UNIVERSIDAD DISTRITAL
FRANCISCO JOSÉ DE CALDAS

Visión Electrónica
Más que un estado sólido
https://doi.org/10.14483/issn.2248-4728

VISIÓN ELECTRONICA
A Research Vision

Implementation of a queue with priority with parameterizable variables to improve QoS

Implementación de una cola de prioridad con variables parametrizables para mejorar la QoS

Luis Adrián Lasso-Cardona[1], Jesús David Mejía-Vergara[2], Jacobo Grajales-Cabrera[3]

INFORMACIÓN DEL ARTICULO

Historia del articulo
Enviado: 03/07/2019
Recibido: 16/08/2019
Aceptado: 20/11/2019

Keywords:
Data structures,
FIFO,
Priority queues,
Quality of service,
Queue theory.

Palabras clave:
Estructuras de datos,
FIFO,
Colas de prioridad,
Calidad del servicio,
Teoría de colas.

ABSTRACT:

This article presents the implementation of a priority queue with simulated parameterized components on the scenario of a banking institution in which four main elements were used: 1) the dispenser, 2) the cashier(s), 3) the server that receives and assigns the priority of the turns, and 4) a screen that displays the order of the turns. In any banking institution, there is always a large number of users that require attention, so they implement special turns to give priority to customers, improving the quality of service. One of the important elements within the cycle of service provision is related to the waiting lines, commonly called queues. Basically one queue is managed with the First-In, First-Out discipline. In this aspect, disciplines such as queuing theory help to understand the behavior of queues by examining all the components of the waiting lists. One of the most used disciplines for the management of a queue, is the enqueue with priority. In a priority queue, the element with the highest priority is removed before any other elements. The research was carried out using mainly a quantitative approach. The main goal of this type of research is the formulation and demonstration of theories and seeks that the studies carried out can be replicated. The algorithm implemented highly parameterizable allowed the correct simulation of a priority queue that can be taken as a basis for a subsequent application in any type of organization.

RESUMEN

Este artículo presenta la implementación de una cola de prioridad con componentes parametrizables simulada sobre el escenario de una institución bancaria en la cual se utilizaron cuatro elementos principales: 1) el dispensador de turnos, 2) el cajero(s) encargado de la atención, 3) el servidor que recibe y asigna la prioridad de los turnos, y 4) una pantalla que visualiza el orden de los turnos. En cualquier institución bancaria, siempre hay un gran número de usuarios que requieren atención, por lo que implementan turnos especiales para dar prioridad a los clientes, mejorando la calidad del servicio. Uno de los elementos importantes dentro del ciclo de provisión de servicios está relacionado con las líneas de espera, comúnmente llamadas colas. Básicamente una cola se gestiona con la disciplina First-In, First-Out. En este aspecto, disciplinas como la teoría de colas ayudan a comprender el comportamiento de las colas al examinar todos los componentes de las listas de espera. Una de las disciplinas más utilizadas para la gestión de una cola, es el encolado con prioridad. En una cola de prioridad, el elemento con la prioridad más alta se elimina antes que cualquier otro elemento. La investigación se realizó utilizando principalmente un enfoque cuantitativo. El objetivo principal de este tipo de investigación es la formulación y demostración de teorías y busca que los estudios realizados puedan replicarse. El algoritmo implementado altamente parametrizable permitió la correcta simulación de una cola de prioridad que puede tomarse como base para una aplicación posterior en cualquier tipo de organización.

1BSc. in Systems engineering, Universidad del Valle, Colombia. MSc. in Educational Technology Management, Universidad de Santander, Colombia. Current position: Unidad Central del Valle del Cauca, Colombia. E-mail: llasso@uceva.edu.co ORCID: https://orcid.org/0000-0002-3354-1554
2BSc. (c) in Systems engineering, Unidad Central del Valle del Cauca, Colombia. Gigae3D Research Group. E-mail: jesus.mejia01@uceva.edu.co. ORCID: https://orcid.org/0000-0002-0799-8294
3BSc. (c) in Systems engineering, Unidad Central del Valle del Cauca, Colombia. Gigae3D Research Group. E-mail: jacobo.grajales01@uceva.edu.co ORCID: https://orcid.org/0000-0001-5331-9769

Cite this article as: L. A. Lasso-Cardona, J. D. Mejia-Vergara and J. Grajales-Cabrera, "Implementation of a queue with priority with parameterizable variables to improve QoS", Visión Electrónica, Special edition, vol. 2, no. 2, 2019.. https://doi.org/10.14483/issn.2248-4728

