

Shi Zhen

Building Business on a Remote Queuing System

Helsinki Metropolia University of Applied Sciences

Bachelor of Engineer

Information Technology

Thesis

14 April 2015

Author(s)	Shi Zhen
Title	Building business on a remote queuing system
Number of Pages	37 pages + 2 appendices
Degree	Bachelor of Engineering
Degree Programme	Information Technology
Specialisation option	Software Engineering
Instructor(s)	Jaana Holvikivi, Principal Lecturer
<p>The goal of the project was to build and commercialize a remote queuing system to reduce customers' queuing time at the service point. The product will allow customers to track their queue numbers without being physically present at the service point. The application was designed to be a web application so that there would be no platform limitation. Customers are able to use any internet accessible devices to monitor their queue numbers. Additionally, the service providers can benefit from offering a better service to their customers.</p> <p>The project consisted of two development parts, a technical part and a business development part. For the technical part, the project used a websocket protocol to achieve real-time communication. The main programming languages used were PHP, Javascript and HTML5. For the business part, the Lean start-up method was used in this project. In addition, the business model also contained B2B and B2C.</p> <p>The web application both on the server side and client side work properly. The business negotiations already have started with the ticket machine providers and several telecommunication operators. Moreover, the project joined the Business Model Competition in Finland where it was awarded the second prize, and in JA Start-up competition it won the first prize.</p>	
Keywords	Websocket, business model, lean start-up, web application, real-time communication

Contents

1	Introduction	1
2	Theoretical background	3
2.1	Introduction to communication protocols	3
2.1.1	Transmission Control Protocol/ Internet Protocol	3
2.1.2	User Datagram Protocol	4
2.1.3	Hypertext Transfer Protocol	4
2.1.4	File Transfer Protocol	4
2.2	Introduction to web real-time communications technology	4
2.2.1	Polling	5
2.2.2	HTTP long polling	5
2.2.3	AJAX	6
2.2.4	Websocket protocol	7
2.3	Overview of business model	9
2.3.1	Lean start-up	9
2.3.2	Business-to-business	10
2.3.3	Business-to-Customers	11
2.4	Overview of the queuing system	12
3	Design and implementation	14
3.1	Introduction to the remote queuing system	14
3.2	Implementation and design of remote queuing system	14
3.2.1	Design of data packages	15
3.2.2	Implementation of lower computers connection	16
3.2.3	Implementation of service point management application	19
3.2.4	Client-side real-time web application	24
4	Business plan and implementation	27
4.1	Overview of the business plan	27
4.1.1	Lean start-up implementation	27
4.1.2	Business plan	28
5	Testing and results	31
5.1	Application testing	31
5.2	Business testing	33

6.1 Technical problems and solutions	34
6.2 Business problems and solutions	34
7 Conclusion	35
References	36
Appendices	
Appendix 1. Future development on IOS platform	
Appendix 2. Qer homepage	

1 Introduction

In 2015, there will be 3.6 billion mobile phone users all over the world. Three billion people are using the internet and 80 percent of internet users are using their phone to access the internet [1]. The relationship between the Internet and mobile devices are closer, more and more practical applications come. Until summer 2014, 75 billion applications had been downloaded, out of which 10 percent are business applications [2]. People are accustomed to using mobile devices to deal with real-life issues, such as goods trackers or remote controllers. Moreover, due to the limitation of space, and high cost of labour, more and more companies prefer to offer online services to gain more satisfied customers and solve time management problems.

The idea of current queuing systems is to provide fairness to queuing customers by having one queue only, and attending customers on the first-come-first-served basis. However, with an increased number of customers and an insufficient number of service points, this system does not provide satisfactory services to customers. A growing number of customers expect something different, which could solve waiting time problems. This project will help service points improve their service without big extra costs. The aim of this project is help service points to cut their customers' waiting time. Furthermore, this project also aims to improve customer satisfaction. The project is mainly built based on the current queuing system by improving some crucial features.

The proposition of the remote queuing system is to allow customers to do multi-tasking at the same time. The product that is developed in this project will give customers more freedom, and moreover, drive the business around service point developing. Customers do not need to physically wait in the specified service point anymore, instead they can leave the service point temporarily without need to worry about missing their turns. Customers are able to check the queuing number situation virtually and do other tasks at the same time. For example, they can have a cup of coffee and at the same time monitor the queuing number virtually. The remote queuing system permits customers to visit the business surrounding the service point and therefore bringing more opportunities to business.

The goal of this part of the project was to design and implement a web application of a remote online queuing system. The web application provides an online queuing system

for customers so that customers can easily check the queuing numbers by monitoring their mobile devices instead of being physically present at the service point. Because the project will be a web application, it does not require customers' own smartphones. Elderly customers who do not have smartphones are also able to track their queuing number anytime. On the other hand, service providers will deliver better service due to fewer customers waiting at the service point. They can also get more satisfied customers, which is a business advantage against their competitors.

This project has been developed by a group of students, and it is still further developing [3]. The students of this group come from different departments and have different professional backgrounds. The division of the tasks was made clear, however ensuring a smooth collaboration. This project has already proved its business potential by being awarded the second prize in the Business Model Competition in Finland. After the Business Model Competition, the team gained useful feedback and improved the project. Finally, this project was awarded prize of JA startup and won 15000 euros. This was a milestone for the project, which pushed forward the remote queuing system to hit the real business.

2 Theoretical background

2.1 Introduction to communication protocols

When two strangers meet at first, they introduce themselves, and exchange some personal information, for example name, age, and occupation. Computers also have this kind of rules when connecting with each other via a network, called communication protocols. The protocol includes for example a communication package, establishment of the communication, and network error detection.

Communication protocols operate on different levels of complexity. The basic protocols are responsible for the simplest functionalities, such as establishing connections. For the more complex protocols, they provide more features such as reliability for the network.

Different communication protocols have different features and unique functionalities. The main common communication protocols are TCP/IP, UDP, HTTP, Ethernet, FTP, SMTP etc [4]. The following paragraphs will give an introduction to the protocols used in this project.

2.1.1 Transmission Control Protocol/ Internet Protocol

The TCP/IP standard for Transmission Control Protocol/ Internet Protocol is the basic internet protocol. It defines the protocol for electronic devices that establish a connection to the internet and communication rules between them. TCP is responsible for dividing data into small packages before they can be sent over a network. When the packets arrive, TCP will assemble the packets. In addition, communication with TCP will guarantee the data delivery to the destination address. When transmission comes across a problem, it will send a signal to the original sender and the sender will send a new data packet until then target receives the packet. IP will give different addresses to each computer on the internet so that they can be uniquely identified [5].

2.1.2 User Datagram Protocol

The User Datagram Protocol (UDP) works on the IP network layer protocol. The UDP is an alternative to TCP. However, compared with TCP, this is a connectionless protocol. UDP cannot guarantee data delivery and duplication. The UDP is suitable for applications that require effective data transmission instead of reliability [6]. For example, UDP has good performance on games and some application that have a slight data transmission.

2.1.3 Hypertext Transfer Protocol

The Hypertext Transfer Protocol is an application layer protocol. HTTP is used for transferring data such as text, graphic images, sound, video and other multimedia files on the World Wide Web. When users open a webpage, it will automatically use the HTTP. The HTTP can establish a connection with servers and push the HTML page to the user's browser. In addition, it is responsible for downloading files from servers to browsers or to any other applications that use HTTP[7].

2.1.4 File Transfer Protocol

The FTP is a standard for File Transfer Protocol based on TCP/IP. FTP is a protocol doing actions with files such as file access, file transfer and file management between computers and the internet. FTP is also a reliable protocol. It has automatic error detection and correction in data transfer. The possibility to connect all devices to each other, and share information between them in real time enables fluent use applications across platforms [8].

2.2 Introduction to web real-time communications technology

Nowadays, nearly everybody has communication devices such as mobiles, computers or tablets. The possibility to connect all devices to each other, and share information between them in real time enables fluent use applications across platforms. The real-

time communication technology has become ubiquitous. The most common application is the online chatting application, where people type texts in the online application and it will automatically share it with others. However, the reliability and speed of real-time communication also depends on the quality of the network [8]. Nevertheless, 3G and Long-Term Evolution have been able to contribute to a reliable and cost-effective solution in many places. Real-time communication is also limited by the number of base stations and quality of signals [9].

2.2.1 Polling

Polling is a way for data transfer and exchange between servers and client-side in the network field. During the data transfer period, the server will send broadcast requirements to all client-sides in the network at constant continuous intervals. When the target client-side receives a signal, it will establish a connection and send the data back. By using this way, servers and all client-side devices in the network will share data in real-time. However, polling is not an effective way to achieve realtime communication, especially when there is no new data need to transfer and exchange in the network. The server will still send requirements to client-sides continuously but at this moment the client-sides will not have data to share. It will cost unnecessary bandwidth consumption [10]. In addition, by sending the requirements and getting the responses will sometimes cost so that data transfer is not really in realtime.

2.2.2 HTTP long polling

HTTP long polling is also a way to build real-time interactivity. Client-sides push the requirements of new data to servers. The server will keep the requesting open until the new information is ready. Once the information is ready for the client-side, servers will send information to client-sides. When the client-side receives data, it will send another requirement to the server immediately. This operation will be repeated until the connection between the server and client-side is finished. A diagram of this operation is shown below in figure 1.

