IOT Multitasking: Design of Smart phone application for systematic execution and scheduling in real time environment

Shaik Asif Hussain
Assistant Professor,
Department of Electronics (E.C.E),
Middle East College, Muscat
Sultanate of Oman.
E- mail: sah.ssk@gmail.com

J Chinna Babu
Assistant professor
Department of Electronics (E.C.E),
A.I.T.S., Rajampet
Andhra Pradesh, India
E- mail: jchinnababu@gmail.com

Chandra Shekar Ramaiah,
Head Strategic projects office
Middle East College, Muscat
Sultanate of Oman.
E-mail: chandrashekar@mec.edu.om

M. N. Giri Prasad
Professor
Department of Electronics (E.C.E),
J.N.T.U.A, Anantapur
Andhra Pradesh, India
E-mail: mahendragiri1960@gmail.com

Abstract—***Scheduling plays a vital role in real time embedded systems to prepare a schedule dynamically and share resources effectively. The scheduling algorithms involve priorities in mixed fashion depending on hard and soft deadlines. Real time scheduling systems basically depends on logical computation and time of the results produced. Multitasking involves parallel execution of request and response based on the scheduling algorithm. It brings automation in many areas with reference to performance parameters like reliability and sensitivity. The major use of scheduling is Effective utilization of resources and avoiding task conflicts. In the proposed work the concept of Scheduling in real time systems is applied in transportation system application for efficient service and effective utilization of resources. One of the best transportation system in the world is railways, however in present days this transport service is under pressure due to mismanagement of operations resulting in customer complaints and long waiting list of passengers. The proposed system is used to accommodate the passengers based on the request and resource availability.***

In this work Internet of Things (IoT) based smart phone application system is designed to introduce automation in addressing the emergency service to the passenger requests in the moving train. The proposed system consists of raspberry pi as a server for maintaining the database and provides wireless communication between the passenger and the ticket collector of train. The aim of this work is to help passengers to have safe and comfortable journey in the train. The user interface designed in each compartment is used to send the request from the passenger to the train employee for allocating the seat in the train in emergency situations. On the request received from the passenger the response is given depending on the resource list loaded in the server. The fair distribution of the seat to the passengers is scheduled using composite scheduling for multitask (CSM) algorithm based on database resource list. The algorithm defines the task priority as its own unique priority level for every task and assign the same priority to the multiple tasks if more number of tasks are introduced in scheduling then they are either time sliced or work round robin fashion. The results obtained in the proposed work justify the performance parameters in real time for the algorithm used.

Keywords—Internet of Things (IOT); Scheduling; Task conflicts; Railways; Smart phone.

I. INTRODUCTION

The transportation is one of the most common need in human being's life. Among various transportation systems railways are widely used as mass transport medium and the best mode of transportation service [1]. During peak seasons most of the passengers find their tickets waiting and are not confirmed till the start date of journey. As it is already known that the passengers availing train service are increasing day by day due to its convenience. The passengers who obtain waiting list ticket have uncertainty in their journey. So in order to avoid this conflicts scheduling approach is considered to address the passenger needs. The scheduling approach in real time systems provide a systematic execution and response as per the requirement of passenger service. The work is proposed with multitasking resource algorithm for real time embedded system applications which is suitable for both real and non-real time embedded systems application. The real time embedded system and scheduling approach of Real time operating system (RTOS) would overcome all the issues generally occurring railways system [2], [3]. Usually in every country the procedure of transportation in railways is common. Normally, when the train is in stationary at the railway station, the Passengers can make a phone call to rail administrator number given on the ticket for their problems and to get required resources. The train crew person is one who actually provides the support to the required resources of the passengers. As per rail administrator instructions crew person's act totally as required. The rail administrator has to ensure about all crew status, i.e. whether any crew person is free or engaged. If anyone is free then, the rail administrator will assign tasks else other crew is checked [6] to assign the task requested by the passenger. Therefore, the existing servicing style is time consuming process. The existing system needs information about the crew person's location to assign the tasks accordingly. In order to know his status as well his location the admin has to establish the wireless connection to him (crew person) with a walkie-talkie device [10]. This hand

978-1-5386-8046-9/19/$31.00 ©2019 IEEE

held devices are operated with some specified range of frequencies and sometimes due to miscommunication between rail admin and crew member or vice-versa because of the poor coverage signals. Hence RTOS multitasking system can provide more convenience to the passengers. The novel proposed method is more suitable for both real and non-real time embedded systems application. This embedded system is designed to achieve all problems that are seen in railways [13].