1. Introduction

In Colombia, the financial system is made up of a set of bodies and institutions, both public and private, through which they capture, manage, regulate and direct the financial resources that are negotiated between the various economic agents. They are people, companies, State and/or public sector [1]. Currently in the country, there are approximately 26 local and international banking institutions that provide such services [2].
At the time of going to any banking institution, there is always a large number of users who require attention for various reasons, so the institutions implement rules such as preferential rows or special turns with the aim of giving priority to pregnant women , seniors (person over 65), clients according to the type of account or affiliation, physical disability, among other criteria, that allow them to attend in the most orderly and timely manner to their users, and thus improve the quality of the service. Also known by its English abbreviation QoS, this has been defined from several points of view giving focus to the scenario where it will be applied [3]. One of the important elements within the cycle of service provision is related to the waiting lines, more commonly called queues. Waiting time is a factor that must be included within the concept of QoS, since this is largely a measure of individual perception of QoS. That is why, it is necessary to design faster systems that balance the waiting time with QoS, and additionally, achieve quality, costs and benefits, become competitive aspects for the organization [4]. In general terms, a waiting line is constituted by a client that requires a service that is provided by a server in a certain period. Clients randomly enter the system and form one or more queues to be served. If the server is unoccupied, according to certain pre-established rules with the name of discipline of the service, the service is provided to the elements of the queue in a determined period of time, called service time and then they leave the system. Waiting lines can be classified according to: 1) The number of clients that can wait in the queue and population. 2) Interval of time elapsed between one client and another. 3) The time of service. 4) The discipline of the queue, which can be: First-In, First-Out (FIFO), Last-In, First-Out (LIFO), with priority or random service. 5) The number of servers. 6) The structure of the service stations. 7) The stability of the system [5]. In this aspect, disciplines such as queuing theory help to understand the behavior of queues by examining all the components of the waiting lists that must be attended to, including the arrival and service process, and the number of clients and places in the system, with the aim of finding the most desirable and supreme level of service [6]. The objective of the theory of queues is to model waiting systems that basically have the following behavior: 1) There is a medium to which customers request a certain service. 2) Then, as a result of which the demand cannot be satisfied immediately, a queue (or waiting line) of clients waiting to be served by the corresponding server (s) is formed. 3) The times between the arrival of consecutive clients to the system and the service times are random, and are represented by random variables with some probability distribution [7].

As mentioned previously, one of the most used disciplines for the management of a queue, is the enqueue with priority. In the priority queuing technique as the packet (client) arrival at the node it is given first to the priority class. Then packets (clients) are stored in the queue on the basis of priority class. Each priority class has its separate queue to store the packet where they waiting for the processing. If the queue is full then it discarded the packet. In the priority queuing higher priority packets are first executed than the lower one [8]. In particular, this article presents the implementation of a priority queue with parameterizable components that allow users to be managed by establishing priority levels that will be assigned to customers in the waiting line in a banking institution that applies preference levels and, in which four main elements were used: 1) the turn dispenser, 2) the cashier in charge of the service, 3) the server that receives and assigns the priority of the turns (depending on the value of the preference), and 4) a screen that displays the order of the turns (priority queue). For this, four algorithms were designed that were implemented in four different applications developed in the Java programming language and that are communicated through TCP/IP connections in a LAN network. Subsequently, a test of tests was carried out in order to validate the behavior of the system. Finally, with the obtained results, the correct functioning of the application in general was evidenced, according to the assigned attention policies, allowing to increase QoS.

2. Methodology

The research was carried out using mainly a quantitative approach. This approach is used in processes that by their nature can be measurable or quantifiable. Take numerical measurements as a center of your research process, use observation of the process in the form of data collection and analyze them to answer your research questions [9]. It is based on which part of an idea is limited and, once delimited, research objectives and questions are derived, the literature is revised and a theoretical framework or perspective is constructed. Variables are determined; a plan is drawn up to test them (design); the variables are measured in a specific context; The measurements obtained are analyzed using statistical methods, and a series of conclusions are drawn. The main goal of this type of research is the formulation and demonstration of theories, accurately establish patterns of behavior of a population or system, and seeks that the studies carried out can be replicated [10].

2.1. The problem of research

Use queue theory as a mechanism for the implementation of a priority queue with parameterizable components that allow managing the waiting lines in the scenario of a banking institution that applies preference levels to assign turns to its clients that allow the improvement of the quality of service.

2.2. Quality of service

When talking about quality of service, it must be taken into account that its application in the literature is varied [11]. The quality of the service has been established as a group of factors that tend to have an important effect on the satisfaction and retention of customers and users. Thus, when a user perceives that the offered attributes are adequately fulfilled, this helps to generate in them the perception of compliance with their expectations, as well as a general feeling of satisfaction [12]. This requires that the service processes of a service organization must be continuously improved in order to guarantee their quality and adaptation to the changing demands of their users [13].

2.3. Queueing theory

The field of queuing theory originated in the early 1900s and is well established with applications in diverse areas including manufacturing, computing, healthcare and telecommunication [14]. The theory of queues is a discipline, within operational research, which aims to study and analyze situations in which there are entities that demand certain services, so that this service cannot be satisfied instantly, which causes a waiting line [15]. A queue is a waiting line and the theory of queues or waiting line phenomena, is a collection of mathematical models that describe particular waiting line systems [16].

The structure of a queue system can be summarized in four main elements, common in all the waiting lines. These are: 1) An input, or customer population. This is the source of the arrivals to the system, it may be finite (limited number of clients) or on the contrary it is infinite (number of clients does not affect the generation rate of new clients). 2) A waiting line formed by customers. The rows of income to a system are designed in the form of a single or multiple. 3) The server or installation of the service, can be constituted by a person, a machine or a combination of both if the process requires it. 4) Priority rule for the selection of clients that enter the system. The most common order is FIFO. In the field of administration of operations there are different entry rules like EDD or priority by expiration date, SPT or shorter processing time, all of them with the only purpose of allowing the organization to improve the perception of waiting by the client [17]. The basic process (Figure 1) assumed by most queue models is that customers require a service, they are generated over time in an input source, then they enter the system and join the queue, at a certain moment a member of the queue is selected to provide the service, through some rule known as the discipline of the queue, the service that the client requires through a service mechanism is carried out and subsequently the client leaves the system [18].

