Global Journals LaTeX JournalKaleidoscope™
Artificial Intelligence formulated this projection for compatibility purposes from the original article published at Global Journals. However, this technology is currently in beta. *Therefore, kindly ignore odd layouts, missed formulae, text, tables, or figures.*

CrossRef DOI of original article: 10.34257/GJCSTEVOL20IS4PG1

Investigating Simultaneous Wireless Connections for a Quiz Management System-A Case Study

Michael Hosein

Received: 15 December 2019 Accepted: 4 January 2020 Published: 15 January 2020

Abstract

Near Field Communication is a set of communication protocols for communication between two electronic devices over a distance of 4cm or less and Bluetooth is a wireless technology standard used for exchanging data between fixed and mobile devices over short distances using UHF radio waves in the industrial, scientific and medical radio bands, from 2.402 GHz to 2.480GHz, and building personal area networks (PANs). Both these protocols facilitate wireless/ internet less communication between devices that have the capabilities. The paper titled ?Establishing Simultaneous Server-Side Connections for NFC/Bluetooth enabled Quiz Management Systems? further expanded the concept and usage of these protocols via the examination of modifications made to a Quiz Management System. It highlighted the QMS which was further developed to address a key limitation that was observed in the prior system.

Index terms— quiz management system, smart classroom, protocol, bluetooth, NFC, UHF.

1 Introduction

he developed Quiz Management System (QMS) of this thesis can be seen as an iterative step from the former QMS model known as BlueQ, which was a Bluetooth Based Quizzing application. The quizzing application was able to allow communication between Server and Client devices via the Bluetooth protocol to allow the transferring of data from one device to another. The generalized functionality of the Client(student) device being, the submission of completed quiz material for marking and the receipt of same from the server device. Moreover, the functionality of the Server(teacher) device was primarily for distributing quiz material to students, receiving quiz material from students, performing data analysis on the information received and subsequent distribution of marked material.

The paper "Establishing Simultaneous Server-Side Connections for NFC/Bluetooth enabled Quiz Management Systems", elaborated on the additional functionality of the implemented QMS system of this thesis. It further shed light on the 2 approaches adopted by the QMS to facilitate more simultaneous server-side connections from a high-level perspective. This paper is aimed specifically at providing the technical analysis and testing details that the improved QMS would have entailed. A breakdown of results from each testing iteration would be discussed whereby the most efficient approach would be deduced based on empirical evidence.

2 II.

3 Literature Review

This Chapter would have already been sufficiently covered in the paper "Establishing Simultaneous Server-Side Connections for NFC/Bluetooth enabled Quiz Management Systems". As this paper's main objective is to compliment the work of the aforementioned conference paper, the Literature Review would Cover all approaches undertaken in the Thesis QMS Solution. Details of this can be viewed in the below section.

4 III.

5 Application Details

There were 3 main approaches employed in the proposed solution to solve the initially identified problems. They are detailed below:

6 Approach 1-Version 2 (multi-channel -Identical UUIDs)

This approach explored the route of adding additional RFCOMM channels that are associated with the same UUID. It mimicked the concepts of Client/Server TCP/UDP communications (Lei Wang et al.,2000)having the server socket wait for connection from a client socket, whilst a listening socket is activated for receiving new connections and mapping unto the server socket. In this concept, if we were to assign maximum of2 channels per UUID, we would accomplish simultaneous connections for data transfer to 2 x 8 client devices at a time. The below Sequence Diagram Shows the flow of events in Approach 1-Version 2: By merit of how RFCOMM works it is stated that it would support a total of 60 simultaneous connections based on the UUIDs assigned to each of those channels. This UUID is what the client uses to firstly identify the channel it wishes to connect to via its SDP call, then a connection is established, but since RFCOMM is a serial communication protocol, it would only allow 1 connection at a time per channel, there is no multiplexing unless you switch from a serial protocol to a parallel protocol. Going further into this, the proposed modification could see the sharing of UUIDs for 2 channels at a time, 2 channels being the upper limit. Given that all the necessary provisions are made available, and the user has the optimum storage and processing power to host 60simultaneous connections, this method of dual channeling per UUID could possibly push the amount of simultaneous connections to ”120”total. Now this is of course the perceived amount of simultaneous connections, however it would still be recorded as 60, since half the figure represents Unique (UUIDs) and the other a replication.