II. Problem Statement

Real time Scheduling policies play a vital part in assigning multiple tasks based on the priorities. The problems associated with transportation of railways devise real time scheduling algorithms for assigning tasks to the train crew by set of priority assignment rules to finish the task in deadlines [19]. However the requirements and resources of Multitasking algorithms carry out few assumptions such as

- Preemptive and non-preemptive resource scheduling
- Execution of tasks with period and non-periodic intervals
- Completion of the task within given deadline based on run ability constraints
- Use of scheduling policies first come first serve, round robin, shortest job first, Priority based, Earliest deadline first, composite resource schedule

In practice the assumptions from the various scheduling theories do not really hold. The rail administrator assigns the task to the train crew and the task is strictly followed with a deadline based on first come and first serve. Certainly various parameters are affected in completing multiple tasks at the same time and the results may not be consistent as it effects the integrity of the task [14]. If any high priority task is assigned the lowest priority is preempted and high priority is performed this may cause timing analysis and sense of response time in completion of the task. This results in priority inversion, dead lock and pitfalls related to time triggering.

Composite schedule resource algorithm is used in IOT system as it is more effective in distributing the resources and different processes in real time efficiently. However the algorithm produce the list of requests collectively on database in sequential steps as follows:

1. The designed IoT project has a single raspberry pi 3 model B as master, and the real time results are carried out between user and crew persons depending on mobile platforms.
2. The Bluetooth technology is used as in-built facility in raspberry pi and this feature is utilized to make a pair with crew person for enabling that raspberry pi to transfer the crew person Id to server. Sometimes the passengers find difficulty in pairing with raspberry pi because the crew person location cannot be updated unless, the raspberry pi must paired with crew persons due to Bluetooth beacons. The limitation of beacons is avoided with the concept of sequential circuit. The solution is to update the crew person's location automatically based on recent received segmented list from the server.
3. The real time results are given through the smart phones and web links provided for both, the passenger and crew person. So that, the passenger can easily get the requirements in moving train.
4. The comparative results between the existing scheduling algorithm and CSM algorithm are based on the parameters called average waiting time, average response time and this calculation is done by a normal mean calculation method.
5. The priority requests for different services and resources in railways can be served as given below

Table 1: Priority list request

S. No.	Request list	Priority
1.	Waiting list ticket	000
2.	Emergency ticket	001
3.	Health emergency	010
4.	Fire Alarms	011
5.	Security or loss of belongings	100
6.	Water issues	101
7.	Food services	110
8.	Any other related	111

Another issue with rail transport is during peak seasons the tickets issued to the passengers are in waiting list which usually bring more problems to the passengers and also the crew members of rail transport [15]. It is a clear fact that every country face this problems during religious festivals. Throughout the world every year the problem has continuously increased from 5 to 20%. Increasing number of waiting list passengers, increase rushes in train which results "happy journey" slogan of Indian Railway in to "unhappy journey"[16].

III. System Design

The main objective of the proposed method is to develop a scheduling algorithm to handle multitasking based on priorities listed. The proposed algorithm use Composite scheduling for multitask (CSM) algorithm. In this method, an Internet of Things (IoT) concept is used to provide the resources such as catering, cleaning, security and first aid etc.