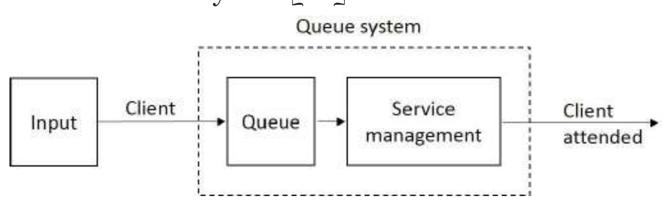

Figure 1. Basic process queuing systems, [19]

It is valid to clarify that this theory does not concern optimal decisions. Instead, it provides information about the behavior of the queue system, to improve it based on the satisfaction of customers or users, and the performance of those offering the service [20]. The most common analytical solution of these models proposes that both the distributions of arrival times and service times follow the exponential continuous probability distribution; this means, by theorem, that the arrival and service events follow the discrete Poisson distribution. Figure 2 presents a characteristic analytical model with this behavior and nature.

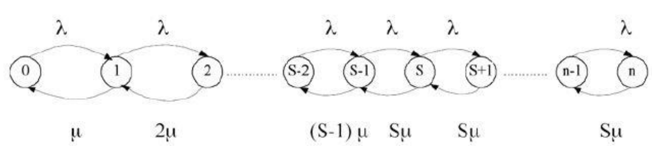

Figure 2. Characteristic analytical model, [21]

The rate diagram:
Being [21]:

$$\lambda_n = \lambda$$

$$\mu_n \begin{cases} n\,\mu \; for \; n = 1, 2, \dots. S \\ S\,\mu \; for \; n^3 \; S \end{cases} \quad (1)$$

2.4. Queue concept

A queue is a data structure of FIFO type that stores elements in a list and allows access to the data by one of the two ends of the list. An element is inserted into the queue (end part) of the list and deleted or deleted from the front (initial part, front) of the list (Figure 3). Applications use a queue to store items in their order of appearance or concurrency [22].

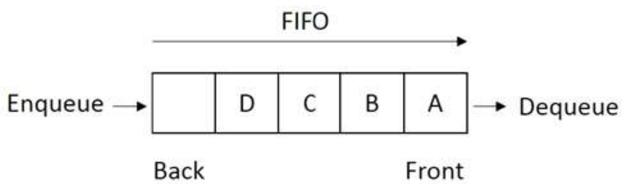

Figure 3. Representation of a queue. Source: own.

FIFO describes the principle of a queue or first-come first serve behavior: what comes in first is handled first, what comes in next waits until the first is finished etc. Thus it is analogous to the behavior of persons "standing in a line" or "queue" where the persons leave the queue in the order they arrive. FIFO is the most basic queuing discipline [23].

2.5. Priority Queue (PQ)

Apriority queue is a data structure similar to a queue, but each element has a "priority". In a priority queue, the element with the highest priority is removed before any other elements [24]. These data structures are designed to ensure that the element at the front of the queue is the largest of all the elements it contains, according to some total ordering defined by their priority, can be given as time-of-occurrence, level-of-importance, physical-parameters, delay/latency, etc. This allows for elements of higher priority to be extracted first, independently of the order in which they are inserted into the queue [25]. In consequence, priority queuing is the basis for a class of queue scheduling algorithms that are designed to provide relatively simple method of supporting differentiated service classes [26].

Although historically, priority queues have been analyzed under the assumption that customer classes have fixed priorities, and that no customer of a given class is admitted to the service while there are customers of higher priority classes, in many situations, this type Priority queuing discipline is not appropriate. Therefore, there is no reason to expect a discipline of absolute priority to produce performance levels that satisfy the service requirements. Therefore, it is desirable to look for a modification of the classical structure, which would allow the administrator of a queue to adjust the discipline of selecting the client so that the service requirements are met [27]. Two basic operations are supported by priority queue, namely (1) INSERT (Q, x), which is generally referred to as ENQUEUE operation. This operation inserts a new element x into priority queue Q. And (2) EXTRACT (Q) which is sometimes referred to as DEQUEUE operation. This operation returns the element with the highest priority [28].

In classic PQ, packets are first classified by the system and then placed into different priority queues. Within each of the priority queues, packets are scheduled in FIFO order, as shown in Figure 4 [29].

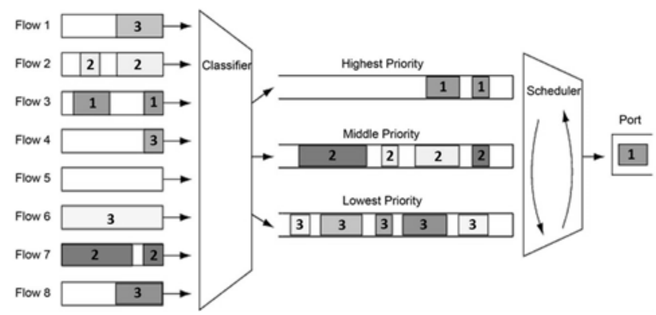

Figure 4. Classical PQ, [29].

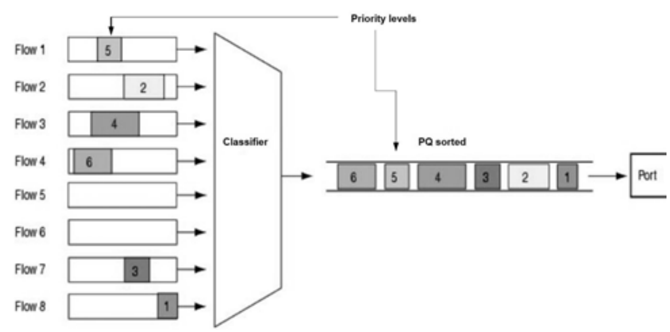

Figure 5. PQ sorted. Source: own ▪

The queue implemented in the present work is similar to that shown in Figure 3, by far, in that a single queue is used to store the sorted data according to the priority level of each of them. Figure 5 exemplifies its operation.

