TEM Journal. Volume 13, Issue 2, pages 1095-1105, ISSN 2217-8309, DOI: 10.18421/TEM132-24, May 2024.

Method for Optimizing Incident Handling Process for Telephony Outsourcing Companies

Jose Neyra-Cruz [1], Jaime Haro-Enríquez [1], Alfredo Daza [2], Fernando Asin [1], Nemias Saboya [1]

[1] *Faculty of Engineering and Architecture, Professional School of Systems Engineering, Universidad Peruana Unión, Lima, Perú*
[2] *Faculty of Engineering and Architecture, Professional School of Systems Engineering, Universidad César Vallejo, Lima, Perú*

Abstract **– Communication service companies strive to maintain quality service, which implies having strategies that facilitate timely attention to incidents. In this context, this study proposes a method to improve incident management in an IP voice communication outsourcing company. The method was developed in three phases: preliminary analysis, improvement, and comparison. The preliminary analysis phase involved identifying the problem, defining indicators, conducting pre-measurement and developing solution proposals. The improvement phase included planning (defining requirements), designing the technical solution, coding and integrating the method, and testing it. Finally, the comparison phase evaluated the results before and after the implementation of the method using four indicators: incidents in queue for attention, incidents resolved outside of business hours, response time to Lightweight Directory Access Protocol (LDAP) incidents, waiting time to be served. The sampling for the study was differentiated for each indicator and data were collected over a period of three months for "pre" and five months for "post". The results showed the complete elimination of waiting time for AD user support. Additionally, the time to resolve incidents was reduced to 36 seconds. The workload of the attentions was also considerably reduced thanks to the automation of the process.**

Keywords **– Active directory, Asterisk, LDAP, integration, incident management.**

1. Introduction

Incident management is crucial to ensure quality service for customers [1], [2], [3] and to achieve this it is necessary to have optimal response times. Customer service will depend on the instruments and strategies that the company uses to offer a quality service [4], [3].

Nowadays, companies that provide customer service to banks, mining entities, among others, have a telephone exchange [5] and a call center called "help desk" [4], in order to resolve different incidents reported by their customers and / or workers [6], [7], [8].

In a study conducted by Lida [1], it was found that the response times to solve incidents were excessively long. In response to this problem, the importance of Information and Communication Technologies in incident management is highlighted, as they can contribute to achieving optimal solutions more efficiently.

On the other hand, Vargas [6] in his research also addressed the long response times in the attention of incidents in an outsourcing service desk. In addition, he highlighted the importance of technology to improve incident resolution processes, with the aim of speeding up response times and improving the quality of service.

Telecommunications have evolved greatly over the years [3], [8] and one of the largest telephony platforms in the world is Asterisk and being open source makes it easier to develop and integrate modules [9], [10], [11]. On the other hand, the use of directories in the business universe has expanded greatly [12] one reason for this is the current tendency to offer increasingly personalized services [9].

DOI: 10.18421/TEM132-24
https://doi.org/10.18421/TEM132-24

Corresponding author: Nemias Saboya,
Facultad de Ingeniería y Arquitectura Universidad Peruana Unión, Lima, Perú
Email: saboya@upeu.edu.pe

Received: 09 November 2023.
Revised: 12 February 2024.
Accepted: 20 March 2024.
Published: 28 May 2024.

In this sense, the use of technologies and knowledge in programming become key tools to optimize processes [2], [3].

In this context, reports of common and repetitive incidents with respect to users have been identified, such as the unlocking of accounts and the change of network keys, whose attention and waiting time generates delays in the attention of other incidents, such as corporate mail problems, application errors, and configuration changes among others [1]. This causes an increase in queue calls, which has generated an excessive demand for time in the attention of incidents of users of the active directory (AD) in the outsourcing service desk [13].

One of the main reasons why this problem is generated is because the process of attention to incidents of LDAP users is not automated [14], [15] in the outsourcing company of voice over IP communication [13], [16]. Currently, these attentions have been carried out manually by the agents.

In this sense, it is essential to have a specialized tool to carry out efficient management [8] and optimize response times and care provided [6]. Therefore, it is necessary to optimize the process in such a way that automated operations can be carried out [14], [15], unlocking accounts or changing network keys. The use of programming languages such as PHP and the use of libraries such as php-agi [17] and php-ldap, can facilitate integrations between an IP PBX [13], [16], [5] and an AD, which most companies have implemented [9].

The importance of the study lies in the development of a module that integrates Asterisk and Active Directory and that will help voice over IP communication outsourcing companies to manage their incident attention services in an automated and timely manner.

The main objective of this research is to propose a method that guarantees the improvement of incident management in an outsourcing company of voice over IP communication through the integration of Asterisk and Active Directory [9], [18], in addition, the study seeks to demonstrate that the method contributes to the reduction of service and waiting times post-implementation of the method.

The work is structured as follows: Section 1 presents the introduction. Section 2 describes the methodology used in the research. Section 3 presents the case study, which includes the pre-post analysis and the technical improvement implemented. Subsequently, the last section presents the main conclusions and findings obtained in this study.

2. Methodology

The study was based on the principles of applied research [19],[20], using scientific knowledge and theories to solve a practical problem in the management of the incident attention service. The study developed an innovative method that integrates Asterisk and Active Directory as a technological solution.

The methodological sequence for the development of the method was based on three phases: preliminary analysis, improvement and comparison [19], [21], see Figure 1.

Phase 1: Preliminary analysis. This phase consisted of problem identification, definition of indicators, pre-measurement and development of solution proposals.

Phase 2: Improvement. This phase included planning, where requirements were defined, design of the technical solution, coding for method integration and method testing.

Phase 3: Comparison. This phase evaluated the results before and after the implementation of the method, using the indicators defined in phase 1.

This study encompasses the methodological requirements of an applied research type.