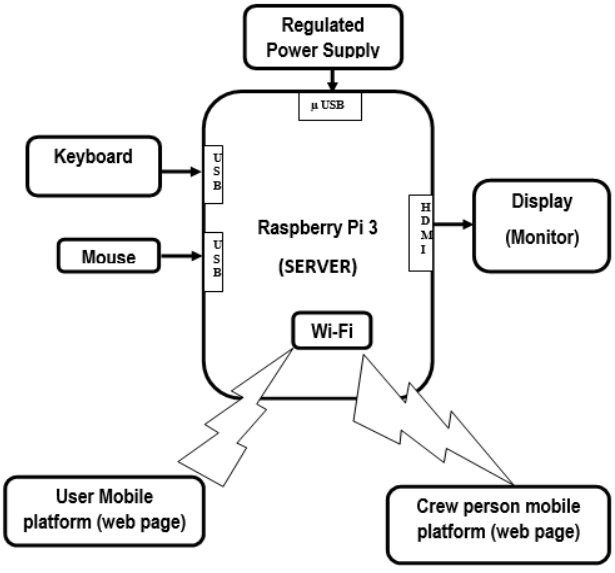

Figure 1: Proposed Block diagram

3.1. Server

A server is a computer program or a device that provides different functionality or all information for other devices like "client" and the architecture is used is client–server model. A Server can provide various functionalities to the clients often called "services", such as data sharing or provide multiple resources to clients, or performing computation for a client. Here, the raspberry pi is acted as a master/server to maintain the database. The proposed CSM algorithm is developed in it. The server receives the passenger requests and

distributes to the crew person to meet the passengers request with suitable resources in moving train.

The raspberry pi has in-built Wi-Fi feature which can provide the wireless communication in between passenger and server. A separate router is placed along with raspberry pi in each compartment. The user can access the internet facility from router and make requests. The client-server model is explained in below section.

3.2. Hypertext Transfer Protocol (HTTP)

The internet web provides distributed information system of client server model as shown below figure 2

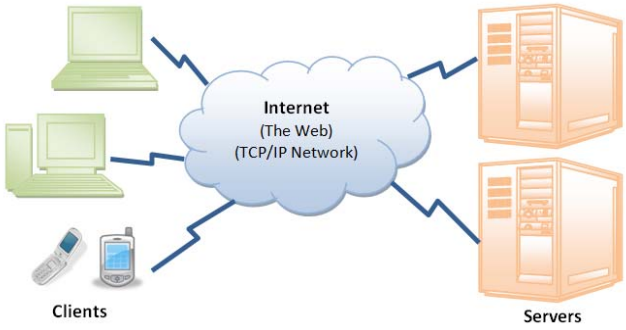

Figure 2: HTTP protocol as client-server model

To have proper communication in between the client and the server, these applications must compatible and agreed on a specific application-level protocols such as HTTP, FTP, SMTP, POP, etc. The HTTP defined to indicate and obtain desired action on the identified resources of server. Normally the user can post their queries to the server through this HTTP protocol this resource represents about the pre-existing (recorded programs or files) data or dynamic data/program. Once the client post gives the request to the server then the server has to give response appropriately.

3.3 Safe Methods

The methods called, HEAD, GET, TRACE and OPTIONS are defined as *safe*, which means they are intended only to retrieve data and these methods should not change the behavior or the present state of the server. Simply, these methods provide effects logging, caching, the serving of banner advertisements or incrementing a web counter to the original data file or program file in the server

3.4. Idempotent methods and web applications:

The methods PUT and DELETE comes under idempotent it means that, the multiple similar requests may have same effect as one single request. In other words, when the identical requests been made then, the response is same for all requests sometimes while the server is on its action, it gives the response as current state or unrelated. The GET, HEAD, OPTIONS and TRACE, are identified as safe. The POST methodology is used in transferring a uniform POST request multiple times. Further to this an effect on state can cause any aspect effects (such as monetary transactions).

3.5. Uniform Resource Locator (URL):

URL stands for Uniform Resource Locator and it acts as a reference address to resources from server/Internet. It has two components:

1. Protocol identifier: This is used to place the requests to the server through http://example.com, here the protocol identifier is hypertext transfer protocol (http).
2. Resource name: This is used to provide the required resources/files/programs to the client.

3.6. Input Devices

The input device is the device which is used to give the instructions to the computer. The keyboard, mouse are used to control the embedded trainer kit (Raspberry pi 3 model B). These two devices are very essential to fetch and store the data from the memory location. It is also used to execute the application software developed in Raspberry pi. These input devices are interfaced with embedded trainer kit through Universal Serial Bus (USB) ports.