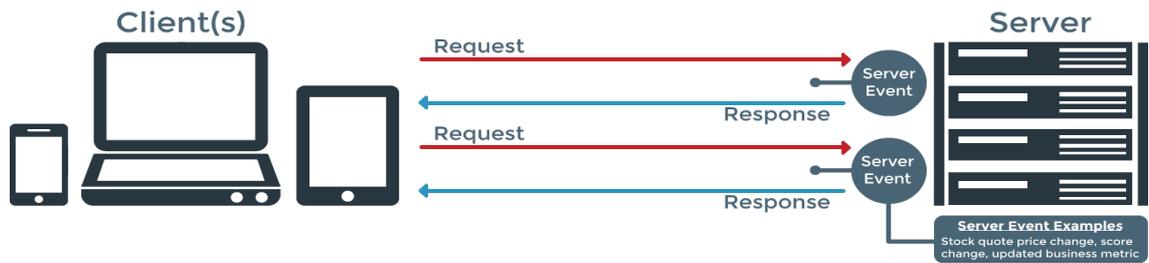


Figure1. HTTP Long polling [11]

Client-sides start polling and sending requests to the server, which is different from polling. By using this, server does not need to send requirements at constant intervals. Servers only need to wait for the new data and send it back. It will save bandwidth and reduce information jams [11]. For this reason, polling is more suitable for a small application, which does not need to transfer and exchange a large amounts data. For the HTTP long polling works better for larger applications.

2.2.3 AJAX

Asynchronous Javascript and XML is not a real-time technology itself. AJAX combines several technologies together. It includes: HTML, XML, XSLT but most important is XMLHttpRequest request object. This new technology was developed by Jesse James Garret in 2005.

When an application uses the AJAX model, it will make a fast update and reload pages without a need to refresh the entire web page. In AJAX, the real-time refreshing on the browser will only update the data changed on the server. The remaining part of the data on the web page will not be refreshed and changed. Figure 2 and figure 3 diagram shows the differences between AJAX and normal real-time technology.

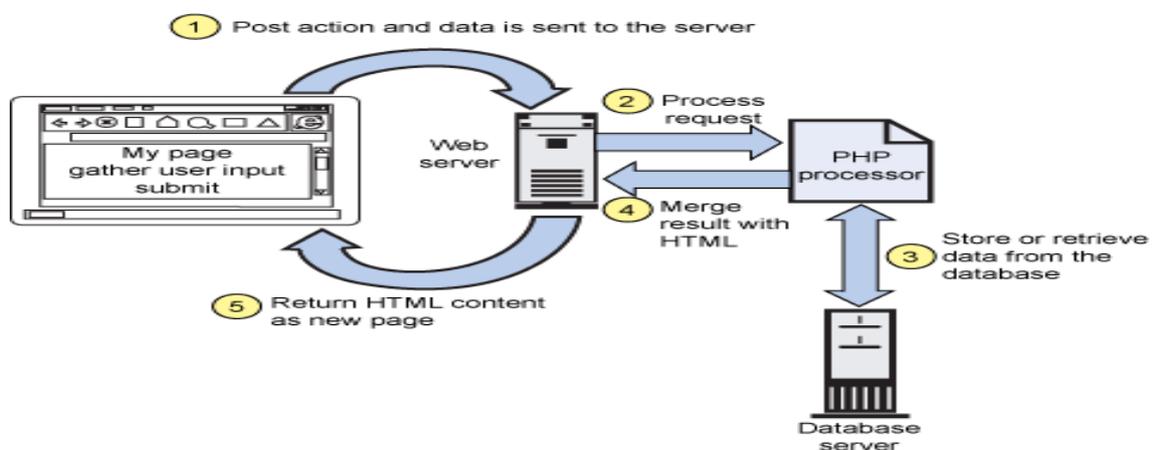


Figure 2. Normal real-time technology

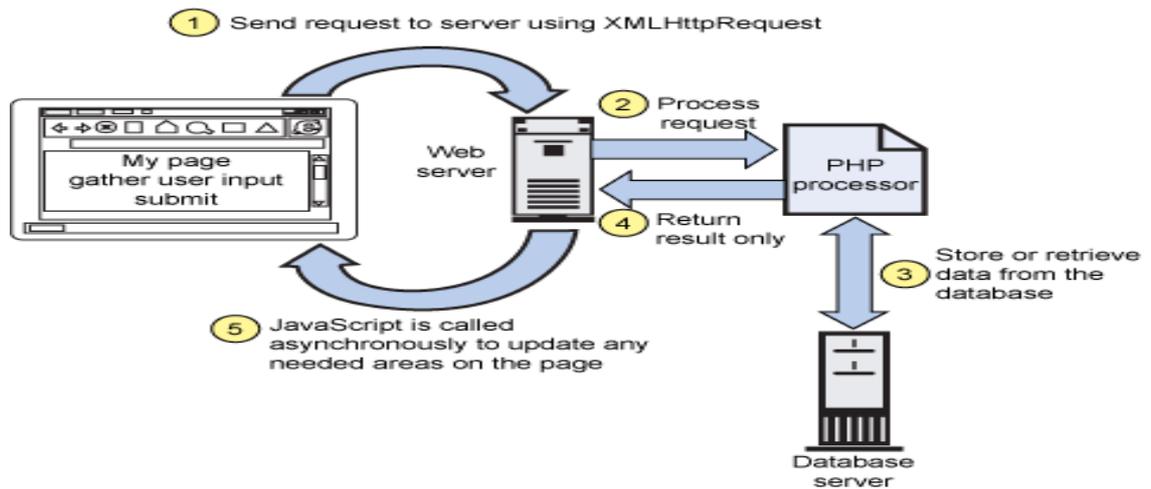


Figure 3. AJAX technology [15]

In AJAX, servers exchange data in realtime with clients by using stream loops. Because of these features, it also consumes bandwidth and memory usage very rapidly [12]. Besides that, AJAX also needs that the client-side sends the initial request first and then the server will give the response. If the client does not require first, the server and client-side cannot be synchronous [13].

2.2.4 Websocket protocol

The websocket is the latest web technology that can establish a duplex real-time communication between a server and client-sides. It means a great improvement of real-time web application technology, because polling and sending requirements always cause problems such as consume bandwidth, costs large memory usage and real-time delay. The websocket can avoid nearly all problems mentioned above caused by other methods. So this project has chosen websocket as the communication technology.

The client-side establishes a connection with the server by shaking hands. Both the server and client-side will listen to each other's updates and refresh. There is no need to shake hands every time. Once the action of the handshake is done, servers and client-sides can exchange data and share information until communication finishes. Because this is the duplex communication method, it will effectively avoid information delay. [14] There is no need for the server to wait the signal from the client-side and give feedback. On the other hand, it is also unnecessary for the client-side to wait for the result, signal or requirements from the server side and give the reply. Both the server and client-side can send information and new updates to each other any time

[15]. The following paragraph will give several examples to explain the advantages of websocket.

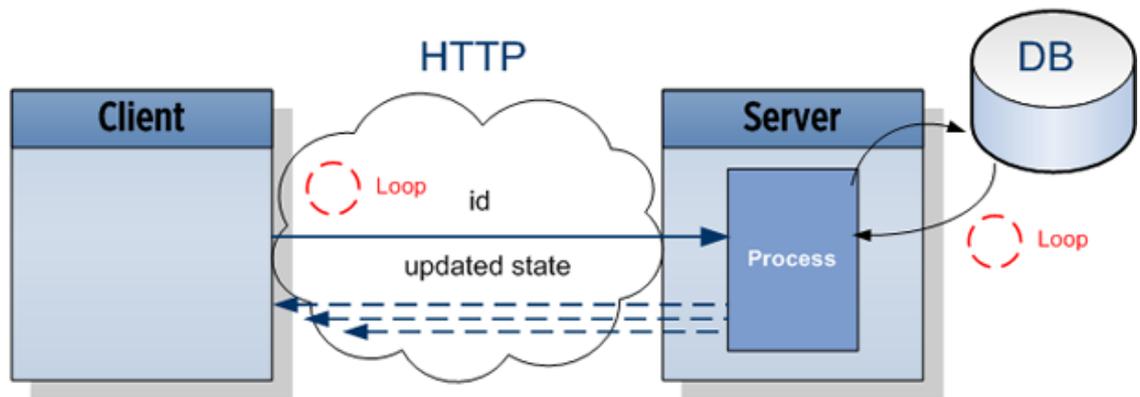


Figure 4. Websocket protocol

The stock application is a typical real-time web application. The data in this application are updated at small constant intervals. This application requires that the data updates show on the client browser accurately and in real time. This example application by using polling and websocket will illustrate the advantages and disadvantages. Network throughput is a measurement to see how much bandwidth is used by this application. In the experiment, three different numbers of clients were used to test the throughput of polling and websocket methods. For HTTP polling part, when there are 1000 clients using the application the throughput is 6.6 Mbps. When the number of clients increases into 10000 and 100000 clients. The throughput becomes 66 Mbps and 665 Mbps. However, when the application was using the websocket protocol, the throughput decreased rapidly. The experiment was still using three different numbers of clients that were the same as in the polling experiment. The numbers of throughput are 0.015 Mbps, 0.153 Mbps and 1.526 Mbps. According to the throughput data, it is easy to figure out that the websocket is extremely efficient in saving bandwidth [13]. Besides that, according to the study, consumption of the bandwidth of AJAX was twice that of the websocket in the same condition [17].

In addition, the amount of delay time is also a benchmark in the real-time application. Websocket also has very good performance in this respect. If an application takes 50 milliseconds to send data from the server to the client-side, polling has to take another 50 ms to send the requesting from the server to the client-side.