A standard implementation using a priority queue with heapsort will have complexity O (n log (n)), where n = number of elements in the queue. This is because performing an operation on a priority queue has complexity O (log (n)), which is done once per node. That is, an operation is performed n-1 times O (log (n)), which has complexity O (n log (n)) [30]. In our case, the dequeue process will have a complexity of O (1), since only the element that is on the front is extracted. O (n) at insertion time since, in the worst case, the new element must be inserted at the front of the queue due to the value of the priority, which implies a displacement in one position of all the elements of the queue.

2.6. Structure of algorithm

In general terms, the policies for assigning turns in the waiting queue are: physically disabled persons, pregnant women, older adult and type of client. In the latter case, customers are classified according to seniority in “Premium” or “Normal”.

2.6.1. Behavior of the queue with respect to the type of account

Regardless of the size of the queue, each time a client with a "Premium" account type is present, it will be first in the queue. Among the clients with this same type of account, the FIFO discipline will apply. This implies a process of displacement in the queue that repeats:

$$df = (t - cp) \quad (2)$$

Being:
df= displacement factor in the queue, *t*= queue size and *cp*= number of customers in the queue with "Premium" account type.

For “Premium” customers who also have physical disabilities, and according to the institution's service policies, this will be first in line, presenting a higher priority than the previous case. Likewise, for the latter case FIFO is applied if the same condition is present. This implies a process of displacement in the queue that repeats:

$$df = \big(t - (cp + cps)\big) + cp \quad (3)$$

Being:
df= displacement factor in the queue, *t*=queue size, *cp*= number of customers in the queue with "Premium" account type and *cps*= number of customers in the queue with "Premium" account type.

On the other hand, the value of the priority assigned to the "Normal" account type is the lowest in the PQ. The administrator of the platform can modify the value of the priority assigned to the “Premium” type account.

2.6.2. Behavior of the queue with respect to the special condition

In this type of condition, the clients are classified: older adults, physically disabled persons and pregnant women. The algorithm uses a value k that represents the position from which it begins to assign a place within the PQ for the previous conditions (point 2.5.3 explains in detail the behavior of the variable *k)k.* can be modified by the administrator of the platform. The condition of older adult has the highest priority value. With respect to the other conditions, the FIFO discipline will be applied.

2.6.3. Queue size

When the queue has size from 1 to 6 clients there is no priority for the special condition. In this case, customers with a "Premium" account type have the highest priority value, and therefore their position will be assigned to the front of the PQ. The value 6 is represented by the variable *k* described above. It means then that from the position *k=6* the people of special condition will begin to have preference, and will be located from position k through a process of insertion in the PQ. This implies a process of displacement in the queue that repeats:

$$df = \big(t - (k + c)\big) + 1 \quad (4)$$

Being:
df= displacement factor in the queue *t*= queue size *k*= position from which the preference applies and *c*= number of customers in the queue with special condition.

2.7. Implementation

A priority queue was implemented with parameterizable components that allow users to be managed by establishing priority levels that will be assigned to clients in the waiting line in a banking institution that applies levels of preference and in which four main elements were identified described below. An algorithm was designed for each element. Later they were implemented in four different applications developed in the Java programming language that communicate through TCP/IP connections in a LAN network.

2.7.1. Dispenser

The “Dispenser” is the application by which the user generates a turn in the queue by its identification. This is sent to the system, which is responsible for consulting the information in the database. In the event that the user is affiliated for a turn in the queue (with a certain priority) assigned to the service policies (Figure 6). Figure 7 shows the screenshot of the "Dispenser" application.

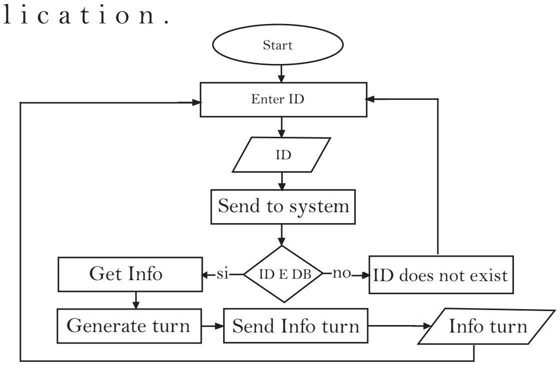

Figure 6. Flowchart-Dispenser. Source: own.

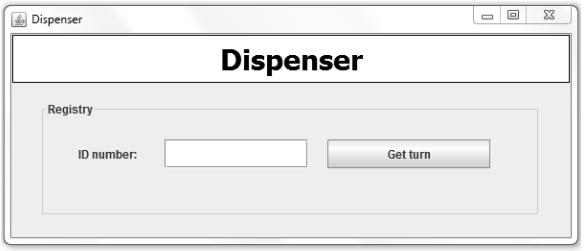

Figure 7. Screenshot-Dispenser. Source: own.

2.7.2. Screen

Application that shows the turns to attend, the current turn and attended turns (Figure 8). It uses a TimerTask whose function is to send a string to the system, and every time the system receives information, it sends all the information to the screen. Figure 9 shows the screens hot of the "Screen" application.