3.7. User Mobile Platform

The user mobile platform refers to a smart phone of the Passenger. The web link is provided in each compartment and passengers can utilize it to get the required resources in moving train. Once the user or passenger copied that web link in smart phone then, he or she has to log in with their registered mobile number and name. After that, user operating window will be displayed with menu. The menu consists of multiple resources such as catering, security/accident, cleaning, first aid etc. The user can place the requests in database by selecting the options. Any user smart phone can access the web-link.

3.8. Crew member mobile platform:

The crew person provides the resources to passengers in railways. In the proposed system, a crew member is in control by the server requests. The crew member also has a web link through smart phone which can get the resource list from server. The resource list refers to the list of passenger requests to be served with appropriate resources. Minimum four crew persons are required to attain good productivity in moving train. The crew persons do not obtain the same resource list. Here, a crew person can provide the multiple resources to the passengers.

3.9. Output Device

Generally, the output devices are monitored through LCD display. All the software structure or architecture explicitly depend on the output device display or monitor. The user can see the result on it. A separate connector called VGA (Video Graphic Array) to HDMI (High-definition Multimedia Interface) is used to interface the display with Raspberry pi. The raspberry pi is invented by a Quadcom company with the help of BCM2387 architecture. It is 64 bit quad-core ARM cortex- a53 processor.

3.10. Raspberry pi Microcontroller: This microcontroller has inbuilt Wi-Fi, Bluetooth and HDMI port, 3.5mm audio jack, Micro USB power port, four USB ports, 40 GPIO pins, video core GPU, 1.2 gigahertz clock speed, 1 GB RAM. This pi 3 is very useful to make excellent prototype system for the IoT [9] development board.

1. The Micro USB power port is used to give power to the Raspberry pi. HDMI port is used to connect pi with HDTV or Monitor easily through Blu-ray quality video using H.264 codec at 40MBits/s.
2. It has totally four built-in Universal Serial Bus (USB) ports any two of them are used as input ports which are mouse and keyboard. Remaining two ports can also use as a power hub with 1.2 Amps through which USB devices are externally connected.
3. It has 40 General Purpose Input Output (GPIO) pins, out of which 27 are available for the user as GPIO and a few pins are used for UART, I^2C, SPI communication protocols. The remaining pins are for 3V and 5V voltage sources.
4. The Broadcom BCM2837 system-on-chip (SoC) consists of four high-performance ARM Cortex-A53 processing cores and each one is running at 1.2GHz.

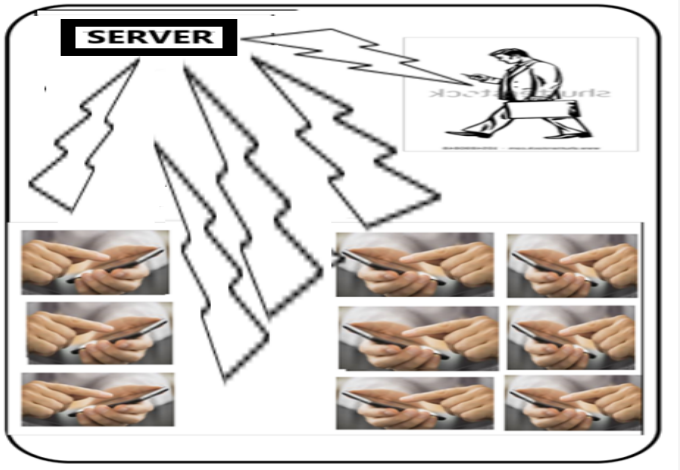

Figure 3: crew member servicing to the passenger with in the train.