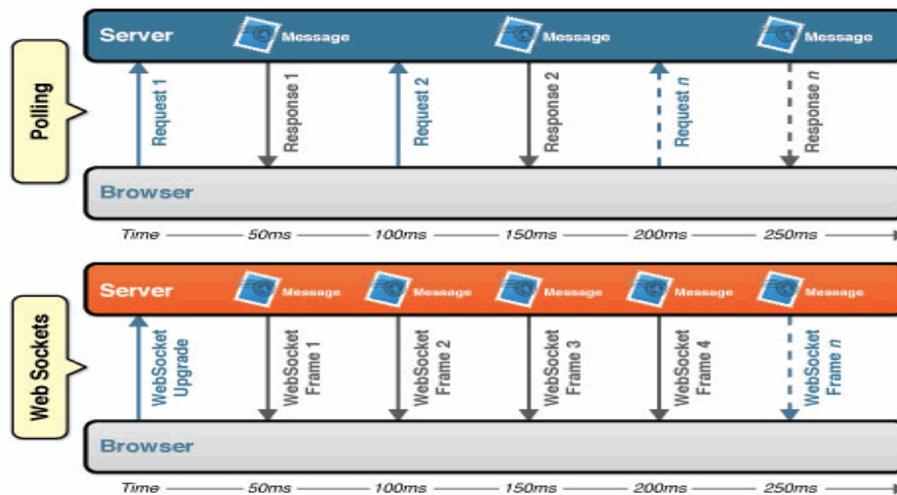


Figure 5. Comparison polling and the websocket protocol [13]

However, the websocket does not need to take those extra 50 ms to send data from the server to the client-side. Once the handshake between the server and the client-side is done, the connection will be established and existing until communication is finished [13].

2.3 Overview of business model

Business as an organization contains value propositions, target customer segments, distribution channels, customer relationships, value configurations, core capabilities, a partner network, cost structure and a revenue model. From the business model, a company develops the strategy for marketing and business goals. The creation of a business model will make companies especially start-ups more aware of the problems and advantages of business ideas in each segments of business development [18].

2.3.1 Lean start-up

There is always an adventure when a new enterprise is launched, whether a traditional production company or a high-technology start up. According to the research by Harvard Business School's Shikhar Ghosh, 75% of all started enterprises fail. Before the products have been tested on the market, it is hard to predict their success. [19]. However the entrepreneurs are always biased towards themselves and tend to believe that their products are best all over the world. Without market testing, this is quite a weak proposition.

However, Eric Rise created an effective way to avoid this problem, called the lean start-up method. It introduces testing customers and marketing before products are made [20]. The idea of lean start-up is shown in figure 6.

Lean startup cycle

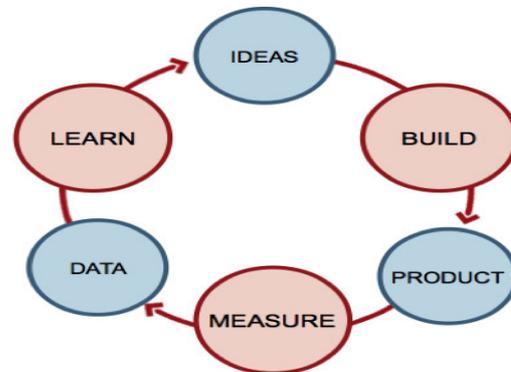


Figure 6. Lean start-up processing [17]

Instead of making products directly, a start-up company can build models of products. A model can be a picture on the paper, a model made of plastic or even Lego bricks. There is no need to put all the functionalities into the model, not only a simple appearance is enough. After that, the entrepreneur goes out of the office to test the model of the products. This is a very important point in the lean start-up called “get out of the building”, which means showing the model to some potential customers and asking advices and feedback from them. It is also possible to ask some questions like “Would you like to buy this product in the future?” or “If you want to buy this product, how much would you like to pay for it? ”. This process enables the starting business to validate whether the idea would have potential and enough customers in the future. If a start-up sets mostly positive answers, it can improve the product according to customers’ feedback. Repeating this process several times will make the most suitable products for the market [21].

2.3.2 Business-to-business

If the company focuses on selling its service or products to another company or business, it is using the business-to-business model. Business-to-business has benefits from the raw materials transfer to the final product. This model of subcontracting is widely used in automobile manufacturing. Nowadays it is also used in e-commerce.

There are mainly two different parts in e-commerce, company websites and product supply [22]. Company websites can offer service or information as their products.

2.3.3 Business-to-Customers

Business-to-customers is a type of commerce mostly based on supplying products and services to customers. For example, people going to a supermarket for shopping is an example of the typical traditional business-to-customer model. In addition, when the Internet rapidly grew in the 1990s, hundreds of thousands of domain names were registered. Online shopping plays a very important role in the market and it is accepted by more and more people [23]. This is an innovative way of using the business-to-customer model.

2.4 Overview of the queuing system

As the world population increases, more and more buildings, cars, and congestion problems will appear. Due to the large numbers of customers, public places will become crowded and chaotic.

In the middle ages, instead of shops every town had markets. If people wanted to buy or sell something they would go to the market. Due to the increase of the number of population and technological development and industrialization, permanent shops became more common. When the shops become larger, the shop owner would not deal with all the business at the same time but had to let customers make a linear queue and to provide service to them according to the queue [23].

However currently, the traditional linear queuing way cannot solve the queuing problems effectively due to the larger number of customers. Therefore, service providers have created an electronic queuing system. The working process is shown in figure 7.

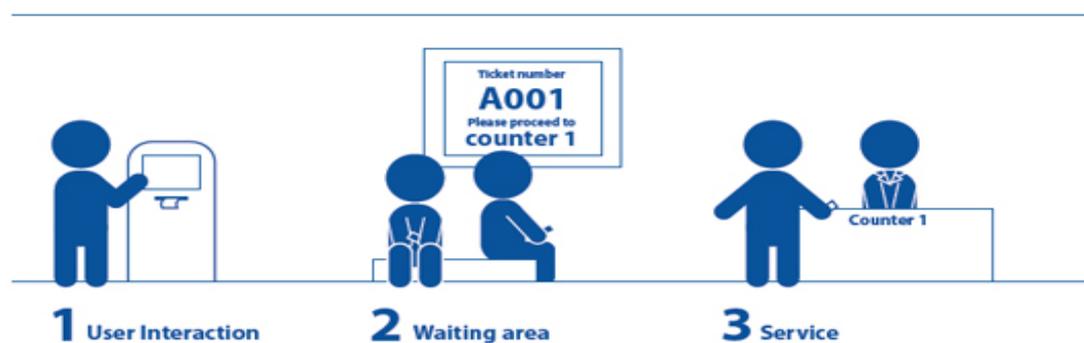


Figure 7. Current queuing system [24]

This innovation of a queuing system has several new features compared with the traditional linear queuing method where customers go to a service point, they need to go to the service desk and stand in front of the service desk directly. Instead of that, in the new system there is a machine always placed in the doorway of service points or offices. There will be some buttons on the machine showing services offered at the service point. Customers can press the corresponding buttons for what kind of service they want. After that there will be a receipt coming out from the machine. This receipt shows the name of the service that the customer selected and the queue number of the customer. This receipt is also a proof to prove they are in the queue when their turn comes.

After getting the queue number from the queuing machine, customers can stay at the service point and they will not need to stay in the line all the time. They can try some new devices or do some purchases in shop. However, they have to stay at the service point and check the current number shown on the screen. When the current number is the same as the number they have on the receipt, it means it is time to do the service for this customer. At that moment, the customer can go to the counter and get some service from the service point staff.

Both of two queuing ways are still used now due to their different benefits. The linear queuing method is suitable for situations where the number of customers does not significantly exceed the capacity of the service staff, and crowds are less frequent. The electronic queuing system has more features compared with the normal linear queuing method. This way is mostly used for service points, which cannot control customers' serving time. Customers do not need to stand in front of counters until their turns. This way helps the service point manage customers so that the service point does not become too crowded and disorderly. However sometimes the serving time for each customer is not fixed, customers do not know how long they have to wait. Especially for a service point where there is a large number of waiting customers, the electronic queuing system cannot fully solve the queuing problem. Customers have to stay at the service point until their turn. The reason is that they go somewhere to do something else and go back to the service point, they will risk missing their turns. That means they have to take the queue number again and queue again. The reason is they cannot know the current queue number without checking the screens at the service point. If this problem is solved, there will be more satisfied customers. On the other hand, the staff working at the service point will be more relaxed and have less pressure to face impatient customers.

3 Design and implementation

3.1 Introduction to the remote queuing system

The remote queuing system was designed for reducing customers' queuing time. This queuing system is based on the existing electronic queuing system where some features have been added. The remote queuing system helps customers multitasking while they are waiting at the service points. By using the remote queuing system customers do not need to physically wait at the service point. This feature can improve customer satisfaction. Besides that, the staff will work more effectively and provide better service for each customer. In this application, customers are able to use their phone access to the web application so that they can check the current number that is shown on service point screens. Because the application was implemented as a web site, there is no platform or phone type limitation. Any phones which can access the internet are able to use this web application.

3.2 Implementation and design of remote queuing system

The remote queuing system is an improvement to the current queuing system. As people know the current electronic queuing system, customers should first take the queue number when they visit the service point. In this project, after a person has taken the queue number, there will be a QR code on the backside of the queue number receipt. The customer can use a phone to scan the QR code, which will lead him or her to the web application that is designed for the client-side. This web application will show the current number on the particular service point screen. Customers will know how many people are waiting before them and estimate the waiting time by themselves by checking the mobile.

To achieve this goal, the project design was separated into three parts: a lower computer connection part, server application part and client-side web application part, which all cooperate with each other. First, the lower computer that is usually the service point staff's computer should connect with the internet and the server. When the next customer comes, the staff can refresh the current number on the server by clicking the software button on their own computers. The server always pushes the new current

number to each lower computer to make sure that all computers at the service point will have the same current number.