But that is observing from a holistic point of view in accordance with the protocol's specifications. As we apply this logic to the presented solution, we can now seek to address the inefficiency of having limited amount of channels used at any given time for material to be distributed by the server device or submitted by the client device. Consider the current application's limit of 8 channels per server device. That alludes to the point that there are 8 UUIDs that are hardcoded into the lowlevel code of the server device by which the RFCOMM channels can be accessed by. The solution being proposed now assigns a total of 2 channels per UUID. This therefore raises the amount of allowable connections to the server by +8 giving a total of 16 connections at a time per server device. The below figure shows the results of sharing UUIDs as more than the maximum limit of 8 devices were able to connect. If in its strictest sense the server device must maintain 8 channels irrespective of UUIDs. Then a queue for each channel with an allocation of 1 allotment for a client device would be established, since the UUID would refer to the channel number in which the service is being provided, the queue would be able to guarantee that a connection has already been made with the channel, however it is currently in use and thus, once completed it will become available.

This guarantee stands on the basis that based on the UUID both the client devices would be treated as one device attempting to access the same service. However, it would all come down to which device connects first. Connecting to the server first would be based on which of the 2 devices is nearer, with the device closest, being successful in connecting to the channel and being able to access the channel and the device furthest from the server being successful in connecting to the server but placed into queue. The below figure shows the results of queuing clients with the samw UUID. This functionality ensures that the client will receive the quiz content without having to re-attempt connecting to the server multiple times. Thus, this method is a viable option for improving the efficiency of Approach 1.

7 Approach 2 -Delegate Function()

Approach 2 -Given that the server device is mainly responsible for the distribution of quizzing materials and can only supply material to up to 8 devices at a time. The proposed solution would see the modification of the existing system to include the 'delegation' functionality. The delegation functionality works as such: Server Device connects to 8/100 devices at a time, given that the assumed sample/classroom size is 100 students. Then in an effort to increase the efficiency of the system, theServer device can select one of the connected client devices and elevate their privileges or rather give/delegate new functionality to allow the distribution of quiz material. Assuming that the selected client device is able to support the same amount of connections as the server device, this therefore means that a total of 16/100 devices can connect to receive quiz material at a time. If the number of client devices 'n' that is given Server privileges increases, then the rate in which quiz material can be distributed would take on the form of an exponential curve, thereby increasing the efficiency of the Bluetooth quizzing system. This approach when compared to the first approach has some immediate advantages: It does not affect the performance of the original hosting device but rather acts as an extension of its functions. What must be taken into consideration is the UUID generator function in which random UUIDs are generated and are hardcoded as the UUID's assigned to the available channels for clients to access their services. This method must ensure that the function delegate() is triggered upon the original Server Device selection of the option to Delegate. As it is an extension of Blue q the interface would consist of the below options. In theory this additional functionality

10.34257/GJCSTEVOL20IS4PG1

can be likened unto a wi-fi peer to peer network as was postulated by developer.android.com, October2020. It is important to recall that the core concept of a peer-topeer network is to partition tasks or workloads between pairs. In this concept peers are equally privileged, similar to the work done by (Sewook Jung et al., 2007). However, in this modification of the system, the server device, determines the functions to send to the client device and thus plays an integral role in determining the amount of privileges a selected client device is afforded. The below Sequence Diagram Shows the flow of events in Approach 1-Version 2:

8 Modeling and Testing

In this section the model or methodology for determining the efficiency of each system as well as which one performs the best comes about upon analyzing the test results for each Approach and comparing the top performing approach's performance, with that of the former BlueQ system.

9 Results Summary:

Based on all previous tests that were conducted, Approach2 performed with the highest level of efficiency and scalability when compared to the previously developed systems and the other approaches tested. The overall performance can be viewed in the below figure13.