3.11. Project Description:

Now a day's, most of the people are relying on train journey as means of comfortable transportation. In future, definitely there will be an increase in crowd which will have impact substantially on passenger's comfort. The passenger residing in the moving train, may not be satisfied with the services provided by a crew because of task mismanagement. A recent economic survey [2] stated that, the productivity in railway network is low. The major cause is either due to negligence of crew person or mismanagement. It also states that, the railways department needs to take responsibility to introduce the automatic resource technique to provide the safety and comfortable journeys to the passengers in the railways. The service quality is poor due to lack of management, dependable communication, and confirmation. These problems have been encountered in the existing system. The Internet of Things (IoT) [1] is a new technology and plays major role in network computations, wherein all devices are interconnected. The process or task is controlled from anywhere through the internet. In its Paradigm, all connected devices have unique identification so that, dependable communication can be possible without conflicts and collisions. In this paper an IoT technology is used to resolve the issues to passengers. Each and every compartment has separate IoT set-up and one among them will act as master. The user can make the requests through the web link. All the requests are gathered in the raspberry pi [6] which forwards the information to the server with the help of built in Wi-Fi module. A router is placed in each coach/compartment. The CSM algorithm creates the optimized resource list and server distributed the resource list to all crew members effectively. The major works of this paper are optimization and distributing of resource list (equal workload) to all crew persons and meet the needy travelers with suitable resource. Minimum Four crew members are needed to implement this project in railways. With the help of an inbuilt Wi-Fi facility of raspberry pi, it becomes very easy to exchange the information from one place to another place. So in order to perform this designed work Raspberry pi 3 model B is used to work as mediator between passenger and the server. The router is also fitted along with the raspberry pi. The passengers are allowed to send the requests to the server via raspberry pi. In response, the server will assigns the task to crew persons based on the arrived requests of the passengers from sleeper/AC coaches. This raspberry pi replaces the need of rail administrator in moving rail. Whenever the user wants to get the resource from a crew person, user can make requests at any time during their journey through the web link provided to him with login credentials as <http://192.168.15.251/RT_HTML/>. The master starts to execute the CSM algorithm and prepare the resource list (new database) after receiving requests. The server has the request list as segments and assigns those segments to all crew persons for the first time no one get the same segment list. The server keeps record of the crew persons location based on the segment list (is the list of requests grouped from the same coach or few consecutive coaches). The server continuously assigns the segment lists to all crew persons based on previous database records. It has the inbuilt Wi-Fi facility so that the server also sends segment list to all crew persons. The crew person also has the web links, through which the crew person will receive the segments continuously and this link is given as http://192.168.15.251/RT_HTML/Admin.php

IV. Composite Schedule for Multitask (CSM) Algorithm

This CSM algorithm execute repeatedly every 100 seconds. The execution steps are followed in sequence. The CSM algorithm is developed with C++ using the GCC Compiler [5].

1) In each and every compartment a raspberry pi is placed and one among them will be acted as master.
2) Each pi model in every compartment will receive randomly produced requests from a passenger's mobile platform.
3) Once the requests are gathered at raspberry pi (server) then, it starts to create a new database (resource list) on static assigned priority list. The new list consists of higher priority tasks are occupying the first few places. Lower priority tasks are occupying last places in resource list queue. If any two requests have same priority then, the priority is given based on their arrival time.
4) The server segments list is prepared as resource list by CSM algorithm and assigns each segment to each crew persons. The location of the crew person is identified based on previous segment records.
5) If new requests are made while algorithm runs, then the master will check the priority of that newly arrived request and if it has higher priority then, immediately assigns that task to him on emergence.
6) If the newly arrived task has low priority then, at that time again all crew members are engaged with a task then, it just put request into waiting state until the appropriate resource becomes free, the server assign that task to that suitable crew person.
7) Once the crew member tasks are finished, then he/she needs to stay there only until the new segment arrives otherwise, he has to arrange all resources from pantry room.

4.1 Algorithm Flow Chart:

The algorithm is depicted in flowchart model. The flow chart refers to a step by step process to represent any algorithm. This flow chart is developed with help of steps that are mentioned in figure 4.