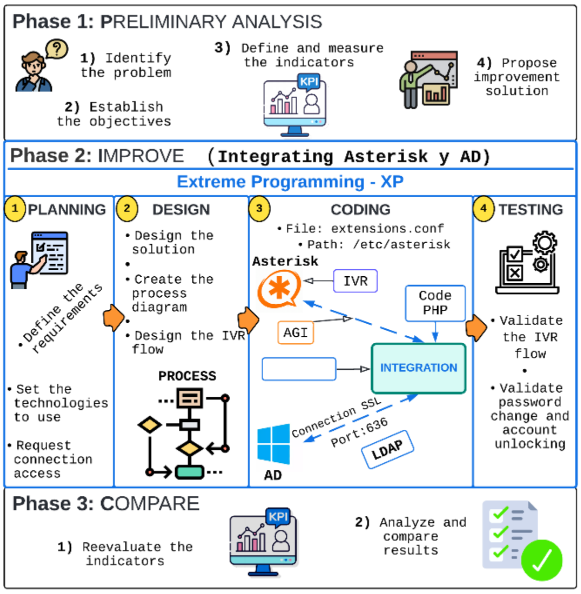

Phase 1: PRELIMINARY ANALYSIS
1) Identify the problem
2) Establish the objectives
3) Define and measure the indicators
KPI
4) Propose improvement solution
Phase 2: IMPROVE (Integrating Asterisk y AD)
Extreme Programming - XP
1 PLANNING
• Define the requirements
• Set the technologies to use
• Request connection access
2 DESIGN
• Design the solution
•
• Create the process diagram
•
• Design the IVR flow
PROCESS
3 CODING
• File: extensions.conf
• Path: /etc/asterisk
Asterisk
IVR
AGI
Code PHP
INTEGRATION
Connection SSL
Port:636
LDAP
AD
4 TESTING
• Validate the IVR flow
•
• Validate password change and account unlocking
Phase 3: COMPARE
1) Reevaluate the indicators
KPI
2) Analyze and compare results

Figure 1. Phases for the development and evaluation of the method

2.1. Preliminary Analysis

At the beginning of the study, an analysis of the current situation of the voice over IP communication outsourcing company was carried out by reviewing call records and the time required to attend them.

This analysis allowed us to identify a problem in incident management: an excessive demand of time in the incident attention to AD users, especially in the outsourcing service desk. In response to this problem, the objectives of the study were proposed in order to solve it.

Based on the above, four improvement indicators for time management were defined. These indicators were selected to measure and evaluate the achievement of the study's objectives, as shown in Table 1 below:

Table 1. Indicators

Unit	Indicator
Number	Incidents queued for attention
%	Incidents resolved outside working hours
Seconds	LDAP incident response time
Seconds	Waiting time to be served

On the other hand, the most common incidents recorded in the voice over IP communication outsourcing company were categorized. The study focused on "Unblocking and changing network keys" incidents (AD users).

To determine the average number of incidents recorded, data were collected randomly from the incident attention system for 41 days over a period of 3 months, and this information was considered as previous data for the study (PRE). The results showed an average of 73 daily incidents of "Unblocking accounts and/or changing passwords", 427 daily incidents of "Various Attentions" and a total average of 500 daily incidents in the outsourcing help desk (Table 2).

Table 2. Number of incidents registered per day

Incidents	Quantity
Account unblocking and/or password change	73
Various attentions	427
Total Incidents	500

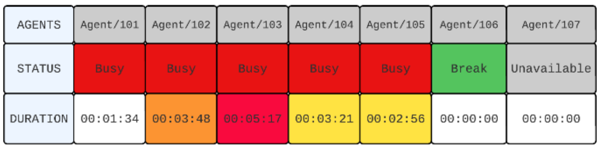
Table 3 shows the average results of waiting time and resolution time for "Account unblocking and/or password change" incidents handled by the operators. These results were obtained from the previous data (PRE), collected from the incident attention system during three months and with 1,572 incident records.

The data reveals that customers experience long waiting times, with an average of 350 seconds in the queue before receiving attention. This is due to the high demand for incidents requiring telephone assistance. In addition, the average time to resolve "Account unlocking and/or password change" incidents is 316 seconds, which is not ideal.

Table 3. Average duration and waiting time for incident attention

Attentions	Duration	Wait
Account unblocking and/or password change	316s	350s
Various attentions	380s	400s

Likewise, it was possible to identify that the attention was conditioned to be attended by trained personnel; which generates an expense for the company. In turn, this staff has a defined work schedule at the help desk. In this sense, care is limited to one opening hours; which means that some incidents reported outside those hours have to wait until the next day to be attended.



AGENTS	Agent/101	Agent/102	Agent/103	Agent/104	Agent/105	Agent/106	Agent/107
STATUS	Busy	Busy	Busy	Busy	Busy	Break	Unavailable
DURATION	00:01:34	00:03:48	00:05:17	00:03:21	00:02:56	00:00:00	00:00:00

Figure 2. Busy agents in real time

Figure 2 shows a real-time visualization of agent status and call duration, revealing a worrying burnout of incident staff. Therefore, this causes an operational over-effort due to the high demand for incidents attended at the service desk. In addition, the attention of incidents for AD users is not automated, since the attentions are carried out manually. That is why automating this process in the attention allows reducing that load of daily incidents.

On the other hand, with the previous data (PRE) collected during the three-month period, the current status was evaluated for each indicator previously defined in Table 1. The results were measured descriptively by means of averages and The following are the results obtained (Table 4):

Table 4. Previous results (PRE) per indicator

Indicators	Frequency	Value
Number of incidents in queue for attention	Diary	73
Percentage of incidents resolved outside working hours	Monthly	0%
LDAP incident response time	Diary	316 sec
Waiting time to be attended	Diary	350 sec

On the other hand, Figure 3 shows that there is a considerable number of daily incidents related to Active Directory (AD) users, these incidents are related to calls to unlock accounts or change network passwords.

These incidents represent approximately 15% of all incidents reported per day, while the remaining 85% correspond to other requests for daily attention.

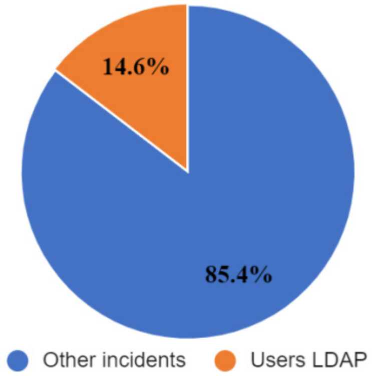


Figure 3. Daily incidents

In addition, as detailed in Figure 4, there is an average waiting time of about 316 seconds to be attended, while a delay of 350 seconds in the service to solve the problem of unlocking an account or changing the password, this time is excessive, which has a negative impact on the attention to other customers and sometimes, it results in a decrease in the quality of service.

