrades, Updates or Fixes, as detailed herein and according to his sole discretion. Licensee hereby warrants to keep The Software up-to-date and install all relevant updates and fixes, and may, at his sole discretion, purchase upgrades, according to the rates set by Licensor. Licensor shall provide any update or Fix free of charge; however, nothing in this Agreement shall require Licensor to provide Updates or Fixes.

      5.1 Upgrades: for the purpose of this license, an Upgrade shall be a material amendment
      In The Software, which contains new features and or major performance improvements and    shall be marked as a new version number. For example, should Licensee purchase The Software under version 1.X.X, an upgrade shall commence under number 2.0.0.

      5.2 Updates: for the purpose of this license, an update shall be a minor amendment in The   Software, which may contain new features or minor improvements and shall be marked as
      a new sub-version number. For example, should Licensee purchase The Software under
      version         1.1.X, an upgrade shall commence under number 1.2.0.

      5.3 Fix:for the purpose of this license, a fix shall be a minor amendment in The Software, intended to remove bugs or alter minor features which impair the The Software's functionality. A fix shall be marked as a new sub-sub-version number. For example, should Licensee purchase      Software under version 1.1.1, an upgrade shall commence under number 1.1.2.

6. Support: Software is provided under an AS-IS basis and without any support, updates or maintenance. Nothing in this Agreement shall require Licensor to provide Licensee with support or fixes to any bug, failure, mis-performance or other defect in The Software.

      6.1 Bug Notification: Licensee may provide Licensor of details regarding any bug, defect or failure in The Software promptly and with no delay from such event; Licensee shall comply with Licensor's request for information regarding bugs, defects or failures and furnish him with information, screenshots and try to reproduce such bugs, defects or   failures.

      6.2 Feature Request: Licensee may request additional features in Software, provided,
      however, that (i) Licensee shall waive any claim or right in such feature should feature be

developed by Licensor; (ii) Licensee shall be prohibited from developing the feature, or disclose such feature request, or feature, to any 3rd party directly competing with Licensor or any 3rd party which may be, following the development of such feature, in direct competition with Licensor; (iii) Licensee warrants that feature does not infringe any 3rd party patent, trademark, trade-secret or any other intellectual property right; and (iv) Licensee developed, envisioned or created the feature solely by himself.

7. Liability: To the extent permitted under Law, The Software is provided under an AS-IS basis. Licensor shall never, and without any limit, be liable for any damage, cost, expense or any other payment incurred by Licensee as a result of Software's actions, failure, bugs and/or any other interaction between The Software and Licensee's end-equipment, computers, other software or any 3rd party, end-equipment, computer or services. Moreover, Licensor shall never be liable for any defect in source code written by Licensee when relying on The Software or using The Software's source code.

8. Warranty:

8.1 Intellectual Property: Licensor hereby warrants that The Software does not violate or infringe any 3rd party claims in regards to intellectual property, patents and/or trademarks and that to the best of its knowledge no legal action has been taken against it for any infringement or violation of any 3rd party intellectual property rights.

8.2 No-Warranty: The Software is provided without any warranty; Licensor hereby disclaims any warranty that The Software shall be error free, without defects or code which may cause damage to Licensee's computers or to Licensee, and that Software shall be functional. Licensee shall be solely liable to any damage, defect or loss incurred as a result of operating software and undertake the risks contained in running The Software on License's Server[s] and Website[s].

8.3 Prior Inspection: Licensee hereby states that he inspected The Software thoroughly and found it satisfactory and adequate to his needs, that it does not interfere with his regular operation and that it does meet the standards and scope of his computer systems and architecture. Licensee found that The Software interacts with his development, website and server environment and that it does not infringe any of End User License Agreement of any software Licensee may use in performing his services. Licensee hereby waives any claims regarding The Software's incompatibility, performance, results and features, and warrants that he inspected the The Software.

9. No Refunds: Licensee warrants that he inspected The Software according to clause 7(c) and that it is adequate to his needs. Accordingly, as The Software is intangible goods, Licensee shall not be, ever, entitled to any refund, rebate, compensation or restitution for any reason whatsoever, even if The Software contains material flaws.

10. Indemnification: Licensee hereby warrants to hold Licensor harmless and indemnify Licensor for any lawsuit brought against it in regards to Licensee's use of The Software in means that violate, breach or otherwise circumvent this license, Licensor's intellectual property rights or Licensor's title in The Software. Licensor shall promptly notify Licensee in case of such legal action and request Licensee's consent prior to any settlement in relation to such lawsuit or claim.

11. Governing Law, Jurisdiction: Licensee hereby agrees not to initiate class-action lawsuits

against Licensor in relation to this license and to compensate Licensor for any legal fees, cost or attorney fees should any claim brought by Licensee against Licensor be denied, in part or in full.

**Bibliography:**

Livingston, James; Russell P. Blink; Kenneth J. Lovegree; On-premises restaurant communication system and method; Mar 30,2004; Patent Number US6712278 B2
http://www.google.com/patents/US6712278

Montague, Wayne; System and Method for Accepting Customer Orders; July 9, 2002; Patent Number: US6415555 B1 http://www.google.com/patents/US6415555

Pollock, Richard A.; Kimberly D. Williams; Personalized and Dynamic Restaurant Interfaces and Systems; November 15, 2012; Patent Number: US2012028728 A1
http://www.google.com/patents/US20120287281

Rock, Wayne; Salvatore Veni:  Pagers Over a Wireless Network; March 7, 2007; Patent Number EP1760949 A1 http://www.google.com/patents/EP1760949A1

Wendler, Bodo; Method for transmitting via digital moile phone networks and for displaying data messages created on computer systems; April 11, 2007; Patent Number: EP 1561664 B1
http://www.google.com/patents/EP1561664B1