10 Conclusion

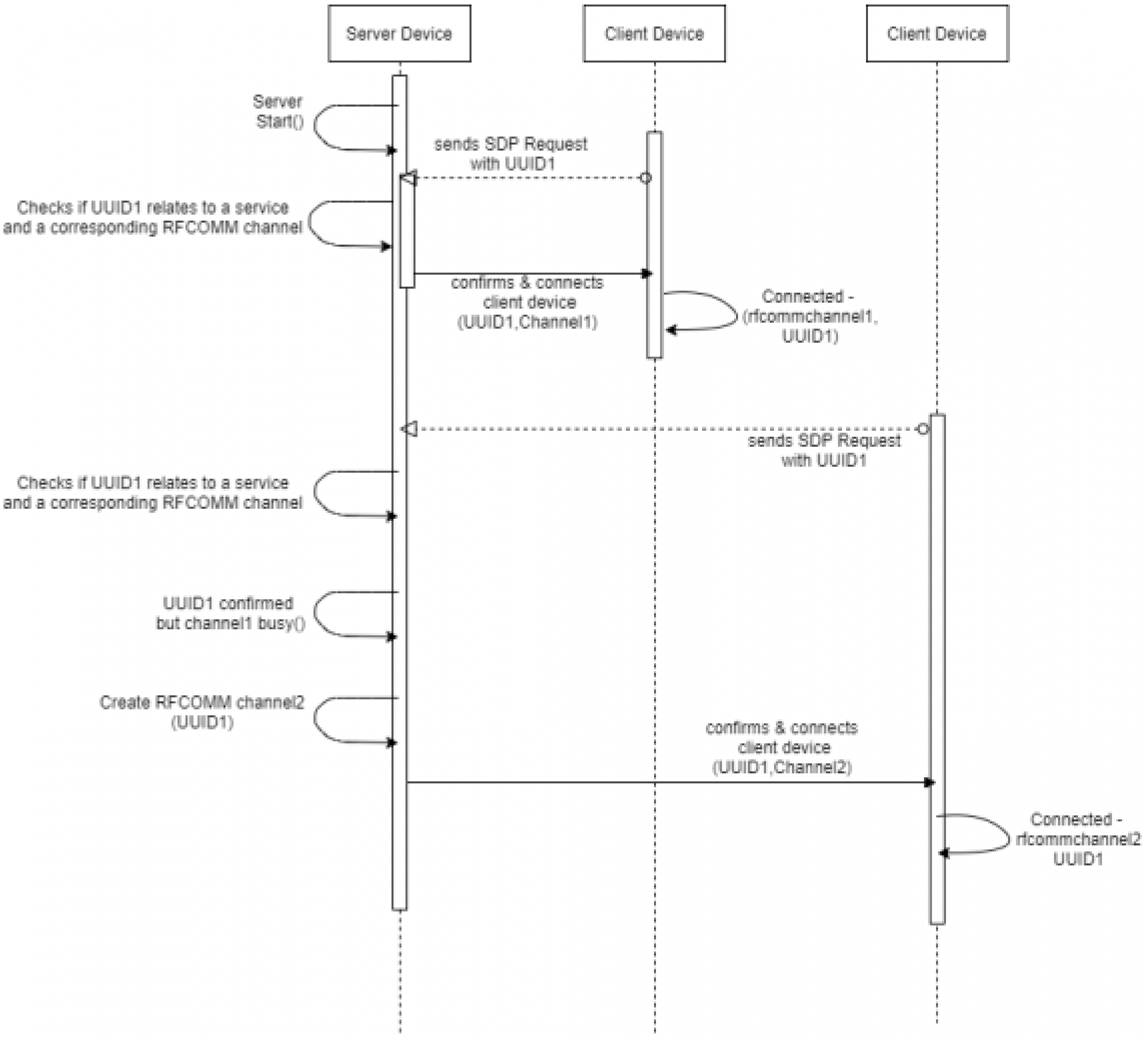
In Conclusion, three solutions/approaches were developed and implemented to solve the problem of lack of popularity or usage of Bluetooth/NFC Quizzing systems within the context of a University Ambient.

These three solutions, saw the usage of the concepts of sharing UUIDs to achieve a greater level of connectivity to the server device as well as the concept of delegating responsibilities from the server device to the client device in an effort to distribute channel sharing load, thus making even more RFCOMM channels available and elevating the privileges of selected clients.

Extensive Testing was carried out to analyze the performance of each approach. Additionally, subsequent testing was conducted on the most efficient solution method against the previous BlueQ model to obtain essential comparative analysis data on the advantages and shortcomings of both systems. [1] [2]

[1] Year 2020 ( ) E Investigating Simultaneous Wireless Connections for a Quiz Management System-A Case Study

[2]