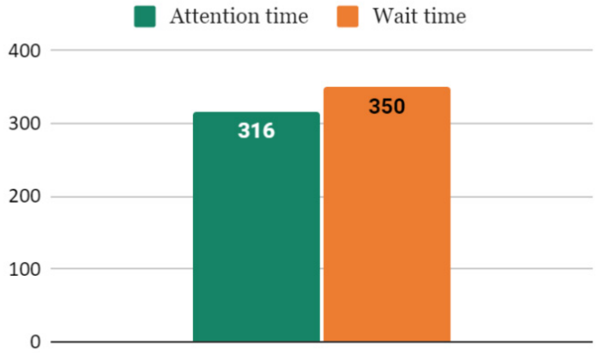


Figure 4. Average attention and waiting time in seconds

After performing the analysis, we proceeded to design the flow of the current process, of the attention that agents perform to customers and / or users via telephone call with respect to any incident, as shown in Figure 5.

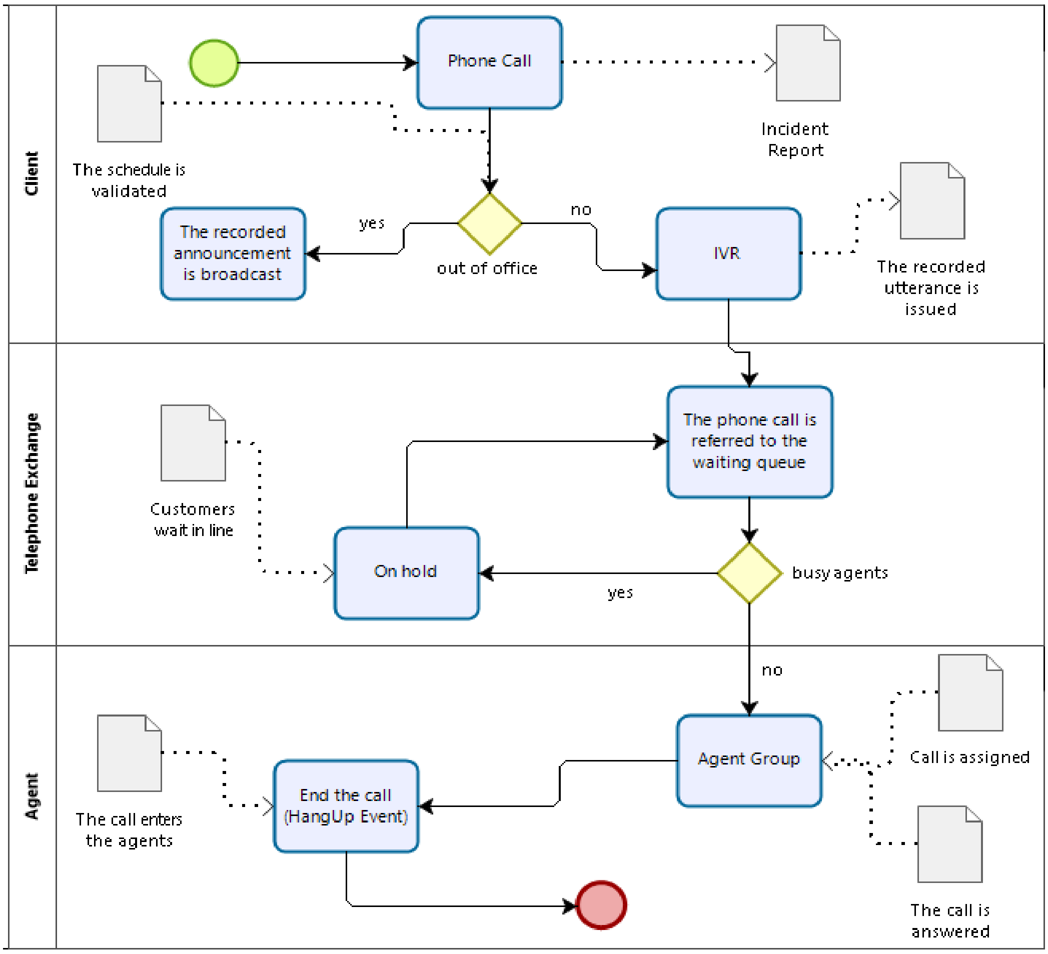

Client
Telephone Exchange
Agent
Phone Call
Incident Report
The schedule is validated
yes
no
The recorded announcement is broadcast
out of office
IVR
The recorded utterance is issued
Customers wait in line
The phone call is referred to the waiting queue
On hold
yes
busy agents
no
Agent Group
Call is assigned
The call is answered
The call enters the agents
End the call (HangUp Event)

Figure 5. Current process diagram

- Improvement proposal

Automate the process of attention to incidents of AD users, by implementing a method that allows integrating Asterisk and active directory through the development of a module.

2.2. Improve

At this stage a previous study was carried out, referring to the operation and use of the technologies that were used for the implementation and development of the proposed technological solution.

At this stage, the configuration of Asterisk was carried out and the code and implementation of the "incident attention module" system was developed; thus, integrating Asterisk and AD using the LDAP protocol [27], [28] to access the directory service. For this, the XP development methodology [22], [23] and the activities carried out for each stage were as follows:

- ***Planning***

In this section, a list of activities with respect to the requirements of the module was defined through the user stories [22] which were developed sequentially as shown below:

1. Create and configure the IVR [17], [11].
2. Generate AGI and pass data from Asterisk to the developed module and vice versa [17].
3. Connection with the AD server [9].
4. Search and obtain user data [9], [15].
5. Unlock an LDAP account from the Asterisk server [9], [15].
6. Generate new user keys [15].
7. Change a network key, from the Asterisk server [9], [15].
8. Send SMS to users [29].
9. Save records in the database [29].

To access active directory, the LDAP protocol was established over TCP port 636 [15], [30], [31] since Windows Server supports LDAPV3 [9].