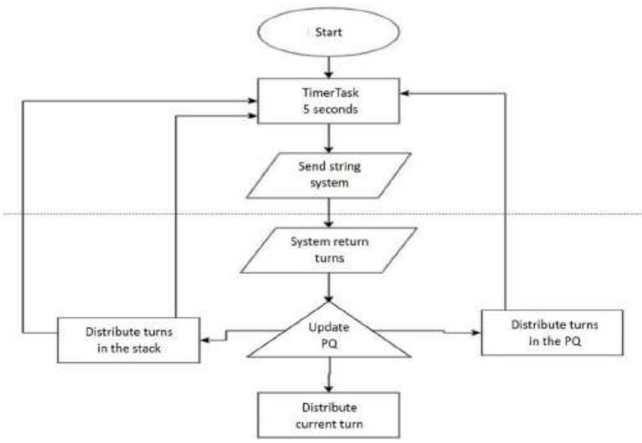

Figure 8. Flowchart-Dispenser. Source: own.

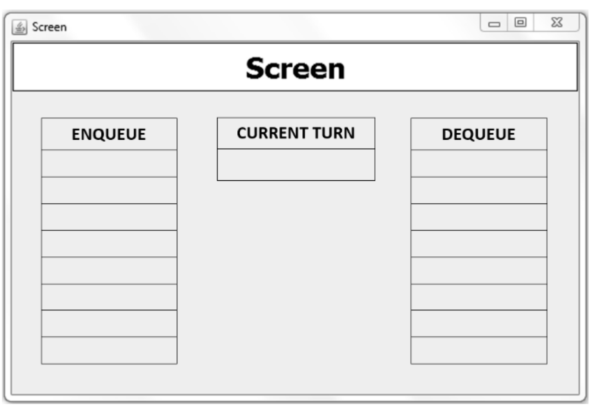

Figure 9. Screenshot-Screen. Source: own.

2.7.3. Cashier

The function of the "Cashier" application is to attend the waiting turns. Through an action, it sends information to the next turn system, eliminates it from the PQ and adds it to the stack after it is attended (Figure 10). Figure 11 shows the screenshot of the "Cashier" application.

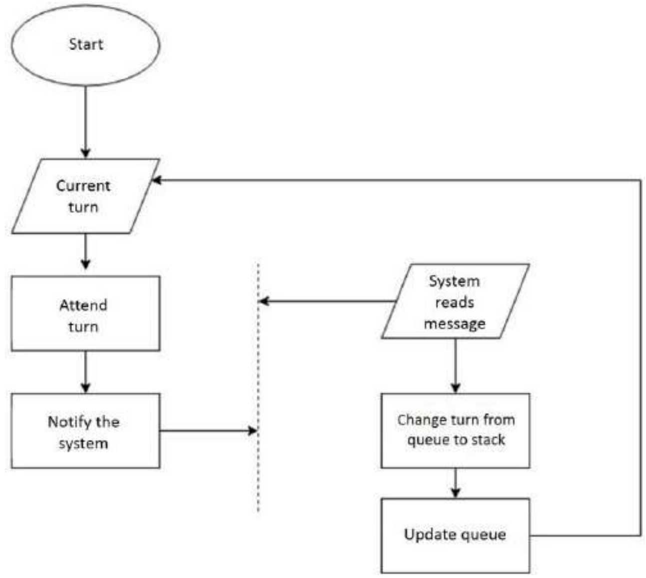

Figure 10. Flowchart-Cashier. Source: own.

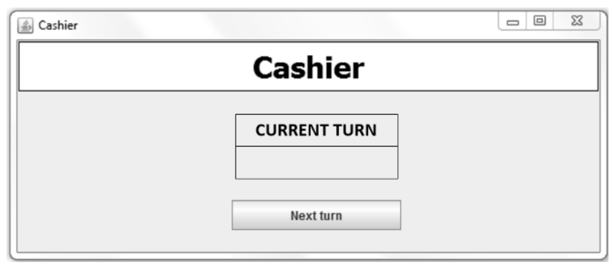

Figure 11. Screenshot-Cashier. Source: own.

2.7.4. System

It is the main application of the whole project. It is responsible for receiving data from the other components, to process them and then return the order in the PQ. The "System" has certain specific tasks in relation to the other applications: 1) System-Dispenser: "System" receives an identification number entered in "Dispenser". Perform a search in the database and if it does not exist, returns an error message. Otherwise, the type of disability and type of account ("Normal" or "Premium") is analyzed. Based on this and the service policies, it determines the priority and position within the PQ. 2) System-Screen: "Screen" using a TimeTask and a Socket sends an update request to "System". This returns the PQ, the current turn and the turns attended, to be shown in "Screen". 3) System-Cashier: "Cashier" through a Socket sends "System" a request to update the turn to be attended. "System" removes (dequeue) the current turn, and moves in one position the turns in the PQ waiting to be attended.

3. Results and discussion

In order to validate the behavior of the platform, a series of tests were carried out, each focused on verifying the behavior of the queue with respect to the application of priorities for the allocation of shifts according to the policies of attention of the banking institution. Next, the description of each test and its results.

3.1. Test N.1A - Premium preferences

Priority validation for the "Premium" account type was the first to be carried out, since turns of this type will always be at the front of the priority queue, so it was a simple test to perform.
Input data: a queue with 6 customer turns, which were denominated: *N1*, *N2*. *N3*. *N4*, *N5* and *N6* The process can be observed in Table 1 and Figure 12.

N1	N2	N3	N4	N5	N6
Front					Back

Table 1. Test 1A-1. Source: own.

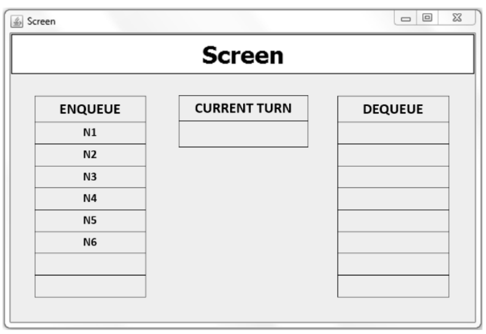

Figure 12. Screenshot-Screen. Test 1A-1. Source: own.