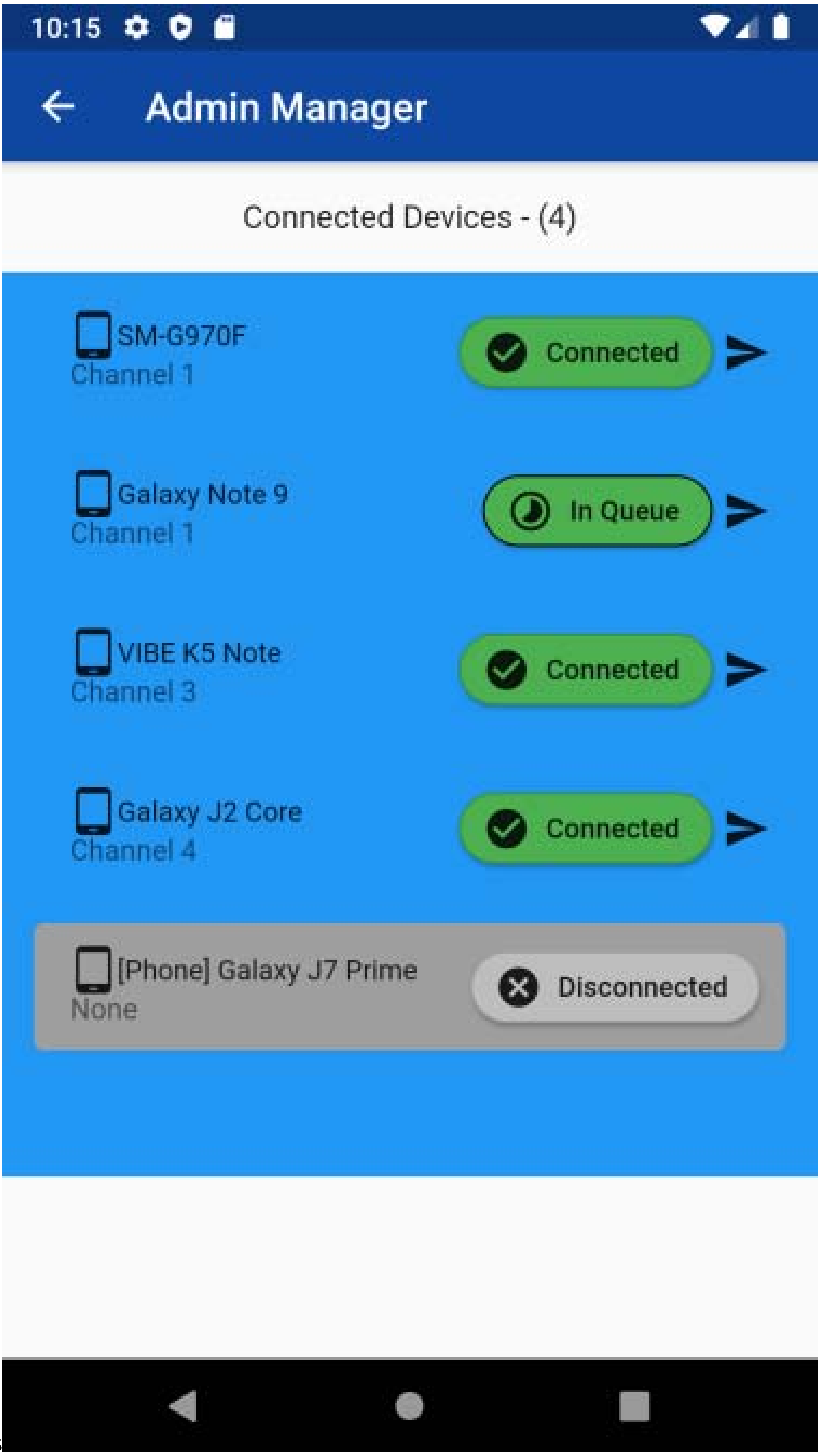
1

Figure 1: Figure 1 :