Table 5 lists the data on the versions of technologies used to develop the study.

Table 5. Software versions used

Technology	Version
(AD) Windows Server	2012 R2
Asterisk	13
PHP	5.6
Database	María DB 5.4
LDAP	V3

Likewise, the data required in Table 6 was taken into account, to achieve the unlocking and / or change of key of the accounts of AD users.

Table 6. Information required for the implementation of the incident handling module

Data required
Base DN and Bind DN (Active Directory)
Name of the field where the ID of the LDAP user is saved
Name of the field where the LDAP user's mobile number is located
Credentials of an LDAP user, to connect to the AD
API for sending SMS
IP of the servers (Asterisk and AD)

- ***Design***

At this stage, a module "incident attention module" is developed using the following programming languages: PHP with its PHP-LDAP functions [17], [34], and the configuration of the Asterisk DialPlan using the Script language [11], [13], [32], while configuring the IVR options structure [17], [11] in the PBX. Therefore, the developed system was integrated with a text message API; this to send an SMS of the new key to users who need to modify their network key.

In Figure 6, you can see the optimization method based on the integration of Asterisk and AD, whose process begins when the user calls the customer service center and links to the IVR; where the system receives the data (DNI and unlock or change of key operation) and proceeds to carry out the corresponding operations.

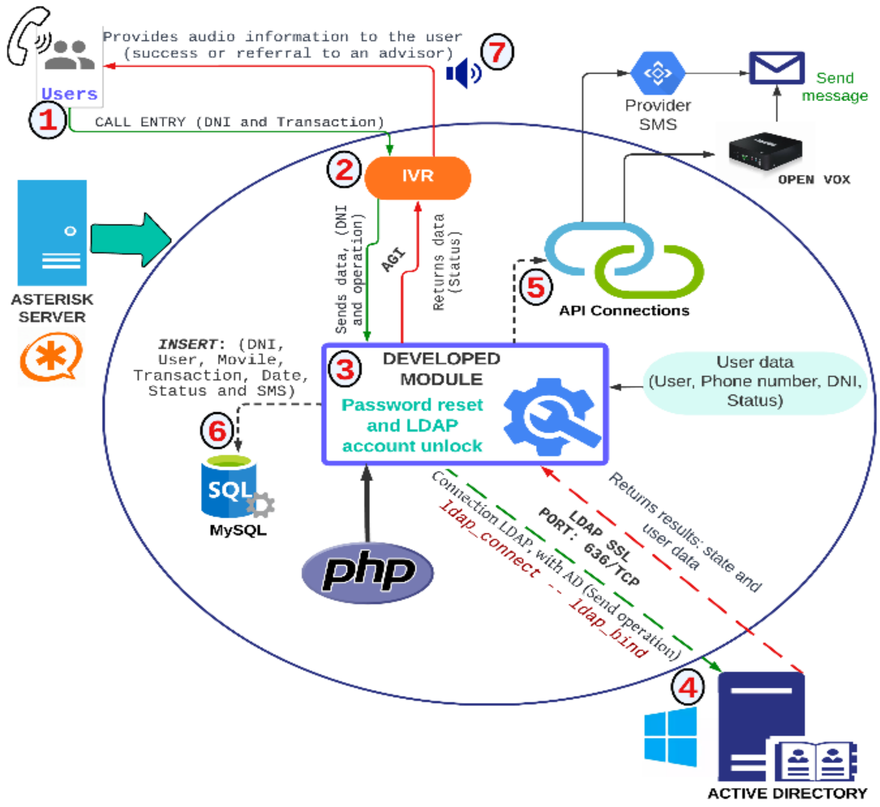

Provides audio information to the user (success or referral to an advisor)
7
Users
1
CALL ENTRY (DNI and Transaction)
2
IVR
Provider SMS
Send message
OPEN VOX
ASTERISK SERVER
Sends data, (DNI and operation)
AGI
Returns data (Status)
5
API Connections
INSERT: (DNI, User, Movile, Transaction, Date, Status and SMS)
3
DEVELOPED MODULE
Password reset and LDAP account unlock
User data (User, Phone number, DNI, Status)
6
SQL
MySQL
php
Connection LDAP, with AD (Send operation)
Ldap_connect -- ldap_bind
LDAP SSL PORT: 636/TCP
Returns results: state and user data
4
ACTIVE DIRECTORY

Figure 6. Optimization method based on the integration of Asterisk and AD

In Figure 7, the flow for the process of the attention of incidents of network users is shown, which restructured the general flow of the IVR of the telephone exchange and the attention of incidents.