Subsequently, a client with a "Premium" account type named *N7* requests a turn; the preference algorithm assigns the priority and reorders the queue. The process can be observed in Table 2 and Figure 13.

N7	N1	N2	N3	N4	N5	N6
Front						Back

Table 2. Test 1A-2. Source: own.

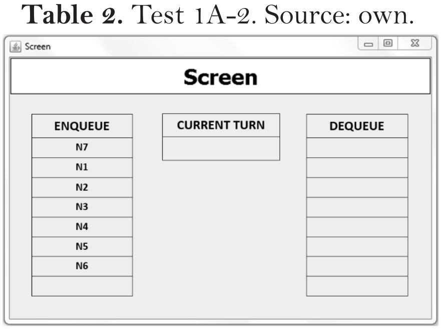

Figure 13. Screenshot-Screen. Test 1A-2. Source: own.

The results obtained when adding the "Premium" client were correct, since the algorithm gave priority to the type of "Premium" account according to the service policies.

3.2. Test N.2A - Premium preferences

For this test, the result of test 1A is based on. Since the algorithm is designed to generate turns according to the types of preferences that are in the queue, it is sought to insert another client of type "Premium". Then the FIFO discipline is applied among these.
Input data: queue resulting from test 1A-2 and "Premium" turn request marked as *N8* The process can be observed in Table 3 and Figure 14.

N7	N8	N1	N2	N3	N4	N5	N6
Front							Back

Table 3. Test 2A. Source: own.

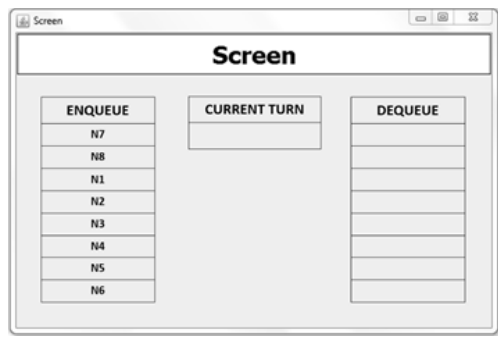

Figure 14. Screenshot-Screen. Test 2A. Source: own.

The result allows to demonstrate the correct application of the FIFO discipline by the algorithm between priorities of equal value, as in this case, "Premium" clients.

3.3. Test N.3A - Premium preferences with special condition

In the "Premium" category there are also clients with physical disabilities. In this case the algorithm will assign a higher priority value that allows this turn to be inserted in the head of the queue.
Input data: queue resulting from test 2A and "Premium" turn request marked as *D9* The process can be observed in Table 4 and Figure 15.

D9	N7	N8	N1	N2	N3	N4	N5	N6
Front								Back

Table 4. Test 3A. Source: own.

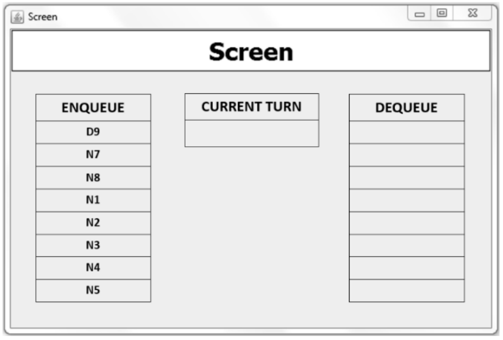

Figure 15. Screenshot-Screen. Test 3A. Source: own.

It is verified that the algorithm assigned the highest value for this priority, allowing to insert the turn at the front of the queue.

3.4. Test N.1B - "Normal" account type and older adults

This test is focused on verifying the priority that must be applied to clients with a "Normal" account type and who are also older adults.

Input data: a queue with 8 customer turns, marked from *N1* Until *N8*, and a turn marked as *T11* The process can be observed in Table 5 and Figure 16.

N1	N2	N3	N4	N5	N6	N7	N8
Front							Back

Table 5. Input test 1B. Source: own.

As explained in point 2.5.3, the algorithm uses the variable k to represent the position within the queue, from which the special preference turn is inserted. In this case *T11* will be inserted in the position *k=6* The process can be observed in Table 6 and Figure 17.

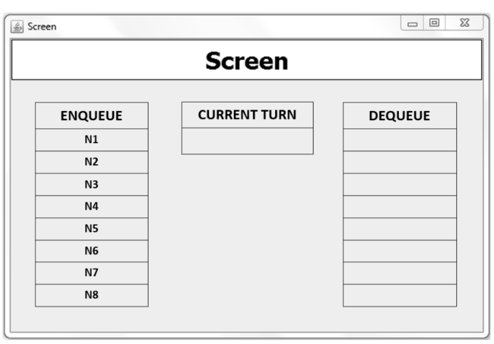

Figure 16. Screenshot-Screen. Input test 1B. Source: own.

N1	N2	N3	N4	N5	N6	T11	N7	N8
Front								Back

Table 6. Test 1B. Source: own.

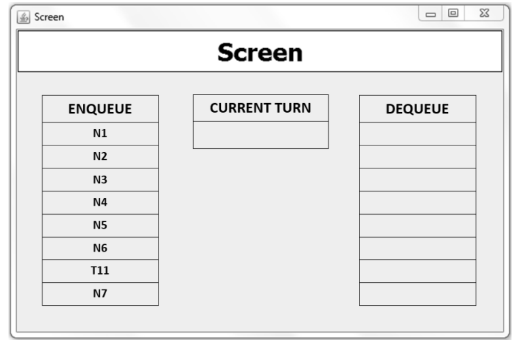

Figure 17. Screenshot-Screen. Test 1B. Source: own

In this case, the algorithm made the insertion of the turn correctly in position *k. I*t is to remember that *k* can be modified by the administrator of the platform.