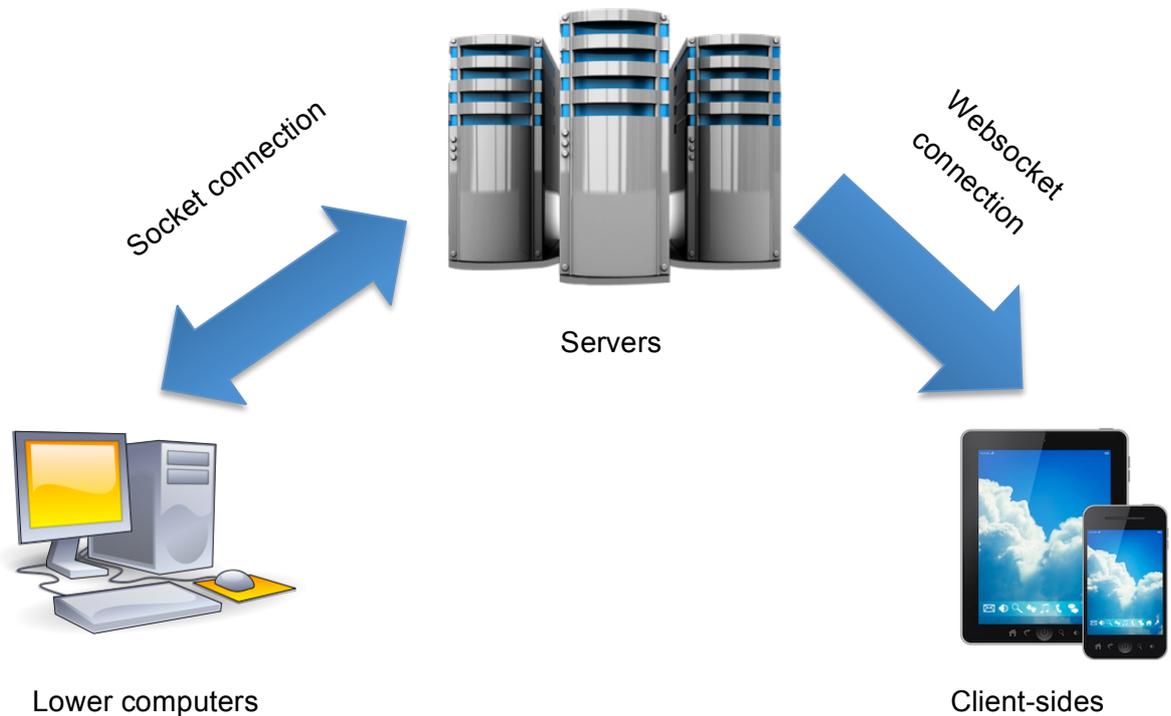


Figure 8. The remote queuing system architecture

The communication between servers and lower computers is duplex communication. At the same time, after customers scan the QR code, a connection between the customers' devices and servers will be established. This connection uses the websocket protocol. When the current queue number refreshes on the server, it will be automatically pushed to customers' devices. That means the customers can check the current number anytime without refreshing their phones.

3.2.1 Design of data packages

In this project, the lower computer will send data to the server in order to update the server in real time. However, the data must be in a package format that contains details about the transmission according to the protocol requirements.

In this project, the data package has been design as table 1 shows.

Table 1. Data package

Header	Length	Name	ID	Data 1	Data 2	Data 3	CRC
0xFB	1 Byte	30 Byte	1 Byte	2 Byte	2 Byte	2 Byte	2 Byte

The package contains the header, length, name, ID, data 1, data 2, data 3 and CRC parts. Each of them plays different roles in the package. The header is working to inform the server side that this is relevant information. Because there will be several different service points, each service point has a name and a corresponding current number. So the name part in this package is going to represent the name of the service points. The name has to be designed to act as a unique ID for the package. Therefore, each service point has a unique ID, which is like an address of the particular part on the server. The lower computer is like a postman. After reading the ID it will know where to send the package.

In this project, three data spaces have been reserved. All the data parts could be used to send the current number and update the data on the servers. In the project, only one data part was used to send the current number. The other two parts are the backups of the data. The two backup data parts could have different functions in the future. For example, when there is enough customer data accumulated in the database the estimated waiting time could be calculated. The backup data part can be used to send the estimated waiting time. This design gives the application enough space for further updates and improvements.

3.2.2 Implementation of lower computers connection

The lower computers connect with servers by using a socket protocol. In this project, the server opens a port (10005) waiting for the data to send from the lower computer via this port, so that the server will receive real-time updating from lower computers. However this number of the port is not fixed.

There are mainly two parts of this function. First the lower computer makes a socket connection with the server via the particular port.

```

$address = '127.0.0.1';
$port = 10005;
//Establish a port
if( ($sock = socket_create(AF_INET, SOCK_STREAM, SOL_TCP)) === false) {
    echo "socket_create() failed :reason:" . socket_strerror(socket_last_error()) . "\n";
}
//Bind
if (socket_bind($sock, $address, $port) === false) {
    echo "socket_bind() failed :reason:" . socket_strerror(socket_last_error($sock)) . "\n";
}
//Listen
if (socket_listen($sock, 5) === false) {
    echo "socket_bind() failed :reason:" . socket_strerror(socket_last_error($sock)) . "\n";
}

```

Listing 1. Socket connection.

The code of the socket connection is shown in listing 1. There were three parts to test whether the connection was successful and make sure that the connection was established. In the project, the 10005 port has been opened as the receiving data port. `$sock = socket_create(AF_INET, SOCK_STREAM, SOL_TCP)` function will create a socket connection and return a socket resource. There are three parameters in the function, namely domain, type and protocol. The domain parameter presents the type of the protocol family used in this function. `AF_INET`, `AF_INET6` and `AF_UNIX` are suitable for being a parameter in this function. The project uses `AF_INET` as the parameter, which is an IPv4 based protocol. TCP and UDP also belong to this protocol family.

The type parameter shows the type of communication ways. It is a significant part in this project. The available parameter options are listed in table 2.

Table 2. Function parameters

Type	Description
SOCK_STREAM	Provides sequenced, reliable, full-duplex, connection-based byte streams. An out-of-band data transmission mechanism may be supported. The TCP protocol is based on this socket type.
SOCK_DGRAM	Supports datagrams (connectionless, unreliable messages of a fixed maximum length). The UDP protocol is based on this socket type.
SOCK_SEQPACKET	Provides a sequenced, reliable, two-way connection-based data transmission path for datagrams of fixed maximum length; a consumer is required to read an entire packet with each read call.
SOCK_RAW	Provides raw network protocol access. This special type of socket can be used to manually construct any type of protocol. A common use for this socket type is to perform ICMP requests (like ping).
SOCK_RDM	Provides a reliable datagram layer that does not guarantee ordering. This is most likely not implemented on your operating system.

In table 2 the options can be compared, and SOCK_STREAM is the most suitable for this project situation. It provides orderable, stable full-duplex data transmission. The protocol is the specified domain name used in the function. This project use SOL_TCP, which means the desired protocol is TCP or UDP [24].

The function of socket_create will return the source of the socket, and it will present the status of the socket creation. If the creation fails, “echo "socket_create() failed :reason:" . socket_strerror(socket_last_error()) . "\n";” will be executed. It will print the reason for the failure.

The second step binds the socket with the name socke_bind() function that can bind the target address with the socket. This function should be done before a connection is established. It is used in the project in the following manner: socket_bind(\$sock, \$address, \$port). In the socket creation part, variable “sock” has been created and carries on the information of the socket. In this function, “sock” has been used as the parameter. In addition, the variable address given as “\$address = '127.0.0.1'” and “\$port =10005”. The function binds the socket to port 10005 with address “127.0.0.1” [25]. After creating a socket and binding it with a name, the port is ready to listen to the incoming data. Socket listening function has been used like socket_listen(\$sock, 5), number 5 means that the maximum backlog can be made in the connection [26].

```
do {
  //Get a link
  if (($msgsock = socket_accept($sock)) === false) {
    echo "socket_accept() failed :reason:".socket_strerror(socket_last_error($sock)) . "\n";
    break;
  }
  //Send welcome to client-side
  $msg = "server send:welcome";
  socket_write($msgsock, $msg, strlen($msg));
  socket_write($msgsock, $talkback, strlen($talkback)) {
    if (false === socket_write($msgsock, $talkback, strlen($talkback))) {
      echo "socket_write() failed reason:" . socket_strerror(socket_last_error($sock)) . "\n";
    }
  }
  $buf = socket_read($msgsock, 8192);
  //$buf ="0002 chinachina 43 09 253";
  $val = explode(' ', $buf);
  $data = array(
    'code'=>$val[0],
    'name'=>$val[1],
    'data1'=>$val[2],
    'data2'=>$val[3],
    'data3'=>$val[4]
  );
  M('device')->where("code=".$data['code'])->save($data);
  //socket_close($msgsock);
} while(true);
//关闭socket
socket_close($sock);
```

Listing 2. Socket implementation.

Listing 2 shows the connection and data communication parts. Function `socket_accept($sock)` will accept the socket connection and get the source from the connection. After getting the information successfully, the server will send the successful signal via `$msg = "server send:welcome"`. Function `$buf = socket_read($msgsock, 8192)` reads the information in the socket and assigns it to variable `$buf`. Parameter 8192 in the function means reading the length of the information. After this step, the data is stored in the buffer and it is time to push the buffer to database by using `$data = array ('code'=>$val[0], 'data1'=>$val[2], 'data2'=>$val[3], 'data3'=>$val[4]);`

3.2.3 Implementation of service point management application

The service point management application is used by the administrator to create accounts, and monitor and manage the queuing states at each service point.