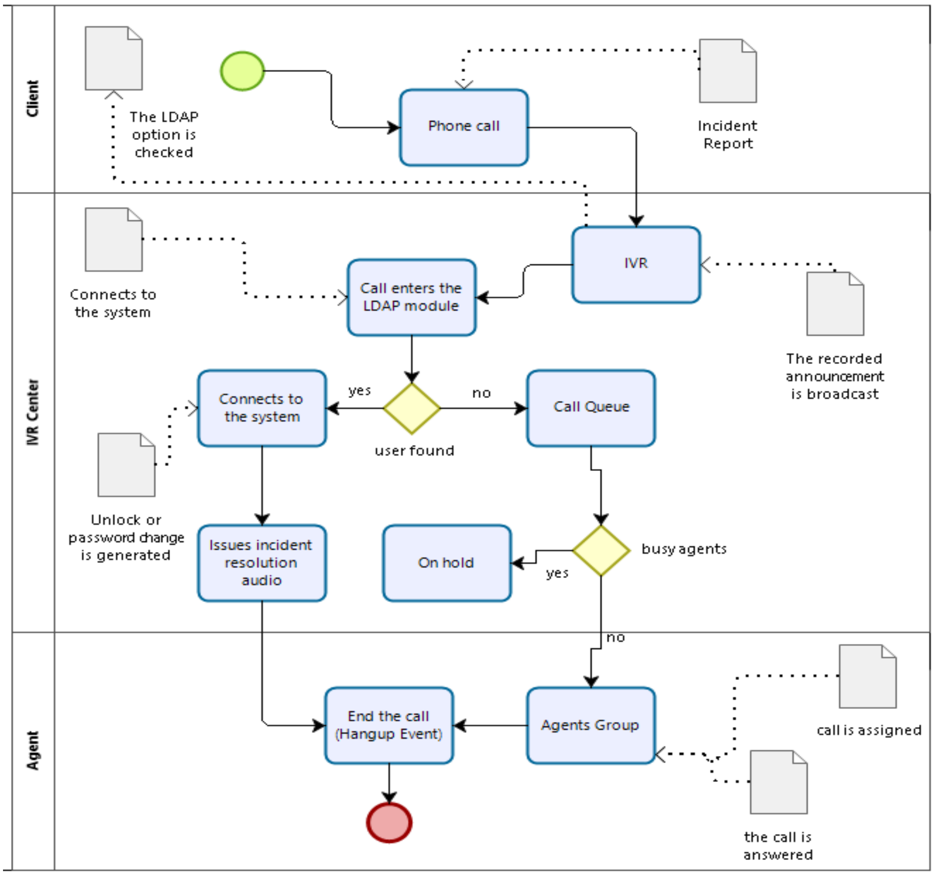


Figure 7. General diagram of the care process for Windows users

Also, in Figure 8, the flow of the new IVR is shown, which allows customers to select the reason for their call, where it is directly linked to the user's requirements and the developed module.

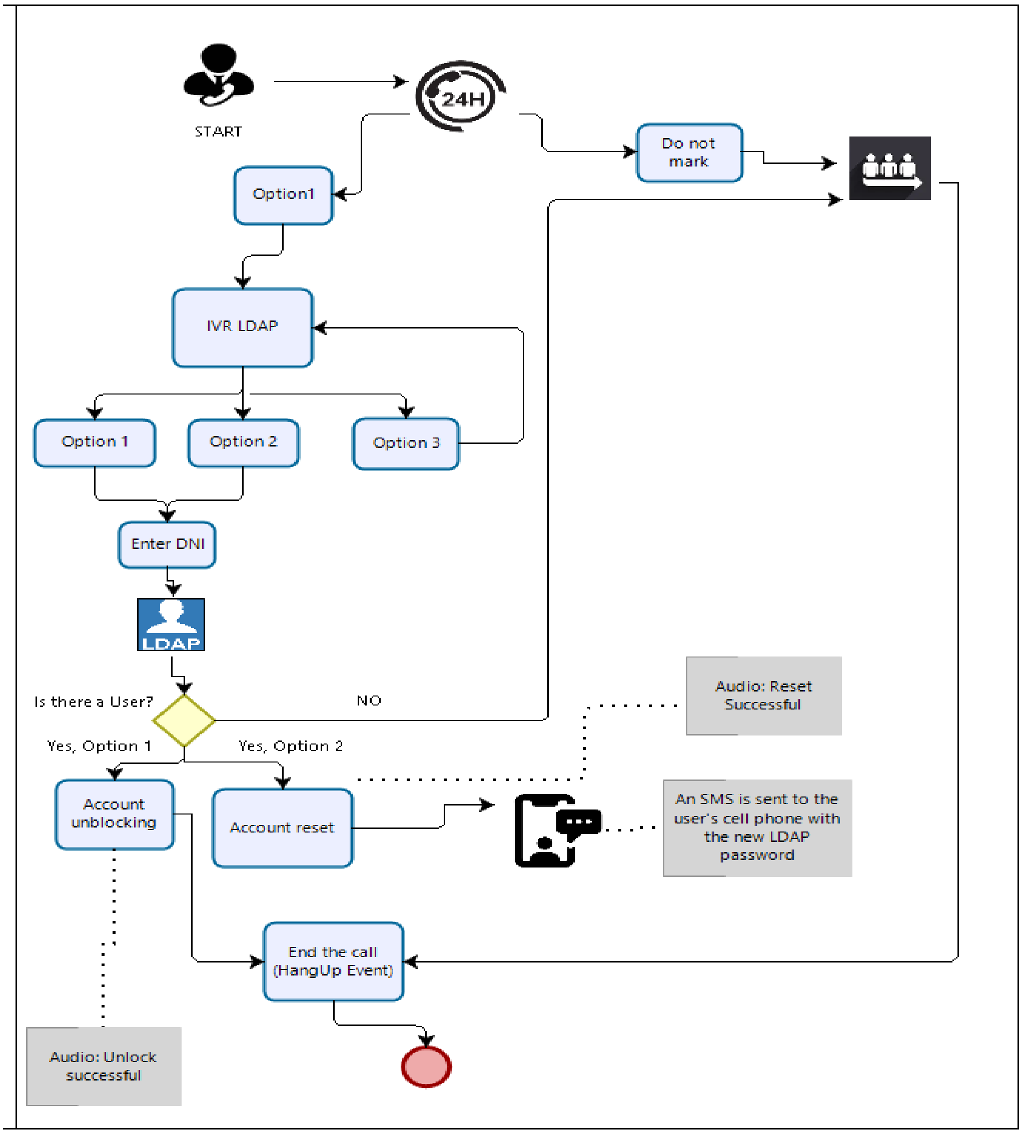


Figure 8. Improved IVR overall flow structure

- ***Coding***

In this substage, an improved flow structure was built for the IVR system to give the customer the ability to interact through selectable options and choose the option according to their requirement, for which Asterisk's BackGround application was employed, which allows the user to listen to the audio and select an option in the IVR; whose syntax used for the BackGround command was:

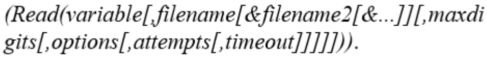
BackGround(filename1[&filename2[&...]][,options[,langoverride[,context]]]).

Table 7 shows the audio format used to play the audio files in Asterisk; which were recorded and uploaded to the server en route /var/lib/asterisk/sounds, so that it can be used on the dial plan [33].