 10.34257/GJCSTEVOL20IS4PG1

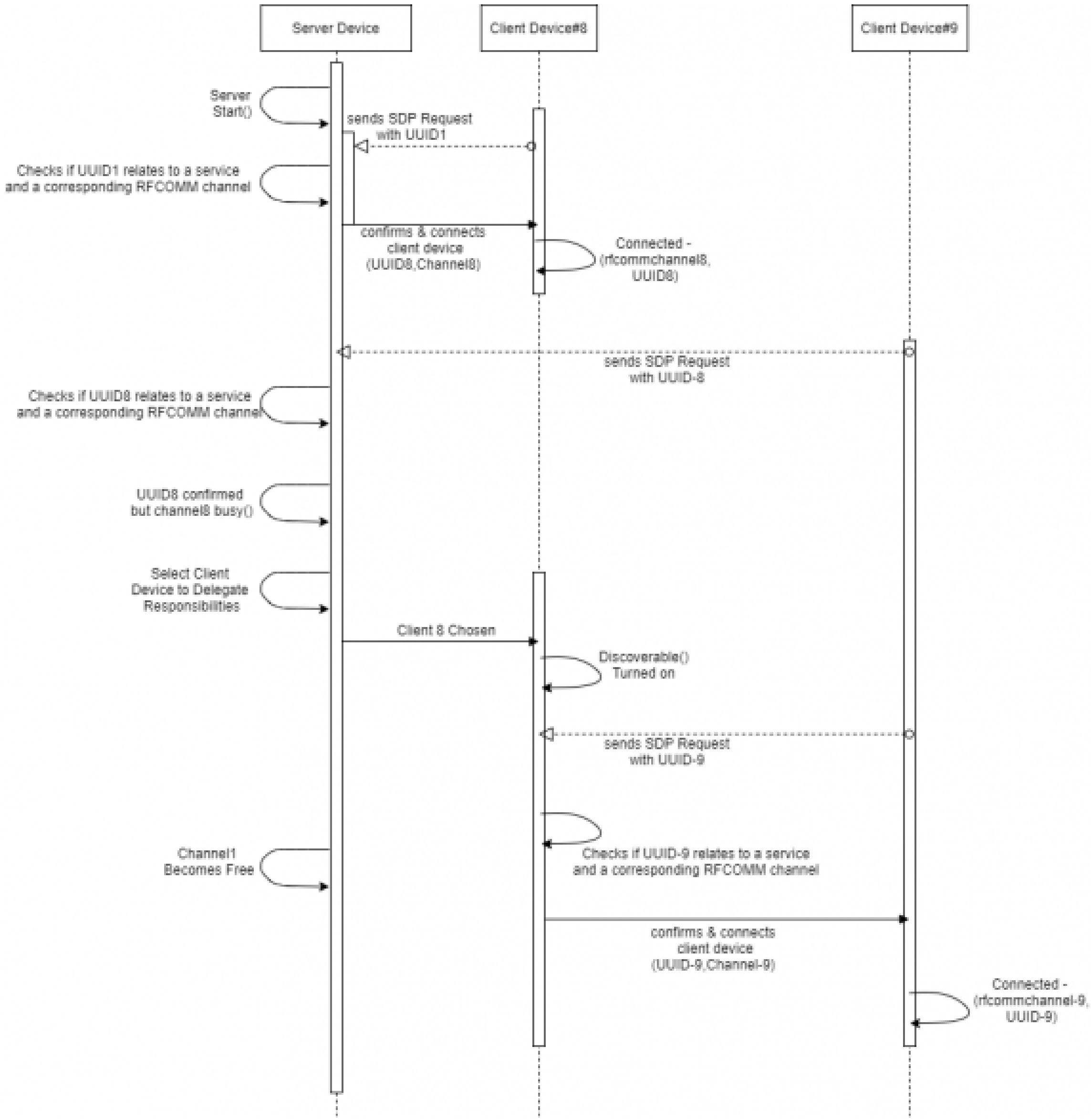
2

Figure 2: Figure 2 :



3

Figure 3: Figure 3 :



4

Figure 4: Figure 4 :

Approaches	Total Amount devices that can be supported in an instance	Average Device Connect Time/s	Total Connection Times/s	Total Amount devices that can be supported
Approach 1 Version 2 - (Shared UUIDs and Multi RFCOMM	16	2.403	38.45	16
Approach 1 Version 3 - (Shared UUIDs and RFCOMM channel	16	2.674	111.78	16
Approach 2 - Delegate	64	2.329	20	64
Previous System - Blue Q	5	2.38	37.3	5