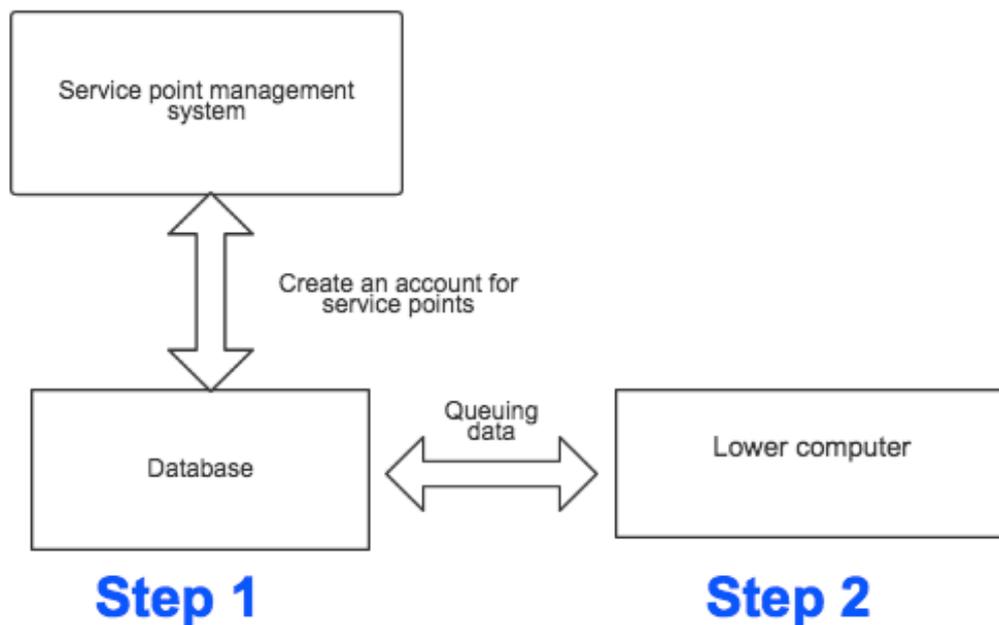


Figure 9. Communication with database

As figure 9 shows, the administrator should use server queuing management to create a service point name, ID and corresponding QR code. After that, it will push them into database so that the lower computer will find the data target when they send the data to server. For this reason, the administrator should use the service point management

application to create an account for the service point on the lower computer at the service point that is sending the data.

There are mainly three functions at the service point management application: the create account for service points, delete the service point account and edit the service account. A service point account contains the service point name, the unique ID, and a QR code. It is also possible to check the queuing status of the service point in this system.

```
<?php
include '../lib/core.php';
if(isset($_POST['name']) && isset($_POST['pass'])){
    $name = trim($_POST['name']);
    $rs = M("admin")->field('username,password')->where("username='$name'")->getOne();
    if($rs['password']!=md5($_POST['pass'])){
        echo alert('username or password is wrong', '../m.php');
    }else{
        $_SESSION['username'] = $rs['username'];
        go('main.php');
    }
}
```

Listing 3. Service point login page

Because of the security protection, the administrators always need to log to the service point management system with valid credentials. Listing 3 shows the code of the login page. Each username will be led into a different service point management main page. Each username corresponds with a different database so that different administrators manage their own systems. The statement `$rs['password']!=md5($_POST['pass'])` compares the password input by clients with the passwords in the database. If the password is correct, it will go to the main page. Otherwise, the alert 'username or password is wrong' comes.

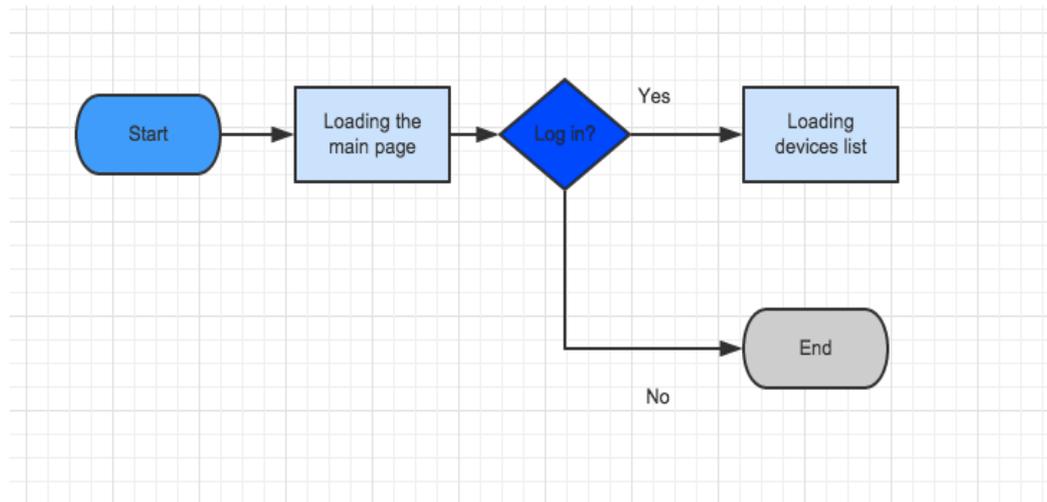


Figure 10. The working processing of the main page of the service point management system

The processing of the service management system main page is displayed figure 10. On the main page function `logout()` will lead the user to log out of this page. The statement `Edit Device` use to checking the device list. After click the 'Edit device', the page will come to the devices list page. The code of the device list page is shown in listing 3.

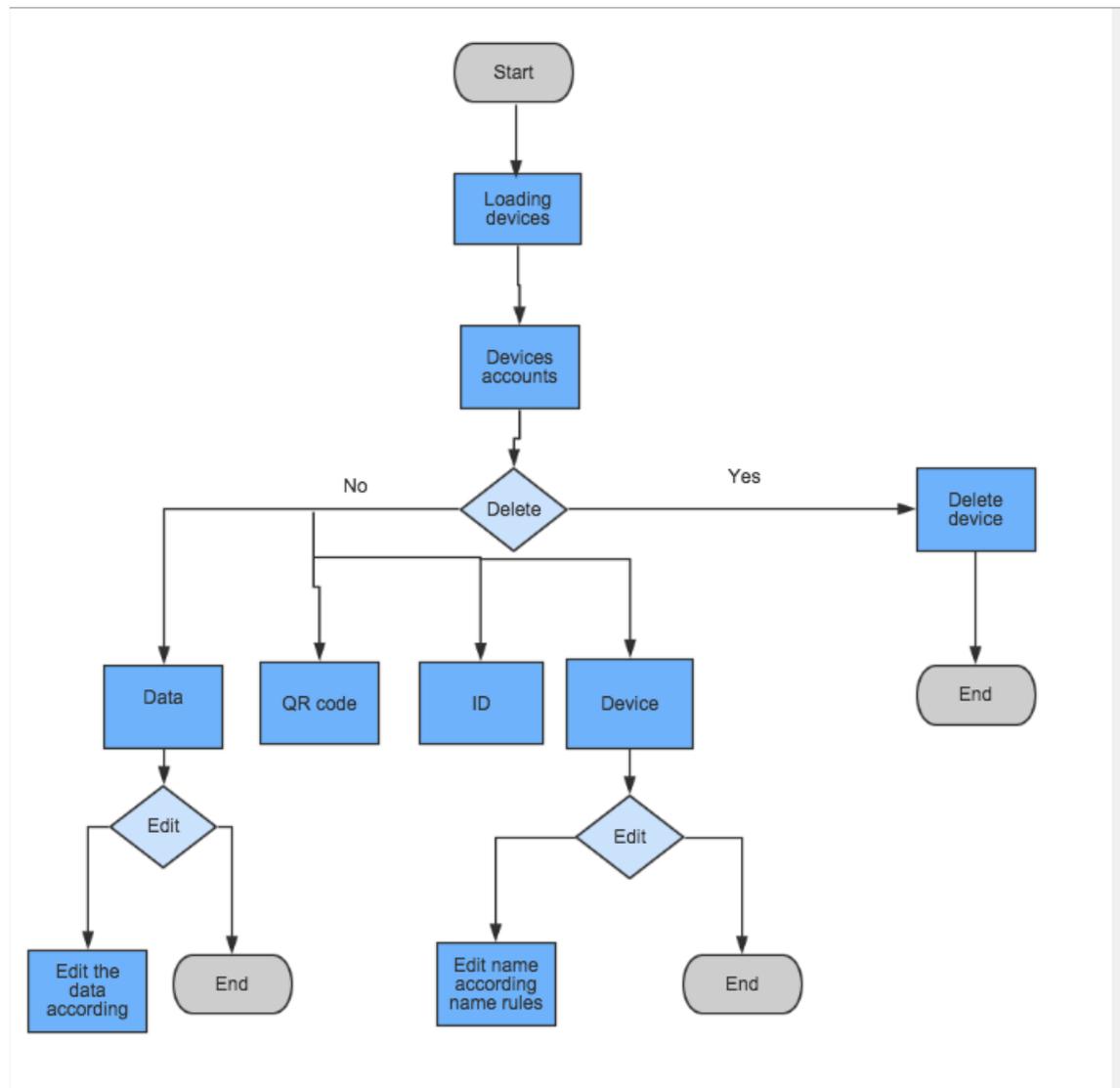


Figure 11. The working processing of devices list of the service point management system

Figure 11 presents the working processing of devices list of the service point management system. The function `del(obj)` is used for deleting service point accounts. The statement `<table class="m-table m-table-row">` shows each item (name, ID, QR code, data1, data2, data3) on the table. Every time, the page loads the list of devices at the service points by accessing the database and reading the data from database. The function for each (`$rs as $key => $val`) reads the service points items from the database in order to ensure that the latest data is displayed. Administrators are able to edit device names and data in the account according to the package rules corresponding to the data package sent from lower computers. The QR code is not available for editing.

Each device has a different unique QR code, which the end customer can scan and the QR code will lead the customer to the client-side web application.