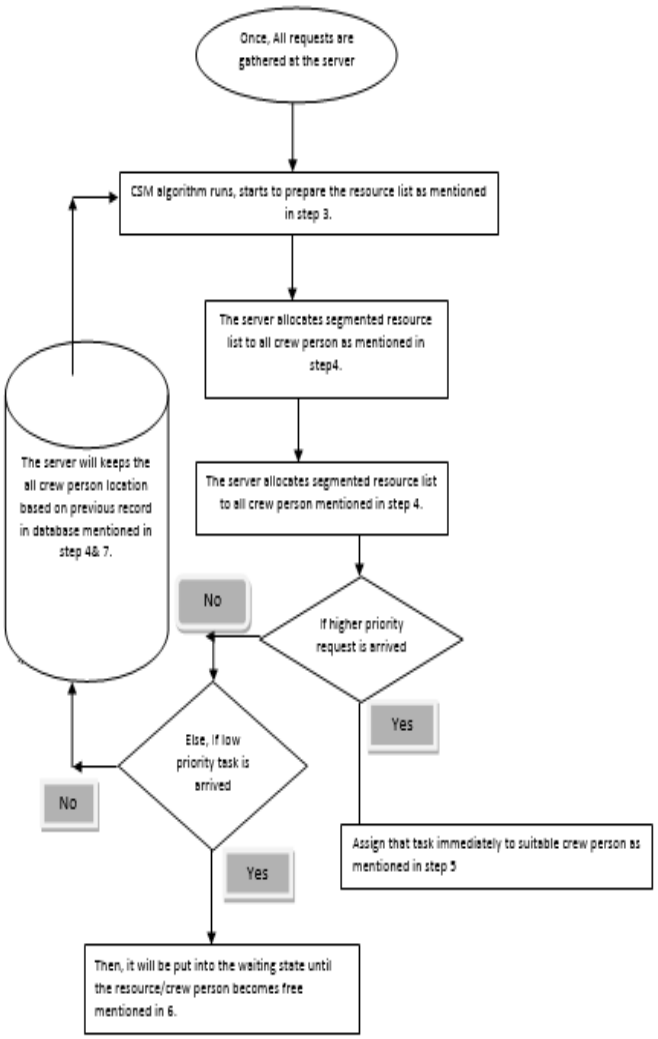

Figure 4: Flow chart of CSD Algorithm

V. Experimental Results

In this section, it is explained the results of research and at the same time is given the comprehensive discussion.

5.1. Passenger mobile operating window:

Passenger mobile operating window web link can be supported by all android based smart phones and also compatible with other platforms too. This window is useful to place the requests from the passengers. This window will be displayed when, the passenger browse the web link provided to him. This web link/IP address is also existed in server such that the server accepts all those requests from the passengers. This window consist of list of resources. This window can also be modified or changed. The resources are listed below:

1. Fire/security.
2. Health emergency
3. Meal/water.
4. Cleaning.

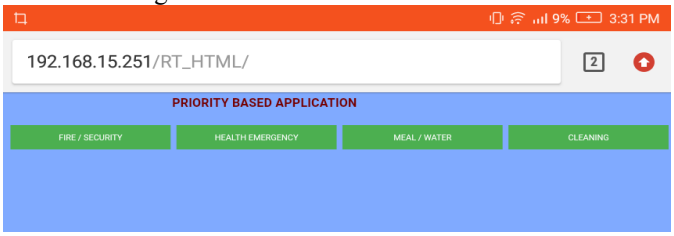

Figure 5: User's operating window.

The user browser has IP address on their mobile which automatically use user operating window for displaying and placing the requests. This window is shown in the above figure 5.

5.2. Crew Persons Resource Window:

The crew person's resource window is displayed on smart phone. It has distinct web link as same as the passenger. The list called segmented resource list sent from the server. The crew person's is directed to perform the request based on segmented list. Crew member do not have privileges to modify or change the current segmented resource list. The location is automatically updated based on the previous data. The crew person's resource window is shown in figure 6 below.

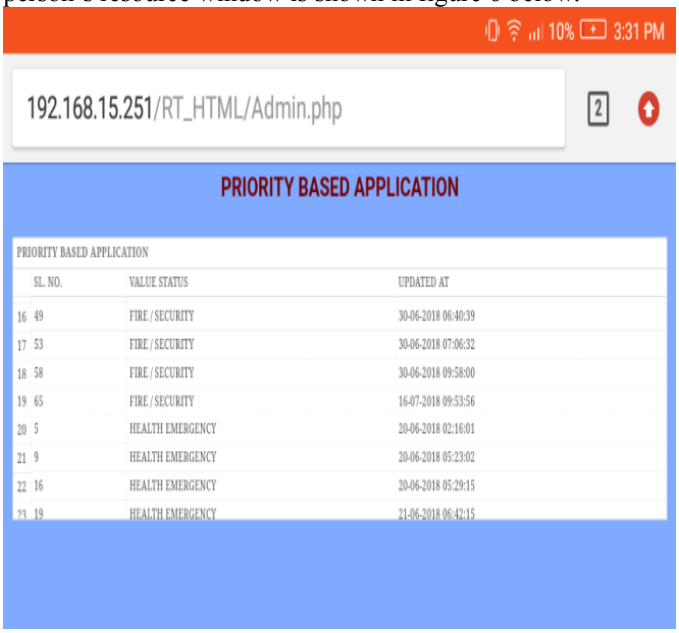

Figure 6: crew person Segment resource window.