Table 7. Audio format supported in Asterisk

Audio features	
Format	wav
Resolution bits	16 Bits
Sampling rate	8000 Kz
Audio Channels	Mono

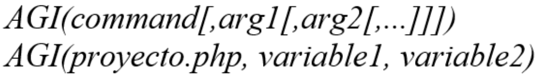
Likewise, we proceeded to capture the data entered in the IVR with respect to type of operation and DNI, using the syntax as shown in Figure 9.



```
(Read(variable[,filename[&filename2[&...]][,maxdi
gits[,options[,attempts[,timeout]]]]])).
```

Figure 9. Asterisk Read application syntax

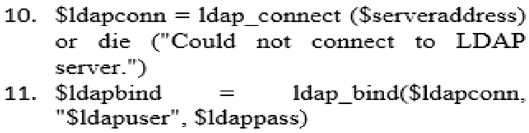
On the other hand, to send the data stored in the variables, from Asterisk to the PHP code of the module, the Asterisk link interface (AGI) option was used, which allowed interacting from the DialPlan with external applications and integrating various systems, as shown in Figure 10.



```
AGI(command[,arg1[,arg2[,...]]])
AGI(proyecto.php, variable1, variable2)
```

Figure 10. Asterisk AGI application syntax

Then, we proceeded to receive the variables sent by AGI from Asterisk to the PHP code, it was done through the $argv variable (Array of arguments passed to an esscript) expressed as follows: $_SERVER['argv'] [1]. Figure 11 shows the code with the PHP functions that were used to connect to the LDAP directory were:



```
10. $ldapconn = ldap_connect ($serveraddress)
    or die ("Could not connect to LDAP
    server.")
11. $ldapbind = ldap_bind($ldapconn,
    "$ldapuser", $ldappass)
```

Figure 11. Code with PHP functions

Likewise, it began to search, validate, and obtain the data of Windows users and assigned them new keys after the change, an algorithm had to be developed, as shown in Table 8 to generate keys in accordance with the security policies that the organization has stipulated.

Table 8. Algorithm built for development

ALGORITHM SEQUENCE

```
serveraddress, dni, contraseña, operación // Initializes LDAP connection values
conexión = ldap_connect(serveraddress ) // Test connection
IF ( conexión = "Exito") THEN { // Validate connection
    acceso = ldap_bind(conexión, userAdmin, contraseña);
    IF ( acceso = "Exito") THEN {
        conex=1; // connection status
        ldap_search(); ldap_first_entry(dni); // User data
        DNI = ldap_get_values("employeeID"); // Get DNI
        celular = ldap_get_values("Mobile"); // Gets cell phone
        IF ( DNI = "" ) THEN {
            DNIusuario = 0; estado = "DNI no existe";
        } ELSE {
            DNIusuario=1;
            IF ( celular = "" ) THEN {
                celular = 0; estado="sin celular";
            } ELSE {
                cuentaldap = ldap_get_values("samaccountname");            // Gets the LDAP account
                nuevaclave = calls_function_Generate_new_key
                newEntry["unicodePwd"] = nuevaclave // performs the password change
                newEntry["lockoutTime"][0]=0; // Unlock the account                IF ( ldap_mod_replace(conexión, dn,
                newEntry) = "Exito" ) THEN {
                    estado="Reseteado";
                    NúmeroCelular=celular;
                    include('algoritmoEnvíaSMS - API');
                    códigoOperación = 1;
                } ELSE {
                    códigoOperación = 0;  estado="Error";
                }
                $resulttext =  ldap_error($ldapconn);
            }
        }
    } ELSE {
        $conex=0;  estado="Sin conexión";
        print_r(ldap_error($ldapconn));       }
} ELSE {
    imprime: "Could not connect to LDAP server."
}
resultado = return array (CódigoOperación, DNIusuario, celular, estado, cuentaldap); // Returns the variables
SQL = "insert into ldap (operacion,DNI,cuenta_user,num_celu,cliente,estado) // Inserts data into the D
fwrite($stdout,"SET VARIABLE dniuser ".resultado[1]." \n");
fwrite($stdout,"SET VARIABLE cod ".resultado[0]." \n");  fwrite($stdout,"SET VARIABLE movil ".resultado[2]."
\n"); // Return the MOVILE NUMBER, DNI Y CODE  found
```

- ***Tests***

In this substage, unit tests were carried out for each requirement and final tests of the developed module, thus validating the correct functioning of the account unlocking and changing network keys. To validate the IVR flow, a softphone [17], [24], [25], [26] was used, which allowed simulating test calls. For failed tests, the code was corrected again; and successful ones are classified as a finished requirement. Thus, it was necessary to perform tests using the following tools as shown in Table 9, this to solve certain problems such as: Failures in the connection with the Active directory, error in the change of password of the accounts and uncompatible characters in the sending of SMS.

Table 9. Tools used for testing

Hardware / Software	Description
Asterisk 13	Telephone system
Centos 7	Operating System
Windows Server 2012 R2, Zentyal	Directories
OPENVOX VSGWM420W	Cellular base, to send SMS
Proxmox VE	Virtualization platform
API (GAMANET)	SMS sending API
Raspberry Dell	Ordenador para virtualizar servidores
Softphone	Zoiper Classic

2.3. Compare

In this phase, the indicators were remeasured to evaluate whether the implementation of the proposed method had improved LDAP incident management. A new data collection of the incident management system was carried out over a period of 5 months. This collection generated a "data post" of 2555 incident records for the analysis of time indicators and 41 records taken randomly for the indicators "Number of incidents queued for attention" and "Percentage of incidents resolved outside working hours". Subsequently, a comparative analysis of the "pre" and "post" data was carried out using bar graphs.

3. Results (Pre and Post Measurement of Indicators)

This section presents the findings of the research, comparing the "Pre" and "Post" results for the following indicators:

I1: Number of incidents in queue for attention.

I2: Percentage of incidents resolved outside working hours.

I3: LDAP incident attention time

I4: Waiting time to be attended to

For ease of understanding, statistical bar charts are used.

3.1. Results of Time Indicators

The results for indicators "I1: LDAP incident handling time" and "I2: Waiting time to be attended" were as follows:

I1: LDAP incident handling time was reduced by 280 seconds. Initially, an average of 316 seconds was required to serve AD users. However, after the implementation of the method, only an average of 36 seconds was required.

I2: The waiting time to be served was completely eliminated. Before the implementation of the method, an average of 350 seconds was required to serve AD users. However, after the implementation of the method, calls do not enter the waiting queue, but are directly served by the system simultaneously. These results can be seen in Figure 12.