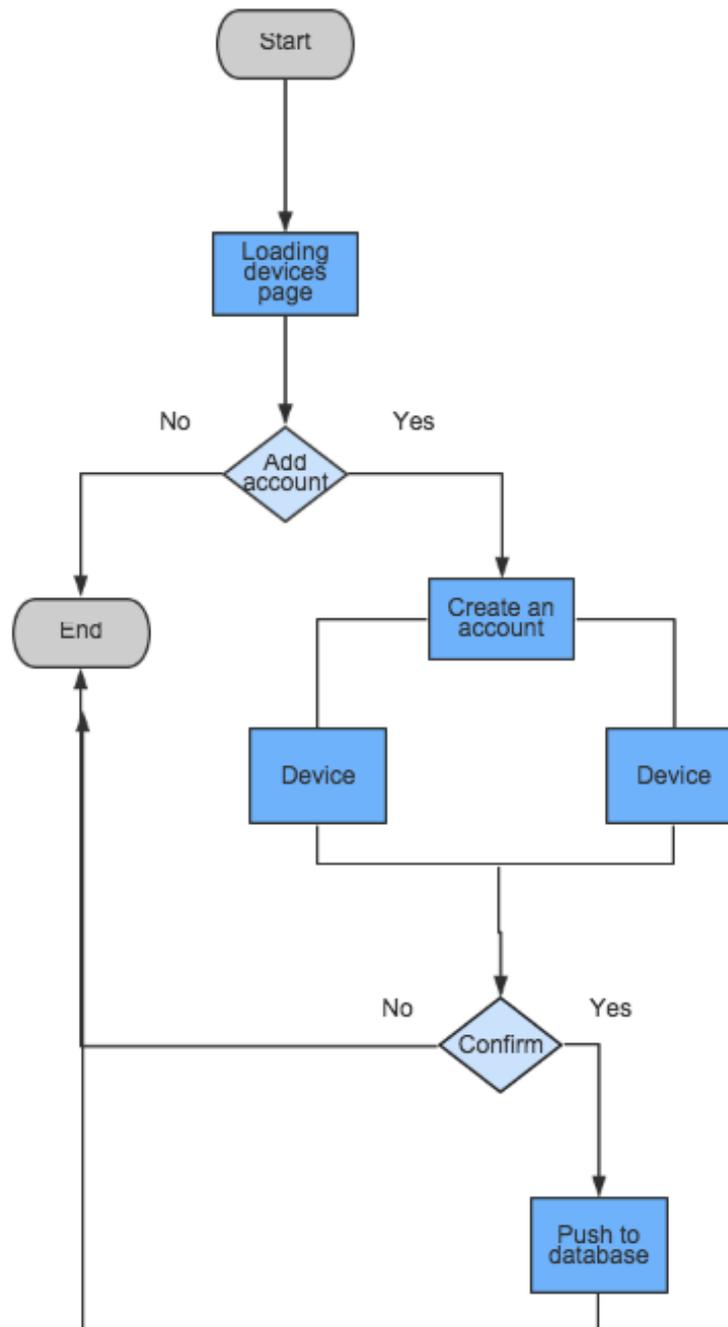


Figure 12. The working processing of creating a new account of the service point management system

When an administrator creates a new account, he is directed to add a new device page, as in figure 12. Listing 3 shows the code of adding a new device page. The tag

</div><form action="" method="post"><div class="form-item"> is used to ask the details of the devices, name of devices and ID of devices. All the device data will be put into the data base by users by clicking the 'submit' button. The tag code <button type="submit" target-form="form-horizontal" class="btn ajax-post_custom submit-btn">Submit</button> shows how the submit button works.

3.2.4 Client-side real-time web application

In this project, there is a web application for customers to check the real-time current number. To make sure the data is updated in realtime, this web application is using the websocket protocol. For this reason, customers do not need to refresh the website when they want to check the real-time current queue number. The website will do the real-time updating automatically. This websocket listening application will run on the server once it receives the new current number from the server-side, and it will push data to the client side automatically

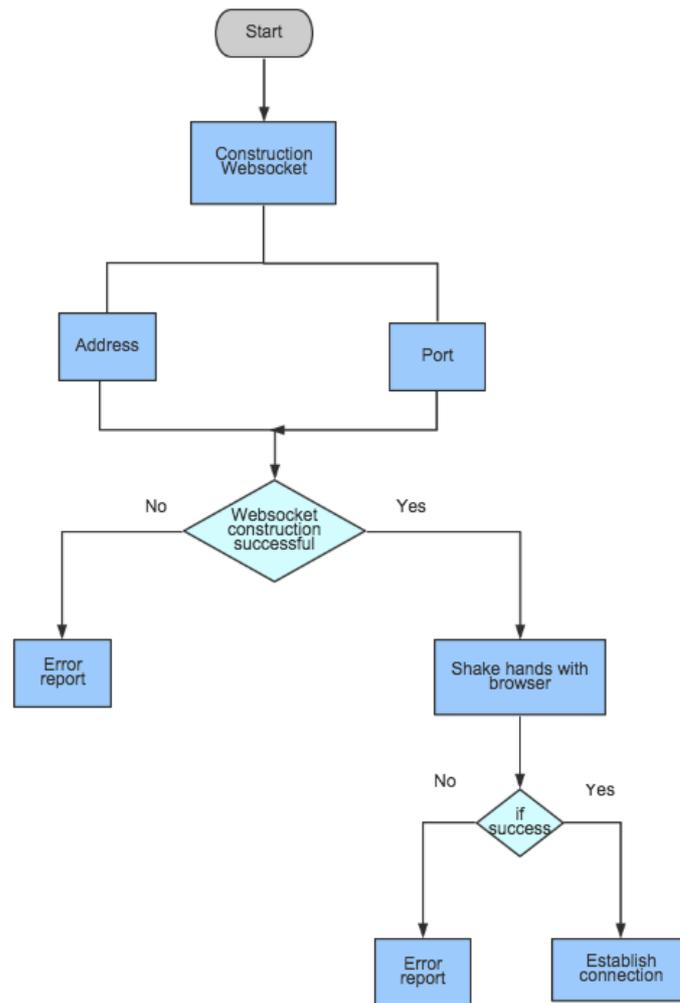


Figure 13. Working processing of the websocket protocol

Figure 13 shows the processing flow of the websocket construction and handshakes on the server side. The first websocket should be constructed by a construct function in the code. It mainly constructs the log part of the websocket and binds the websocket to the particular address and port. If the construction of the websocket fails, it will come into the error report parts, finding out the problem and printing it. After constructing the websocket, it is time to shake hands with the browser and prepare for establishing a connection. In order to ensure security, the key in the handshaking function will be encrypted with the `base64_encode()` statement that is used to encode the key, so that it is not in readable letters.

If the server shakes hands with the browser successfully, the connection will be established. The data can be sent in real-time from the server-side to the client-side. How-

ever the data will not be sent in readable type. All data is sent encoded by hexdec() function, the return value of this function is the number type. Corresponding with the encode function, there is a decode function to decode the received data. The decode function will make data readable so that it can directly show on clients' devices.

There is a client-side user interface for customers checking the current number and information of the service points. To make the application more user-friendly and easy to use, a big current number is displayed at the centre of the device. On top of the device there is service point name and current time. All the web application codes are written by HTML5 to make it a responsive web design application.

4 Business plan and implementation

4.1 Overview of the business plan

The remote queue management system is applicable around the world, and the ultimate goal is to sell the service to the business sector as well as the consumer sector. To achieve this goal, there are several steps needed to be considered first.

The project has used the lean start-up principle in order to prove the value of the project and determine the customers. All three main customer segments are applicable for our business: business to business (B2B); business to government (B2G) and business to consumers (B2C). The first step is to sell the service to businesses, secondly to government-owned companies, and later on we will develop an advanced service to be marketed directly to consumers. For smooth business communication purposes, the project has registered a company name 'Qer'.

4.1.1 Lean start-up implementation

The lean start-up method helps saving costs and validating the idea, which has the market value. Instead of making the product directly, the idea was first tested according the lean start-up method. First the idea was drawn on the paper, which is a kind of a simple user interface for the client-side users. Then a survey was made face-to-face to end-user customers by showing them the user interface on paper and explaining how the web application works.

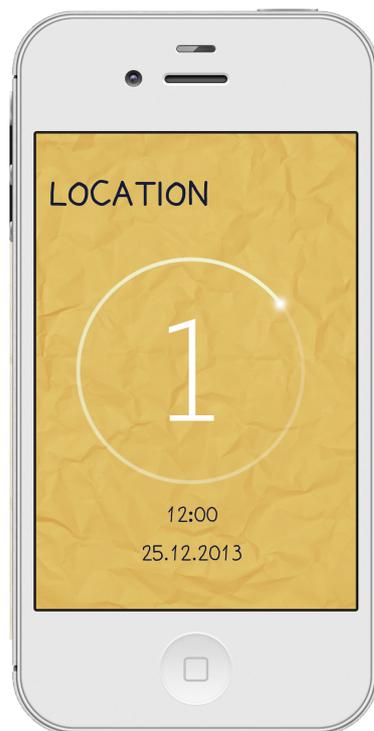


Figure 13. First version of user interface

The survey also asked clients' willingness to use the application and the price they want to pay for the application. According to the survey, 91% of the respondents wanted to use this web application if it was available. In addition, even 57% of respondents wanted to pay for the service. Besides that, surveys were also given to the staff at some service points and feedback of this product was asked from them. All of them agreed with this idea, and they thought this product would improve their quality of service. All results above proved that the idea of the remote queuing system was worthy and acceptable by customers.

4.1.2 Business plan

By surveying customers' opinions on queuing times, the team figured out that service providers were expected to give fast customer service, but they were lacking the means for it. By using the segment business to business (B2B) services, we would be able to create a sustainable and growing source of revenue as we could either charge a monthly service fee that would covers the maintenance and use of service, or a fee that would covers only the use of service. The B2B product was fairly easy to implement for Qer. As the idea and need for the service was clear for the businesses and

the consumers, Qer was able to launch the application and hopefully start making revenue fast. Customers in the B2B sector are all offices, banks, shops, etcetera that experience queues. The most obvious ones are operators, banks, post offices and pharmacies. The remote queue management system is also applicable for stores, cafeterias, restaurants and airports.