VI. Discussion

The discussion is purely based on the comparison between the proposed CSM algorithm and existing scheduling algorithm. The comparison table is developed with the parameters called average waiting time and average response time. Here, the higher priority request should be serviced within less time but it is not mandatory for lower priority request.

Table 2: Real Time passenger Requests Details.

Request Ids	Assigned static priority value	Execution time (msec)	Arrival time At server
Catering(food/water)	3	7	5:30 PM
Health emergency	2	6	5:31 PM
Fire accident	1	5	6:40 PM
Cleaning	4	9	6:50 PM

From table 2, it shows the different requests along with their execution and arrival time. Execution time [9] [10] refers to, number clocks are needed to complete one task/process, and arrival time refers to, the time at which the request is generated. The above collected data is real time data. In generally, the algorithm is opted for real time embedded system to produce lesser delay and provide good response.

Table 3: Relation between $\overline{W(t)}$ and $\overline{r(t)}$

Algorithms	Average Waiting time W(t) (sec)	Average response time r(t) (sec)
FCFS (First come first serve)	9.22	16.00
SJF (shortest job first)	8.22	12.75
RR (Round Robin)	17.55	17.55
PRIORITY based	10.75	17.50
EDF (Earliest deadline first)	9.25	16.25
Proposed CSM (Composite Schedule Multitasking)	8.00	11.45

From table 3, the proposed CSM algorithm required small average waiting time and response time compare to existing algorithms. This above table is developed with the data taken from table 2, these parameters W and r decreases. Whereas, the remaining existing scheduling algorithms cannot be decrease because, the kernel cannot minimize the static data. The graphical representation gives explicit explanation from the table 3. And the graphical data clearly is shown in figure 7.

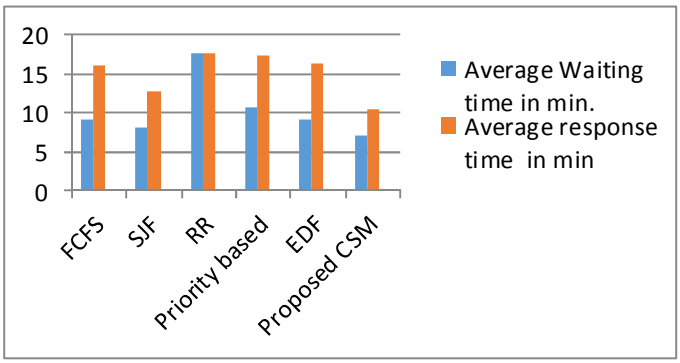

Figure 7: Graphical Representation of Relation of W and r

This result is purely prototype based one, at the end; the proposed algorithm can also be implemented in railways. In the proposed system, the crew persons and CSM algorithm achieves the problems addressed earlier.

VII. Conclusion

An IoT based prototype model is developed to provide the services to every passenger in moving train. This automation concept in railways will bring more customer satisfaction and also improves the profitability to the government in future days. Each and every compartment has separate IoT installation with raspberry pi, one among them acts as master or server. The raspberry pi is responsible to provide wireless communication between passenger and server. The proposed CSM algorithm requires minimal delay to meet the needs of the passenger when compared to existing algorithms. The server has the resource lists which segments the resource list and distributes to all crew persons equally divide the work-load to all crew persons. All passenger requirements are provided with suitable resources without delay in action. Therefore, the people who ensure the journey in train will have safe and comfort travel through the developed embedded application.

Acknowledgment

I am very much thankful to Middle East College, Muscat for providing me lab facilities and also creating an environment of renowned professors who helped me at every step.