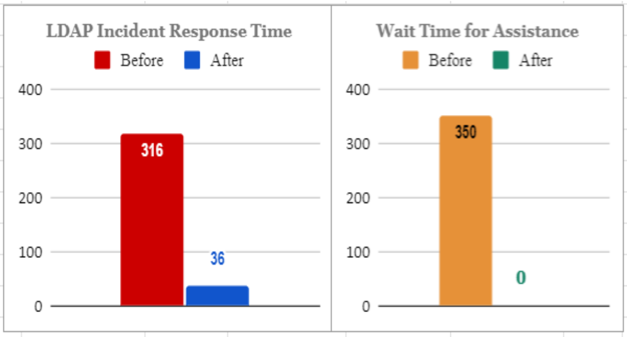


Figure 12. Results of time indicators

3.2. I3: Number of incidents in queue for attention

With regard to the number of incidents attended to daily in the queue by the agents. As shown in Figure 13, the number of incidents attended daily in queue was reduced to 4, compared to the 73 incidents attended daily in queue before the improvement was made, which means that 94.5% of the requests for unblocking accounts and changing network passwords were automated.

On the other hand, 5.5% of AD user incidents are still attended in queue because the user does not have important information such as ID card (DNI) or cell phone number registered in the system.

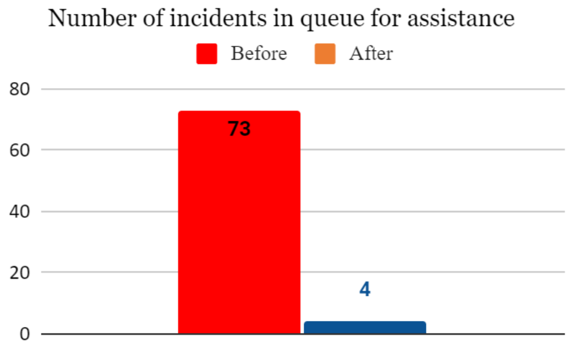


Figure 13: Number of incidents in queue for attention

3.3. I4: Percentage of incidents resolved outside working hours

The automation of the support process has made the service for AD user incidents available 24 hours a day. This means that after-hours incidents can now be attended to in their totality.

The results of the study, comparing "pre" and "post", confirm this. Before the implementation of the method, no after-hours incidents were attended (0.0%). However, after implementation, 99.4% of the incidents reported outside working hours were attended (see Figure 14).

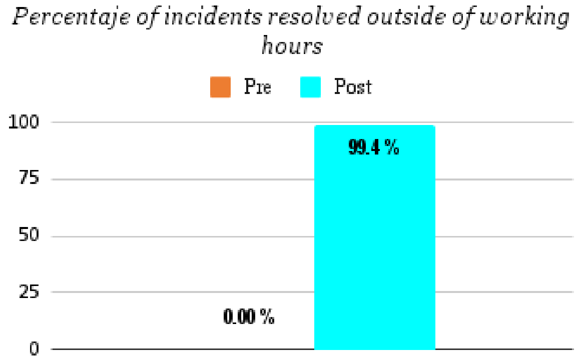


Figure 14. Percentage of incidents resolved outside working hours

4. Conclusion

The proposed method significantly improved the AD user incident management process in all evaluated indicators.

The results showed a considerable decrease in the waiting time for AD user support. With the proposed method, the waiting time was completely eliminated. Additionally, the time to resolve incidents was reduced to 36 seconds.

The workload of AD user support was also considerably reduced. The support that was previously provided by help desk agents is now automated by the developed system. This means that the system is available to handle incidents 24 hours a day, 7 days a week.

The algorithm and solution model developed in this research to automate the incident response process can also be applied in other sectors, such as health centers, municipalities or others, as long as they use an Asterisk telephone exchange and an AD. In addition, the proposed method can be integrated with other technologies or database systems, in order to automate processes or care that require queries of static or dynamic information stored in a database, such as (debts, taxes, medical appointments, etc.). In this way, expenses in care personnel are reduced, since the system carries out the attentions automatically.

The results obtained from the work were focused on a system developed in PHP, which consists of Asterisk 13, Windows server 2012 R2 and LDAP v3. In order to reach the generalization of the results, the implementation in lower and higher versions is recommended, as well as the application in different companies of the same area.

References:

[1]. Zuleta Alemán, L. C. (2021). *Diseño de una propuesta metodológica para la optimización de procesos de gestión de incidentes y requerimientos* [Masters thesis, Universidad EAN] Repositorio EAN.

[2]. Vásquez Muñoz, A. E. & Laguna Molina, K. M.(2016). *Importancia de la implementación de un manual de atención al cliente en la empresa de materiales de construcción y ferreteros CONSTRUNORTE en la ciudad de Estelí en el segundo semestre del año 2015* [Thesis, Universidad Nacional Autónoma de Nicaragua].

[3]. López, Y., & Vázquez, A. (2016). Management Support Services in the life cycle software development. *Revista Cubana de Ciencias Informáticas, 10*(2),46-60.

[4]. Lira Mejía, M. C. (2009). ¿Cómo puedo Mejorar el Servicio al Cliente?: Técnicas para perfeccionar la actitud en el servicio a clientes. México.

[5]. Abualhaj, M. M., Al-Khatib, S. N., Kolhar, M., Munther, A., & Alraba'nah, Y. (2020). Effective Voice Frame Pruning Method to Increase VoIP Call Capacity. *TEM Journal, 9*(1),48-54.

[6]. Santamaria, C. C. (2018). Optimización de tiempos de respuesta y solución de incidentes tecnológicos a través de una mesa de ayuda. *Gerencia Integral de Proyectos,*1-21.

[7]. González Reay, A. X., & Bustos Reina, E. G. (2020). *Formulación acciones de mejora en el servicio de mesa de ayuda de la empresa Computer Consulting GB, aplicando la metodología y buenas prácticas de ITIL 4 e ISO 20000 en la ciudad de Bogotá* [Theis, Universidad Cooperativa de Colombia] Repositorio UCC.

[8]. Avdagić-Golub, E., Begović, M., & Kosovac, A. (2020). Optimization of agent-user matching process using a machine learning algorithms. *TEM Journal, 9*(1), 158-163.