The government seems to be the elephant in the room. Consumers are crying for help in the government sector; there are big problems with huge queues. Customers have to queue for hours at tax-offices, health-care centers, Kela-offices and the like. It seems like the perfect starting point, but business-wise, it will take time to get the Elephant moving, due to a slow decision making process. This is why the government is only our second target for implementing the service. When the company will get deals with government-owned places, a large share of the market will be covered and easy to apply it after implementing it in one government owned place.

After surveying the queuing problems and expectations, this was a surprise that consumers would be willing to pay for the service as well. Moreover, they were willing to pay for our application above the average price compared to the application prices in general. As soon as the business is in a growing position, the company will implement the service application for consumers, and with more features than the web-based service. B2B product development and sales will gain good market data. It is easy to target the application for a range of consumers and get extra revenue from this huge market. The consumer pays once for the service, when they purchase the application, and the use of this service is free for the consumer. Businesses pay for the application maintenance and development fees. The company gets more pleased customers by creating extra value for the consumer with the application.

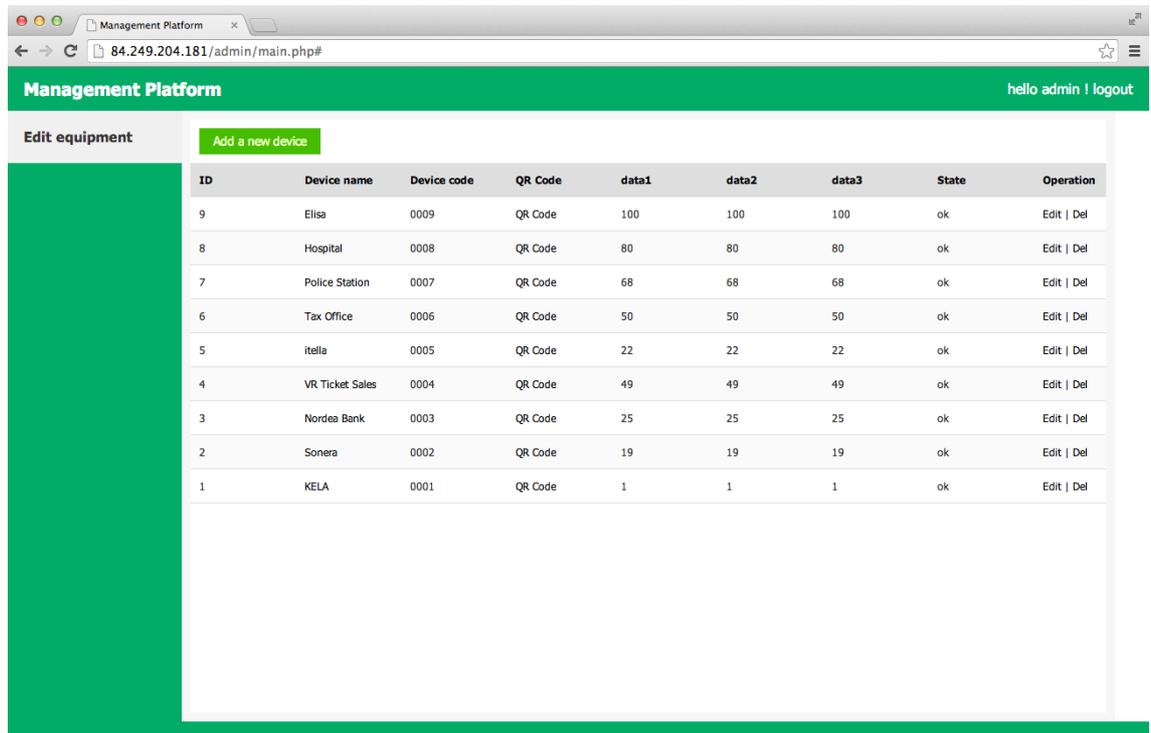
Besides that, there are countries that are experiencing a lot more problems with queuing than Finland. Obviously countries and cities with huge populations have more places where long queuing times are a huge problem. For example, in Brazil, there are professional “queuers”, who queue for money, because the waiting times are ridiculous. In China, The Bank of China has developed a text message service for customers so they will not need to wait inside, because the waiting times are very long. The problem with this service is accuracy, but it shows there is a real problem with queues, which needs to be solved. Finland may not have a huge market for this queuing service, but it

provides good start-up opportunities and a platform. For this reason, Finland will be the first product test place.

5 Testing and results

5.1 Application testing

In order to test the application both on the server-side and client-side, the experiment should be simulating a real queuing case. First, the administrator should create a service account on the service point management system on the server side. The account should contain the service point name and unique ID, which is shown in figure 14.

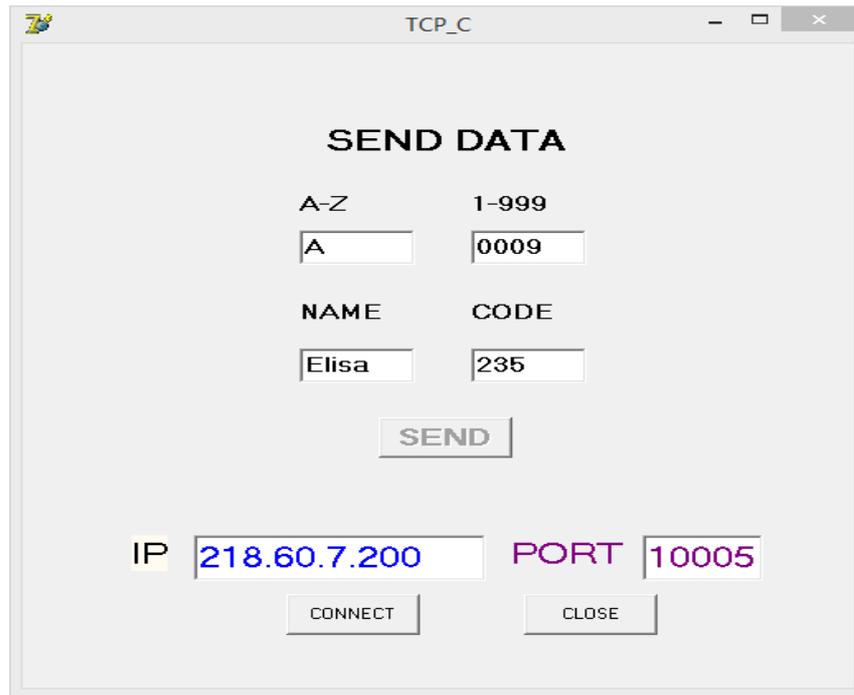


The screenshot shows a web browser window with the address bar displaying '84.249.204.181/admin/main.php#'. The page title is 'Management Platform' and the user is logged in as 'hello admin ! logout'. The main content area is titled 'Edit equipment' and features a table of service points. A green sidebar on the left contains a button labeled 'Add a new device'.

ID	Device name	Device code	QR Code	data1	data2	data3	State	Operation
9	Elisa	0009	QR Code	100	100	100	ok	Edit Del
8	Hospital	0008	QR Code	80	80	80	ok	Edit Del
7	Police Station	0007	QR Code	68	68	68	ok	Edit Del
6	Tax Office	0006	QR Code	50	50	50	ok	Edit Del
5	itella	0005	QR Code	22	22	22	ok	Edit Del
4	VR Ticket Sales	0004	QR Code	49	49	49	ok	Edit Del
3	Nordea Bank	0003	QR Code	25	25	25	ok	Edit Del
2	Sonera	0002	QR Code	19	19	19	ok	Edit Del
1	KELA	0001	QR Code	1	1	1	ok	Edit Del

Figure 14. Service point management system

After creating the account of the service points, the testing used the first account on the device list to test the web application. The second step was to simulate how the staff at the service point would send the current number to server-side and at the same time the customers used their phone to monitor the number change in realtime. To verify the number changing in realtime, the service point side first made a connection with the server and sent a number and the corresponding device name to the server as figure 15 shows.



The screenshot shows a window titled "TCP_C" with a "SEND DATA" section. It contains two columns of input fields: "A-Z" and "1-999". The "A-Z" field contains "A" and the "1-999" field contains "0009". Below these is a "NAME" field with "Elisa" and a "CODE" field with "235". A "SEND" button is positioned below the "NAME" and "CODE" fields. At the bottom of the window, there are "IP" and "PORT" fields. The "IP" field contains "218.60.7.200" and the "PORT" field contains "10005". Below these fields are "CONNECT" and "CLOSE" buttons.

Figure 15. Sending a new current number

After sending the current number to the server, it was time to use the mobile device to monitor the number and test whether it was changing properly in real-time.



Figure 16. Client side user interface

Scanning the QR code corresponding with the web application was automatically generated when the administrator created an account for the service point. The device was directed to the client-side web application, and the customer could use this page to check the current number in the queue.

5.2 Business testing

There are many competitions and exhibitions for start-up companies conducted in the Helsinki area. They provide opportunities for customers to learn about this product and also to push this product to the market. The project and business idea were first proved in the Business Competition Model and won the second prize. It also proved that the method of lean start-up is an effective way for saving the cost and finding out the problems of the project. Moreover, the project was taken into the JA-start up competition and was awarded the first prize and 15,000 euros as the company's initial funding.

The project has been taken to several telecommunication operator service points (e.g. Sonera, Elisa) for testing and getting feedback. Telecommunication operators are interested in this project and willing to cooperate with Qer. The price of the service is still under negotiation between Qer and the telecommunication operators.