3.5. Test N.2B - "Normal" account type and older adults

This test seeks to verify the application of FIFO discipline for case *N1B* Input data: queue resulting from test *1B*, and a turn marked as *T12* The process can be observed in Table 7 and Figure 18.

N1	N2	N3	N4	N5	N6	T11	T12	N7	N8
Front									Back

Table 7. Test 2B. Source: own.

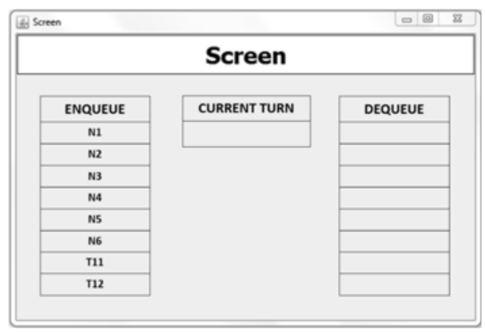

Figure 18. Screenshot-Screen. Test 2B. Source: own.

The results obtained when inserting the turn *T12* were correct, when respecting the order of arrival of the turns, assigning *T12* after *T11*

Since people with physical disabilities and pregnant women with a "Normal" account type share the same priority value in the queue, FIFO discipline applies.

5. Conclusions

The developed project fulfilled the objective as it was set, in the students to lay strong foundations in terms of knowing the philosophy of data structures such as lists, stacks and queues, and apply their use to solve a typical

programming problem implemented an application in Java with database and communication Client/Server using Sockets or a timer (TimerTask).
Although it is true that the implementation of the PQ followed a naive approach, that is to say only the use of primitive structures such as arrays and lists, these allowed the correct simulation of a priority queue that can be taken as a basis for a later application of business type. Likewise, since it is a highly parameterizable algorithm, its adaptation to any other service policy scenario would be very easy for any organization

References

[1] Escuela Nacional Sindical, "Sector Financiero y Bancario Colombiano. Características económicas, laborales y de negociación colectiva", 2015. [Online]. Available at: http://www.ens.org.co/wp-content/uploads/2016/12/DOCUMENTOS-DE-LA-ESCUELA_100-Sector-financiero-y-bancario-colombiano-Econ%C3%b3mico-laboral-y-de-negociaci%C3%b3n-colectiva-2015.pdf

[2] Superintendencia Financiera de Colombia. "Entidades vigiladas por la superintendencia financiera de Colombia", 2019. [Online]. Available at : https://www.superfinanciera.gov.co/inicio/industrias-supervisadas/entidades-vigiladas-por-la-superintendencia-financiera-de-colombia-13067

[3] F. E. Ávila, "Estrategia de calidad de servicio (QoS) para tráfico de aplicaciones de hospitalización domiciliaria", thesis MSc., Universidad de Antioquia, Colombia, 2016.

[4] R. Orrego-Posada, "La calidad del servicio, la gestión de flujos y la teoría de colas", *RUE*, vol. 30, no. 94, pp. 23-19, 2012.

[5] C. E. Martínez-Eraso, "Análisis de redes de colas modeladas con tiempos entre llegadas exponenciales e híper erlang para la asignación eficiente de los recursos", thesis, Pontificia Universidad Javeriana, Colombia. 2009.

[6] W. Ritha and S. J. Vinnarasi, "Analysis of priority queues with pentagon fuzzy number", *IJETMR*, vol. 5, no. 4, pp. 90-101, 2018. https://doi.org/10.5281/zenodo.1250510

[7] E. M. Gámez-Castellanos, "Propuesta de mejora mediante modelo de teoría de colas para el estudio de frecuencias en la empresa transportes Fontibón S.A, ruta ZP- C66", thesis, Universidad Católica de Colombia, Colombia, 2018.

[8] C. J. Ghyar, et al., "Basics of Quality of Services (QoS)", *IJSRST*, vol. 4, no. 7, pp. 105-110, 2018.

[9] M. E. Cortés-Cortés and M. Iglesias-León, "Generalidades sobre Metodología de la Investigación", 1ra ed., Campeche, México: Colección Material Didáctico, 2004.

[10] R. Hernández, et al., "Metodología de la Investigación Científica", 6ta ed., México: Mc Graw Hill, 2014.

[11] C. Villalba-Sánchez, "La calidad del servicio: un recorrido histórico conceptual, sus modelos más representativos y su aplicación en las universidades", *Punto de vista*, vol. 4, no. 7, pp. 51-72, 2013. http://dx.doi.org/10.15765/pdv.v4i7.445

[12] J. Vera and A. Trujillo, "El efecto de la calidad del servicio en la satisfacción del derechohabiente en instituciones públicas de salud en México", *Contaduría y Administración*, vol. 63, no. 2, pp. 1-22, 2018. http://dx.doi.org/10.1016/j.cya.2016.07.003

[13] P. Gutiérrez, et al., "Valoración de los factores determinantes de la calidad del servicio público local: un análisis de la percepción de los ciudadanos y sus repercusiones sobre la satisfacción y credibilidad", *Innovar*, vol. 20, no. 36, pp. 139-156, 2010.

[14] A. Komashie, et al., "An Integrated Model of Patient and Staff Satisfaction Using Queuing Theory", *IEEE Journal of Translational Engineering in Health and Medicine*, vol. 3, 2015.

https://doi.org/10.1109/JTEHM.2015.2400436

[15] J. E. Arias-Caro and M. P. Correa-Fuenzalida, “Estudio de la teoría de colas como una metodología en la optimización de tiempo del departamento de control en la municipalidad de San Nicolás, provincia de Ñuble”, thesis, Universidad del Bio-Bio. Chile, 2016.