References

[1] L. Atzori, A. Iera, and G. Morabito, “The internet of things: A survey,” Computer networks, vol. 54, no. 15, pp. 2787–2805, 2010.

[2] Pritish Sachdeva.A, Shrutik Katchii.B, , “A Review paper on Raspberry pi” Electronics and telecommunication department, Available online 05 Nov 2014, Vol.4, No.6,pp.2347 - 5161 (Dec 2014).

[3] Deepa.T, Dr Dhanaraj Cheelu, “ A comparative study of static and dynamic load balancing algorithm in cloud computing ” ISBN 978-1-5386-1887-5, published in ICECDS, 2017.

[4] Nitu Gangwar, tushar senwal, Nair.B, “CARE: An IoT based system for passengers comfort in railways, ” communication system and networks, 2017., 2155-2509.

[5] R. Khan, S. U. Khan, R. Zaheer, and S. Khan, “Future internet: The internet of things architecture, possible applications and key challenges,” in 10th International Conference on Frontiers of Information Technology (FIT), Dec 2012, pp. 257–260.

[6] M. Chen, S. Gonzalez, and V. Leung, “Applications and design issues for mobile agents in wireless sensor networks,” Wireless Communications, IEEE, vol. 14, no. 6, pp. 20–26, 2007.

[7] D.B. Stewart, R.A. Volpe, and P.K. Khosla, “Design of dynamically reconfigurable real-time software using port-based objects,” IEEE Trans. on Software Engineering, vol. 23, no. 12, pp. 759-776, Dec. 1997.

[8] T.A. Henzinger, B. Horowitz, and C.M. Kirsch, “Embedded control systems development with Giotto,” in Proc. of Languages, Compilers, and Tools for Embedded Systems (LCTES’01), UT, June 2001.

[9] E.A. Lee, “What's ahead for embedded software,” IEEE Computer, vol. 33, no. 9, pp. 18-26, Sept. 2000.

[10] C. Liu and J. Layland, “Scheduling algorithms for multiprogramming in a hard real-time environment,” Journal of the ACM, vol. 10, no. 1, 1973, pp. 46-61.

[11] J.P. Lehoczky, L. Sha, and Y. Ding, “The rate monotonic scheduling algorithm: exact characterization and average case behavior,” in Proc. of IEEE Real-Time Systems Symposium, Dec. 1989, pp. 166-171.

[12] N. C. Audsley, A. Burns, M. Richardson, and A. Wellings, “Hard real-time scheduling: The deadline monotonic approach,” in Proc. of IEEE Workshop on Real-Time Operating Systems and Software, May 1991, pp. 133-137.

[13] K. Tindell, A. Burns, and A. Wellings, “An extendible approach for analyzing fixed priority hard real-time tasks,” Real-Time Systems Journal, vol 6, no. 3, pp. 133-151, 1994.

[14] M. Spuri and G. Buttazzo, “Scheduling aperiodic tasks in dynamic priority systems,” RealTime Systems Journal, vol. 10, no. 2, pp. 179-210, 1998.

[15] D.B. Stewart and P.K. Khosla, “Mechanisms for detecting and handling timing errors,” Communications of the ACM, vol. 40, no. 1, pp 87-94, Jan. 1997.

[16] T. Dean and M. Boddy, “An analysis of time-dependent planning,” in Proc. of the Seventh National Conference on Artificial Intelligence, St. Paul, MN, August 1988, pp. 49–54.

[17] Jie Liu, Responsible Frameworks for Heterogeneous Modeling and Design of Embedded Systems, Ph.D. Dissertation, EECS, University of California, Berkeley, Fall 2001.

[18] D.H. Shim, T.J. Koo, F. Hoffmann, and S.S. Sastry, “A comprehensive study of control design for an autonomous helicopter,” in Proc. of 37th Conference on Decision and Control, Tampa, FL, 1998. pp. 3653-3658.

[19] B. Horowitz, J. Liebman, C. Ma, T.J. Koo, T.A. Henzinger, A. Sangiovanni-Vincentelli, S. Sastry, “Embedded software design and system integration for rotorcraft UAV using platforms,” in Proc. of 2002 IFAC world congress, Bacelona, Spain, July 2002.