6 Discussion

6.1 Technical problems and solutions

There has been a connection problem caused by the websocket protocol. Because the websocket is a new protocol and technology, it does not yet work perfectly on all browsers. Although it can send data in realtime and costs less bandwidth, it is however not stable on some non-mainstream browsers. As tested with some browsers, the number does not change in realtime. Another example problem is testing with a mobile the case when the customer puts the phone in the sleep mode and awakes the phone again. Then disconnection situation will occur.

As this problem does not happen too often, the reconnection function can solve it. However, it is not a particularly smart solution, and it will also consume memory usage of the suitable browser. Designing a header to test the type of browsers and loading different codes on the page will solve this problem.

6.2 Business problems and solutions

The remote queuing system is an improvement of the current queuing systems. The service points prefer to update their current ticket machines instead of buying a totally new machine again. On the other hand, this will causes a business problem. The staff working at the service point will push the current number to the server. That means the computer in the servers point have to push the current number to our server. However the rights of the computer at the service points belong to the current queuing service company. In the Finnish market, there are mainly two queuing service companies: EWQ and Qmatic. EWQ is a smaller one, established in Espoo. The team tried communication with EWQ about sharing the queuing number data. However, most of their queuing machines are quite old, and it is impossible to connect with the internet. If we want to receive their data, we will have to buy internet models from EWQ and plug them on the ticket machine. However the model is really expensive. Until now the team is still negotiating with EWQ about sharing the data.

7 Conclusion

Customers are increasingly eager to use mobile devices to deal with various kinds of everyday issues and problems. A long queuing time is a real-life problem, which frequently irritates people. Customers expect a fair queuing system, but on the other hand they also want to have a less waiting time system. The waiting time also means wasting time in some occasions.

The goal of this project was to create a remote queuing system, which would allow customers to do multitasking while they are queuing. The system will make customers free to leave from the service point and reduce the queuing time at the service point. The project on the technical side is fully functional. However, this is not entirely a technical project, as the project has ramifications on people's daily lives. The project involves a number of business issues, as well. It should benefit both the service provider and the end-use customer.

For future development, the system will be more customized. Customers would save their particular queue numbers in their mobile devices. In addition, customers can store the receipt of the queuing number by taking a photo with this application. The application will read the photo of the queuing number and when only a few people are before in the queue, the application will send a notification. All the designs and improvements to the application will make people's lives easier and more convenient.

References

1. Most popular APP store categories. [Online]. Statista. Accessed April, 2015
URL: <http://www.statista.com/statistics/270291/popular-categories-in-the-app-store/>
2. Danyl Bosomworth, Mobile marketing statistic 2015. [Online]. Smart insights. Accessed April, 2015
3. Hai Sun. Design and implementation a remote queue system. [PDF]. Metropolia University of Applied Science. Accessed April, 2015
4. Communication protocol. [Online]. Techopedia. 2015.
URL: <http://www.techopedia.com/definition/25705/communication-protocol>. Accessed April, 2014
5. H. Gilbert. Introduction to TCP/IP. [Online]. Yale. USA. FEB 1995.
URL: <http://www.yale.edu/pclt/COMM/TCPIP.HTML>. Accessed April, 2015
6. Gorry Fairhurst. The User Datagram Protocol (UDP). [Online]. University of Aberdeen. United Kingdom.
URL: <http://www.erg.abdn.ac.uk/users/gorry/course/inet-pages/udp.html>. Accessed April, 2015
7. Joe Hildebrand, Craig Kaes, David Waite. XEP-0025: Jabber HTTPing. [Online]. June 2009.
URL: <http://xmpp.org/extensions/xep-0025.html>. Accessed April, 2015
8. Adedumola Afolayam. Middlesex University. 4G WIRELESS COMMUNICATION TECHNOLOGIES – Connecting People. [Online]. Devices and Information Hendon. London. UK. 2015.
URL: http://www.academia.edu/7633845/4G_WIRELESS_COMMUNICATION_TECHNOLOGIES_Connecting_People_Devices_and_Information
9. The Most Reliable and Cost-Effective 3G Solution for Remote Device Communication. [Online]. MOXA. February 2014
URL: http://www.moxa.com/newsletter/connection/2014/02/feat_02.htm. Accessed April, 2015d
10. Saeed Awamleh. What is polling?. [Online]. What.com. April 2005.
URL: <http://whatis.techtarget.com/definition/polling>. Accessed April, 2015
11. Joe Hanson. What is HTTP Long Polling?. [Online]. Pubhub. December 2014.
URL: <http://www.pubnub.com/blog/http-long-polling/>. Accessed April, 2015
12. Mozilla developer network. Ajax. [Online]. January 2014.
URL: <https://developer.mozilla.org/en/docs/AJAX>. Accessed April, 2015
13. Ajax for client-server communication. [Online]. Ebase.
URL: http://www.ebasetech.com/ufs/doc/Ajax_communication.htm. Accessed April, 2015
14. Peter Lubbers, Frank Greco. HTML5 Web Sockets: A Quantum Leap in Scalability for the Web. [Online]. WebSocket.org. 2013. USA.

URL: <https://www.websocket.org/quantum.html>. Accessed April, 2015

15. WebSockets vs Server-Sent Events vs Long-polling. [Online]. Frankfurt. Germany. 2013.

URL: <http://dsheiko.com/weblog/websockets-vs-sse-vs-long-polling/>. Accessed April, 2014

16. Ken Ramirez. Build Ajax-base website with PHP. [Online]. IBM developer works. September 2008.

URL: <http://www.ibm.com/developerworks/library/wa-aj-php/>. Accessed April, 2015

17. Darshan Gajanan Puranik. Real-time monitoring of distributed real-time and embedded systems using web. [Online]. Purdue University. 2012.

URL: <http://docs.lib.purdue.edu/dissertations/AAI1548053/>. Accessed April, 2015

18. Patrick van der Pijl. Bestseller Business Model Generation. [Online]. Business Model. 2009.

URL: <http://www.businessmodelsinc.com/method/our-book/>. Accessed April, 2015

19. Eric Rice. Lean start-up. [Online]. The lean startup.

URL: <http://theleanstartup.com>. Accessed April, 2015

20. Steve Blank. Why the lean start-up changes everything. [Online]. Harvard Business Review. USA. 2009.

URL: <https://hbr.org/2013/05/why-the-lean-start-up-changes-everything>. Accessed April, 2015

21. Laurence McCahill. 10 things I've learnt about lean startup

. [Online]. May 2013.

URL: <https://welovelean.com/2013/05/01/10-things-ive-learnt-about-lean-startup/>. Accessed April, 2015

22. Katherine Arline. What is B2B?. [Online]. BusinessNewsDaily Contributor. January 2015.

URL: <http://www.businessnewsdaily.com/5000-what-is-b2b.html>. Accessed April, 2015

23. Elaine J. Hom. What is B2C?. [Online]. BusinessNewsDaily Contributor. September 2013.

URL: <http://www.businessnewsdaily.com/5085-what-is-b2c.html>. Accessed April, 2015

24. Web socket create. [Online]. Php.net. 2015

URL: <http://php.net/manual/en/function.socket-create.php>. Accessed April, 2015

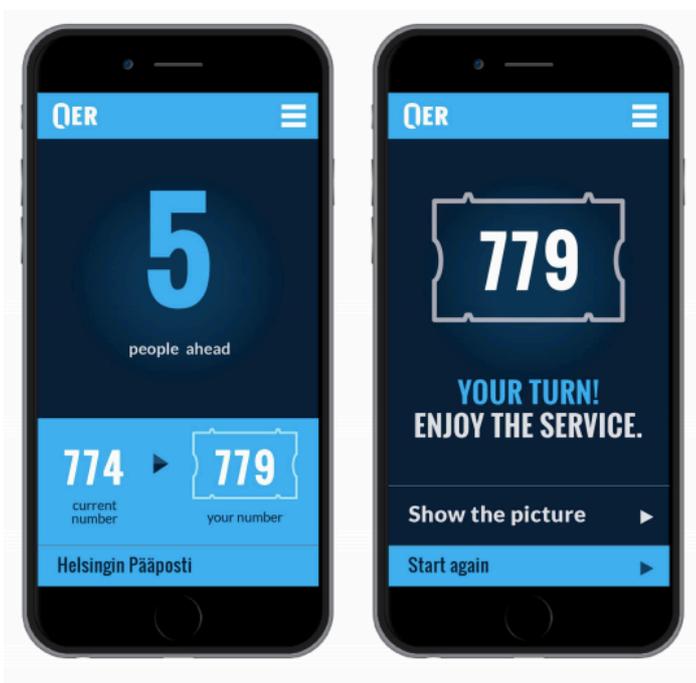
25. Web socket bind. [Online]. Php.net. 2015

URL: <http://php.net/manual/en/function.socket-bind.php>. Accessed April, 2015

26. Web socket listen. [Online]. Php.net. 2015

URL: <http://php.net/manual/en/function.socket-listen.php>. Accessed April, 2015

Future development on IOS platform



Qer Home page

QER HOME PRODUCT CONTACT

FREE YOUR CUSTOMERS TO FOCUS ON SHOPPING QUEUING VIRTUALLY.

Queuing virtually.

Instead of waiting in the actual line, free your customers to leverage their time at your place. Customers feel being cared for and valued while you get a chance to sell more.

Our solution

Since not all of your customers are using smartphones and you want to keep using your print ticket units our solution is integratable with those.

All the user need to do is to take a picture of the ticket and follow the app.

Stay tuned

Be the first one to know when the app is ready

Email address

[Like on Facebook](#) [Share on Twitter](#) [Share on Google+](#)

We love the press

Sharing is caring, please feel very much welcomed to feature us

[Download press kit](#)

QER
Address 42
FI-00200, Helsinki, Finland
Y-tunnus 2646674-9

Merchants and media: hallo@qer.fi
Support: support@qer.fi
General inquiries: info@qer.fi

© Qer Oy 2015, All rights reserved.