[16] J. Arista-Arévalo, “Aplicación de la teoría de colas al problema de atención al cliente para la optimización del número cajeros en ventanillas en la organización BCP”, thesis, Universidad Nacional Mayor de San Marcos, Perú, 2016.

[17] C. E. Rojas-Franco, “Optimización del proceso de cargue de mercancías En Envía - Colvanes S.A.S”, thesis, Universidad Libre, Colombia, 2013.

[18] J. G. González-Bravo, “Propuesta de mejora del sistema de recepción de usuarios del centro de servicios administrativos jurisdiccionales Hernando Morales Molina de Bogotá”, thesis, Universidad De La Salle, Colombia, 2017.

[19] F. Hillier and G. Lieberman, “Introducción a la investigación de operaciones”, 9a ed., México: Mc Graw Hill, 2010.

[20] E. López-Huang and L. G. Joa-Triay, “Teoría de colas aplicada al estudio del sistema de servicio de una farmacia”, *Revista Cubana de Informática Médica*, vol. 10, no. 1, 2018.

[21] R. Terrazas-Pastor, “Aplicación de la simulación a un sistema de colas de canal simple”, *Revista Perspectivas*, no. 26, pp. 91-112, 2010.

[22] L. Joyanes-Aguilar and I. Zahonero-Martínez, “Estructuras de datos en Java”, 1ra ed., Madrid, España: Mc Graw Hill, 2008.

[23] E. O Momanyi, et al., “QoS Performance Comparison of FIFO and Priority Packet Queuing Mechanisms in MANETs”, *Journal of Sustainable Research in Engineering*, vol. 1, no. 3, pp. 39-44, 2015.

[24] M. Dorin, “Kipu - Lightweight threads for Java”, *Interfaces*, no. 7, pp. 79-86, 2014. http://dx.doi.org/10.26439/interfases2014.n007.143

[25] J. Ros-Giralt, et al., “Multiresolution Priority Queues and Applications”, 2017. [Online]. Available at: https://arxiv.org/ftp/arxiv/papers/1705/1705.09642.pdf

[26] M. El Gili-Mustafa and S. A. Talab, “The Effect of Queuing Mechanisms First in First out (FIFO), Priority Queuing (PQ) and Weighted Fair Queuing (WFQ) on Network's Routers and Applications”, *WSN*, vol. 8, no. 5, 2016. http://dx.doi.org/10.4236/wsn.2016.85008

[27] D. A. Stanford, et al., “Waiting time distributions in the accumulating priority queue”, *Queueing Systems*, vol. 77, no. 3, pp 297-330, 2014. https://doi.org/10.1007/s11134-013-9382-6

[28] C. H. Sun, et al., “Accelerating Graph Algorithms with Priority Queue Processor”, Regional Postgraduate Conference on Engineering and Science (RPCES), 2006.

[29] M. Goyal, et al., “Modelling & simulation of queuing disciplines over the n/w carried applications (ftp, video and voip) for traffic dropped & time delay”, *IJCSMC*, vol. 4, no. 1, pp. 562-570, 2015.

[30] J. A. Cruz-Kouichi, “Desarrollo de un algoritmo de compresión de datos optimizado para imágenes satelitales”, thesis, Universidad Nacional de Córdoba, Argentina, 2017

[31] M.A. Lopez - Inga, R. M. Guerrero-Huaranga, “Cloud Business Intelligence and Analytics Model for smes in the Retail Sector in Peru”. Ingenieria Solidaria, Vol 14, No. 24. p.15. https://doi.org/10.16925/in.v14i24.2157

[32] L. Del Franco y A. Gómez, "Contabilidad ambiental. Una reflexión en el marco de la gestión socialmente responsable de las empresas colombianas". Aglala. vol. 10, no. 2. pp. 60-80. 2019. http://revistas.curnvirtual.edu.co/index.php/aglala/article/view/1432

[33] E. Carballo, "Modelo de negocio para una tienda virtual de venta de llantas al sector público mediante la modalidad de contratación". Conocimiento Global. vol. 2, no. 1. pp. 20-40. 2017. http://conocimientoglobal.org/revista/index.php/cglobal/article/view/14

[34] L. Durán, "Estrategia de gestión basada en el cuadro de mando integral para la empresa Herrametal C.A. ubicada en municipio Iribarren, Estado Lara". Revista Enfoque Disciplinario. vol. 3, no. 1. pp. 49-71. 2018. http://enfoquedisciplinario.org/revista/index.php/enfoque/article/view/12

[35] L. Vargas, C. Lattá, A. Puello, A. Castro, C. Vidal y A. Ruiz, "Estrategias del uso de las tecnologías de la información y comunicación para proyectos productivos en instituciones educativas rurarles en Colombia", en Gestión del conocimiento. Perspectiva Multidisciplinaria, vol. 18. V. Meriño, A. Martinez, A. Cruz, L. Morán, H. Urzola y M. Pérez, Eds. Colombia: Fondo Editorial Universitario dela Universidad Nacional Experimental Sur del Lago Jesús María Semprúm. 2020. p. 341

[36] Castro Alfaro, A. (2016). Estrategias para la disminución de la carga impositiva en las organizaciones empresariales. Enfoque Disciplinario, 1(1), 21-34. Recuperado a partir de http://enfoquedisciplinario.org/revista/index.php/enfoque/article/view/6