[9]. Ruiz Cañamero, M. A. (2010). *Autenticación y administración centralizada en sistemas VoIP con Asterisk y LDAP* [Master's thesis, Universidad Carlos III de Madrid].

[10]. Khan, M. A., & Shahriar, K. M. (2015). Asterisk Based Open Source IP-PBX System for Accountable Customer Support Service. In *2015 3rd International Symposium on Computational and Business Intelligence (ISCBI),* 85-88. IEEE.

[11]. Hernandez, L., Guzman, H., Ospino, J., Freyle, J., & Pranolo, A. (2019). Design and Implementation of a Marking Strategy to Increase the Contact Ability in the Call Centers Based on Machine Learning. *International Journal on Advanced Science, Engineering and Information Technology, 9*(1), 1-7.

[12]. Binduf, A., Alamoudi, H. O., Balahmar, H., Alshamrani, S., Al-Omar, H., & Nagy, N. (2018). Active directory and related aspects of security. In *2018 21st Saudi Computer Society National Computer Conference (NCC)*, 4474-4479. IEEE.

[13]. Hendrawan, H., & Aditya, B. (2019). Asterisk and Radio Over IP Integration at Voice Communication System Air Traffic Control. In *2019 IEEE 13th International Conference on Telecommunication Systems, Services, and Applications (TSSA)*, 271-276. IEEE.

[14]. Gladkikh, T., Korobova, L., Chernyaeva, S., Tolstova, I., & Pracheva, E. (2022). Telemarketing automation based on the MIKO IP-telephony module. In *AIP Conference Proceedings*, *2647*(1). AIP Publishing.

[15]. Lopez Baraldini, L., & Casanova, G. (2016). *Automatización Active Directory/LDAP* [Thesis, Universidad Argentina de la Empresa].

[16]. Xiang, D., & Sun, L. H. (2011). The application of asterisk-based IP-PBX system in the enterprise. In *Electrical Power Systems and Computers: Selected Papers from the 2011 International Conference on Electric and Electronics (EEIC 2011) in Nanchang, China on June 20–22, 2011, 3*, 753-757. Springer.

[17]. Ávila Alvarado, Á. J. (2016). *Sistema de información de telefonía IP para la asistencia de cartera, utilizando librerías AGI (Asterisk Gateway Interface)* [Bachelor's thesis, Universidad del Azuay].

[18]. Palka, J., & Motyl, I. (2010). Use of active directory in securing the client applications. *Annals of DAAAM & Proceedings,* 407-409.

[19]. Arias, F. G. (2012). *El proyecto de investigació*n (6th ed.). Fidias G. Arias Odón.

[20]. Cordero, Z. R. (2009). La investigación aplicada: una forma de conocer las realidades con evidencia científica. *Revista educación*, *33*(1), 155-165.

[21]. Guffante Naranjo, T., Guffante Naranjo, F., & Chávez Hernández, P. (2016). *Investigación Científica-El Proyecto de Investigación*. Enero.

[22]. Meléndez Valladarez, S. M., Gaitan, M. E., & Pérez Reyes, N. N. (2016). *Metodología ágil de desarrollo de software Programación Extrema* [Thesis, Universidad Nacional Autónoma de Nicaragua].

[23]. Cadavid, A. N., Martínez, J. D. F., & Vélez, J. M. (2013). Revisión de metodologías ágiles para el desarrollo de software. *Prospectiva*, *11*(2), 30-39.

[24]. Khan, B. M., Fahad, M., Bilal, R., & Khan, A. H. (2022). Performance Analysis of Raspberry Pi 3 IP PBX Based on Asterisk. *Electronics*, *11*(20), 3313.

[25]. Tabsombat, S., Pimpuch, N., Hiranya-ekaparb, A., Raksapatcharawong, M., Yamaoka, K., Phatrapornnant, T., ... & Duangtanoo, P. (2010). Radio over IP prototyping: A communication system for emergency response. In *2010 7th International Conference on Service Systems and Service Management*, 1-5. IEEE.

[26]. Hidayat, R., Lestari, N. S., Sujana, A., & Ramady, G. D. (2019). Optimizing Branch Telephone Networks for Campus VoIP with Mobile Clients. *Journal of Physics: Conference Series, 1175*(1), 012061.

[27]. De Clercq, J. (2011). *How-To: Use LDAP Over SSL to Lock Down AD Traffic*. ItPro Today. Retrieved from: https://www.itprotoday.com/windows-78/how-use-ldap-over-ssl-lock-down-ad-traffic [accessed: 19 October 2023].

[28]. Wu, Z. Y., Huang, W. P., & Yu, L. (2014). Design and implementation of unified identity authentication system based on LDAP in digital campus. *Advanced Materials Research*, *912*, 1213-1217.

[29]. Goranov, G., & Hristova, R. (2019). An Approach for Integrating Joomla with LDAP. In *2019 II International Conference on High Technology for Sustainable Development (HiTech)*, 1-4. IEEE.

[30]. Andjarwirawan, J., Palit, H. N., & Salim, J. C. (2017). Linux PAM to LDAP authentication migration. In *2017 International Conference on Soft Computing, Intelligent System and Information Technology (ICSIIT)*, 155-159. IEEE.

[31]. Tang, J., Xiao, C. S., & Wu, J. P. (2014). Permission centralized control system based on LDAP and module. *Applied Mechanics and Materials*, *494*, 1262-1265.

[32]. Khan, S., & Sadiq, N. (2017, March). Design and configuration of VoIP based PBX using asterisk server and OPNET platform. In *2017 International Electrical Engineering Congress (iEECON)*, 1-4. IEEE.

[33]. Goel, S., & Bhattacharya, M. (2010). Speech based dialog query system over asterisk pbx server. In *2010 2nd International Conference on Signal Processing Systems*, *3*. IEEE.

[34]. Viteri, J. T. M., Valero, M. I. G., & León, A. R. E. (2016). Gestión de Usuarios Con LDAP (Lightweight Directory Access Protocol) para el Acceso a los Servicios Tecnológicos ya la Información en las Empresas. *Journal of Science and Research*, *1*(1) 10-15.