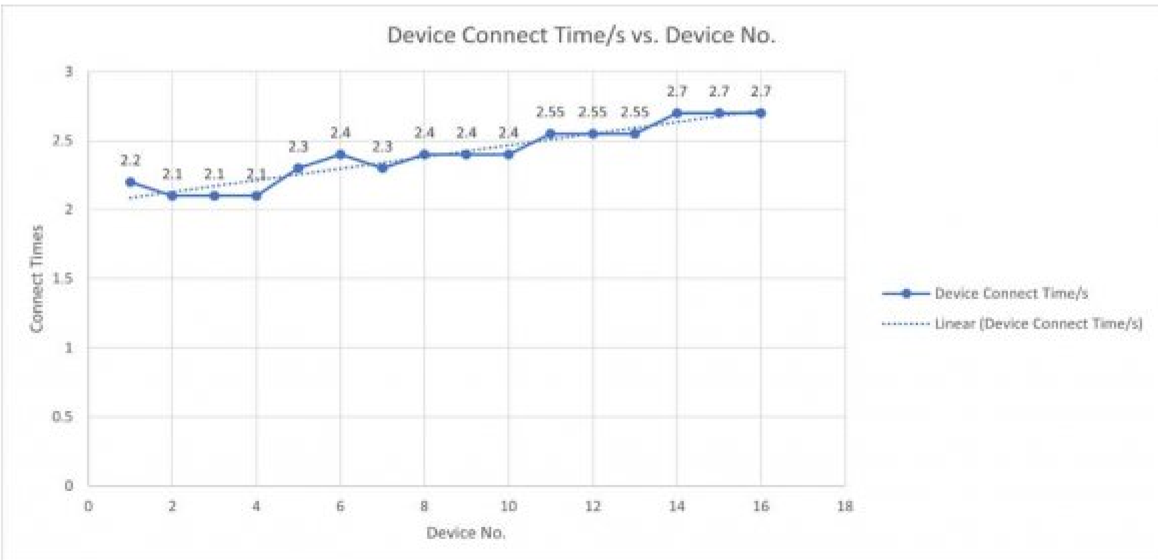
Figure 5: Investigating

UUID No.	Channel No.	Device No.	Device Connect Time/s	Disconnects
1	1	1	2.2	0
1	2	2	2.1	0
2	3	3	2.1	0
2	4	4	2.1	1
3	5	5	2.3	0
3	6	6	2.4	1
4	7	7	2.3	0
4	8	8	2.4	0
5	9	9	2.4	2
5	10	10	2.4	0
6	11	11	2.55	0
6	12	12	2.55	0
7	13	13	2.55	0
7	14	14	2.7	0
8	15	15	2.7	0
8	16	16	2.7	0

Total Connection Time/s	Average Device Connect Time/s	Average Time in Queue/s	Total Disconnects	Rate of Change
38.45	2.403125	N/A	4	/+ 0.15 per 3 devices

5 Rate of Change = +0.15 per 3 devices

Figure 6: Figure 5 :



6

Figure 7: Figure 6 :

UUID No.	Channel No.	Device No.	Device Connect Time/s	Disconnects	Time in Queue/s
1	1	1	1.9	0	
1	1	2	2.1	0	8
2	2	3	2.2	0	9.7
2	2	4	2.3	1	
3	3	5	2.2	0	
3	3	6	6.54	0	8.3
4	4	7	2.3	0	
4	4	8	2.4	0	
5	5	9	2.5	0	
5	5	10	2.5	0	8.6
6	6	11	2.5	0	
6	6	12	2.62	0	8.6
7	7	13	2.62	0	
7	7	14	2.62	0	8.6
8	8	15	2.74	0	
8	8	16	2.74	0	8.6

Total Connection + Queue Time/s	Average Device Connect Time/s	Average Time in Queue/s	Total Disconnects	Rate of Change
111.78	2.67375	8.628571429	1	/+ 0.12 per 3 devices

7

Figure 8: Figure 7 :

8

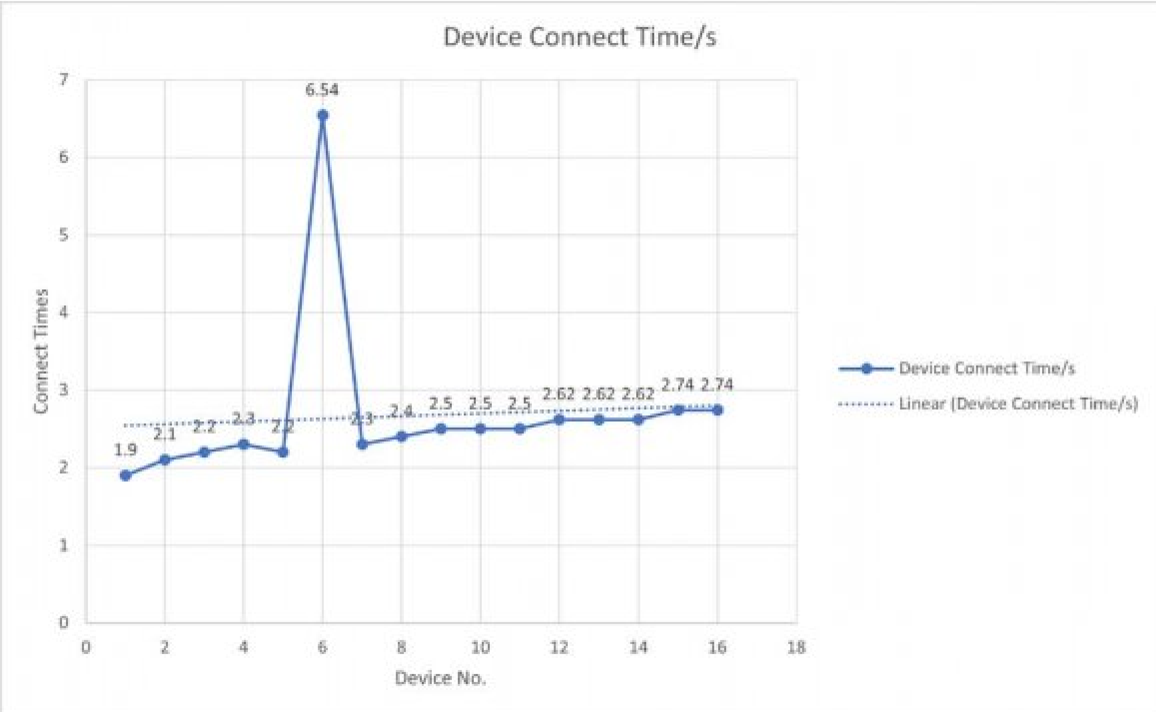
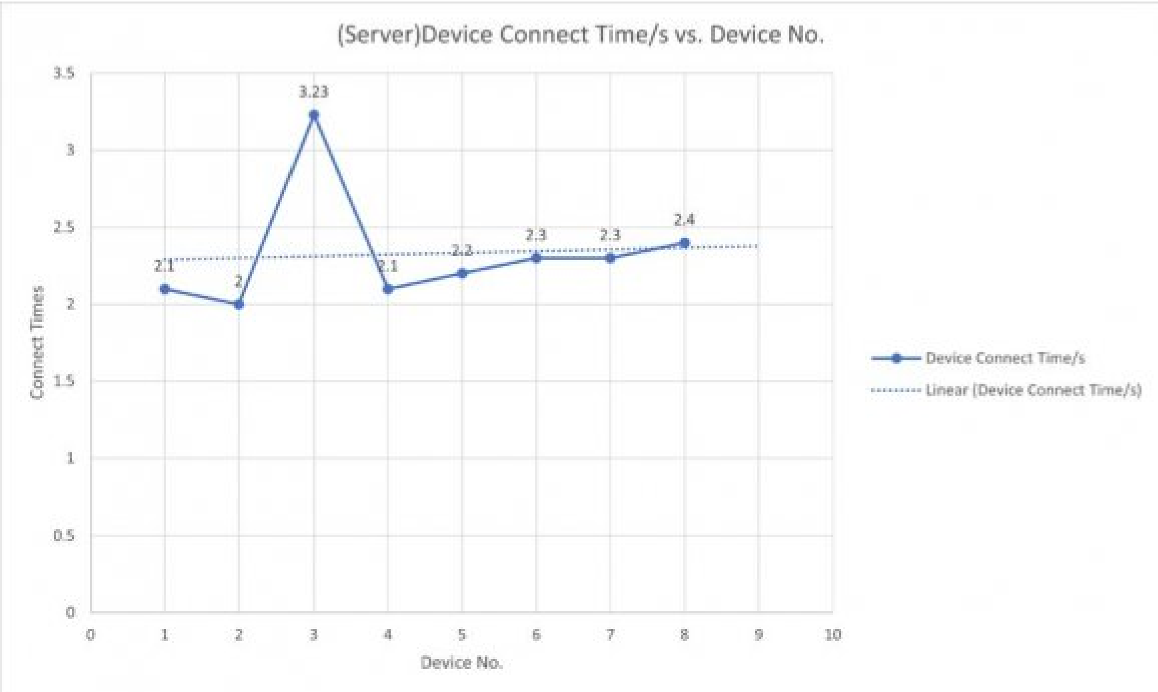


Figure 9: Figure 8 :

9

Server Device Results						Total Connection Times/s	Average Device Connect Time/s	Average Time in Queue/s	Total Disconnects
UUID No.	Channel No.	Device No.	Device Connect Time/s	Disconnects		18.63	2.32875	N/A	2
1	1	1	2.1	0					
1	2	2	2	0					
2	3	3	3.23	0					
2	4	4	2.1	0					
3	5	5	2.2	1					
3	6	6	2.3	1					
4	7	7	2.3	0					
4	8	8	2.4	0					
not Available	not Available	9							

Figure 10: Figure 9 :



10

Figure 11: Figure 10 :

Delegated Client Device 1 Results				
UUID No.	Channel No.	Device No.	Device Connect Time/s	Disconnects
1	1	2	2.1	0
1	2	3	1.9	0
2	3	4	2.1	0
2	4	5	2.1	0
3	5	6	2.3	2
3	6	7	2.2	1
4	7	8	2.3	0
4	8	9	2.4	0

Total Connection Times/s	Average Device Connect Time/s	Average Time in Queue/s	Total Disconnects
17.4	2.175	N/A	3

Figure 12:

Additional testing was conducted with Approach2 and the Previous System to further analyze performance, whereby the below results show the substantial advantage the implemented Approach2 had over the previous system known as BlueQ.

.1 Figure 11: Approach2 VS. BlueQ -Test Results

A series of four (4) experiments were conducted on both the BlueQ system and Approach2 of BlueQ2. Whereby, each experiment saw an increase in the amount of client devices requiring connection to the server device. As shown in the table above, Experiment 4 highlighted the considerable improvement Approach2 provided, as it did not require multiple cycles to facilitate the client devices but rather, only needed to delegate responsibilities to already connected client devices which an upper limit of 8 was given, this meant that a total of 8 x 8 client devices could have been simultaneously connected, once registered with the system.

The shortcoming of the previous BlueQ system was that it only allowed a total of 5 devices at a time to connect simultaneously, but for the purpose of the comparative analysis its upper limit of 8 devices were

[Jung ()] , Sewook Jung . *Pervasive and Mobile Computing* 2007. 3 p. .

[Michael and Laura ()] 'An Educational Bluetooth Quizzing Application in Android'. Hosein Michael , Bigram Laura . IJWMN) 5.6. *International Journal of Wireless Mobile Networks* 2013.

[Mcdermott-Wells ()] 'Bluetooth scatter net models'. P Mcdermott-Wells . *IEEE Potentials23* 2005. 5 p. .

[Developer (ed.) (2020)] *Create P2P connections with Wi-FiDirect*, Developer (ed.) October 2020.

[Aneesh et al. (2012)] *Design and implementation of Bluetooth MAC core withRFCOMM on FPGA*, R Aneesh , Sreekumari , Jiju . Dec 2012.

[Davidrajuh (2009)] 'Evaluating performance of a Bluetooth based classroom tool'. Reggie Davidrajuh . *International Journal of Mobile Learning and Organization* January 2009.

[Grandon et al. (2005)] 'Factors Influencing Student Intention to Adopt Online Classes: A Cross-Cultural Study'. Elizabeth E Grandon , Obyung Khaled Alshare , Kwun . *J. Comput. Sci. Coll* 1937-4771. Apr. 2005. 20 p. .

[Hongwei ()] Du Hongwei . *NFC Technology: Today and Tomorrow*, 2013. (and Communication 2.4.)

[Graham (ed.) (2012)] *Methods and Devices for automatic multiple paring of Bluetooth devices*, Adams Mark GrahamJr . Zimbric Frederick J. (Gurnee, IL US) Walsh Michael G. (ed.) Oct 2012. Johnsburg IL US; Hoschton GA US.

[Mohammed et al. ()] Salah Mohammed , Ismail Abood , Rosdiadee Mahamod , Nordin . *A Quiz Management System Based on P2P Near-Field Communication On Android Platform for Smart Class Environments*, 2016.

[Wang ()] 'Multipath source routing in wireless ad hoc networks'. Lei Wang . *Conference Proceedings. Navigating to a New Era (Cat. No.00TH8492)*, 2000. 2000. 1 p. .

[Naser Hossein, Motlagh (ed.) (2012)] *Near Field Communication (NFC) ATechnical Overview*, Naser Hossein, Motlagh (ed.) May 2012. University of Vasa Faculty of Technology Telecommunication Engeineering -Research Gate

[Ajay (2019)] *NFC vs. Bluetooth: A Detailed Comparison*, Goyal Ajay . May 2019. (In: endivesoftware.com)

[Kulkarni (2013)] 'Peer and Self Assessment in Massive Online Classes'. Chinmay Kulkarni . *ACM Trans. Comput.-Hum. Interact* 1073- 0516. Dec. 2013. 20 (6) .

[Miklos ()] 'Performance aspects of Bluetooth scatter net formation'. G Miklos . *First Annual Workshop on Mobile and Ad Hoc Networking and Computing. Mobi HOC (Cat. No.00EX444)*, 2000. 2000. p. .

[Frydenberg ()] 'Persistence in University Continuing Education Online Classes'. Jia Frydenberg . *The International Review of Research in Open and Distributed Learning* 2007. 8 (3) .

[Ratna ()] *Phones compatible with NFC in*, Sneh Ratna , Choudhary . March 2020. 2019.

[Brian et al. ()] 'Sending and Receiving Data via Bluetooth with an Android Device'. Wirsing ; Brian , Bär , Guido Henning , Rößling . *Erik Tews, and Elmar Lecher*, 2014. 2006.

[Robert (2019)] *What is NFC and how does it work*, Triggs Robert . androidauthority.com. June 2019.

[Ken and Jeff ()] *What is Scrum?*, Schwaber Ken , Sutherland Jeff . 2013.

[Shalini (2018)] *What's a Smart Classroom and Why DoYou Need It*, Khemka Shalini . December 2018.