ISSN: 2594 - 2905

AUTOMATION OF THE PROCESS FOR THE MANAGEMENT OF THE TECHNICAL INFORMATION OF THE PROJECTS IN PWO DE MEXICO, USING POWERSHELL OF .NET TECHNOLOGY.

AUTOMATION OF THE PROCESS FOR THE MANAGEMENT OF THE TECHNICAL INFORMATION OF THE PROJECTS IN PWO OF MEXICO, THROUGH THE USE OF POWERSHELL OF .NET TECHNOLOGY

[1]M.C. Violeta Martínez Ramírez, [2]M.I. Alejandro Gil Vázquez,
[3]M.C. Juan Gonzalo Alarcón Xicotencatl, [4]Ing. Ángel de Jesús Servin Chávez

[1,4]Instituto Tecnológico de Puebla
Tecnológico Nacional de México
Department of Systems and Computer
Science
Av. Instituto Tecnológico #420, colonia Maravillas. Puebla, Puebla
[1]violeta.martinez@itpuebla.edu.mx , [(4)]i15221977.19@puebla.tecnm.mx

[2]Instituto Tecnológico de Tláhuac
Tecnológico Nacional de México Systems
and Computing Department
Estanislao Ramírez Ave. #301 Col Selena, Tláhuac, Mexico City
[2]ing.gil@ittlahuac.edu.mx
[3]Instituto Tecnológico de Puebla
Tecnológico Nacional de México
Electrical and Electronics Department
Av. Instituto Tecnológico #420, colonia Maravillas. Puebla, Puebla
[3]gonzalo.alarcon.x@gmail.com

Summary - The constant growth of companies, as well as the increase in demand for the products they offer, has led them to increase their processes and need a more accurate control of them, which have required automating certain processes that do not need to be controlled by humans, as these can save time and reduce the workload of some employees.

Through the use of the agile methodology Scrum, the development of the automation of a production process of the company PWO de México was carried out.
S.A. de C.V. This was done through iterations (Sprints) in short times (1 to 4 weeks) during which deliveries were made to customers, thus ensuring a reliable development and according to the tastes of these. The results obtained with this project were a saving of almost 98% in the time that an employee invests in the realization of this process.

Keywords: Automation, process, powershell, directories, access permissions.

Abstract -- The constant growth of companies, as well as the increase in demand for the products they offer, has led them to increase their processes and thus need a more exact control of them, with which they have required to automate certain processes They do not need to be controlled by humans, as these can save time and reduce the workload of some employees.

Using the agile Scrum methodology, the development of automation was carried out in a production process of the company PWO de México S.A. de C.V. Which was carried out through iterations (Sprints) in short times (from 1 to 4 weeks) during which deliveries were made to customers, in this way a reliable development was ensured according to their tastes. The results obtained with this project were obtained a saving of almost 98% in the time that an employee invests in carrying out this process.

Key words - Automation, process, powershell, directories, access permissions.

INTRODUCTION

Due to the increasing demand for work, the optimization of production processes in companies seeks an optimization of these, sometimes, automation is the answer to their needs, due to considerable time savings in the tasks and minimization of personnel interaction in them, avoiding the

generation of errors due to erroneous data capture.

This research work provides a detailed description of the development process carried out at PWO de México S.A. de C.V. to automate the generation of the directory structure of part numbers manufactured by the plant in Mexico, as well as the assignment of access permissions to different work groups in the directories obtained.

PROBLEM STATEMENT

The creation and control of the storage structure, as well as the assignment of access permissions in the directories of new part numbers created in PWO de México S.A. de C.V., are activities that have been performed manually, although there are scripts that create directories and assign access permissions to the groups already established by the company, these activities must be repeated every time a new part number is established, which results as a tedious and repetitive task for the person or persons in charge.

Another parameter to consider is the time of use. Being a reactive process, the time of the activities will be directly proportional to the time required to obtain the updated part number structure, this task is not performed sequentially, but every elapsed interval, delaying the display times and, consequently, the production process.

As a result of all the above problems, the creation of directories and the control of the data storage structure is slow and, therefore, the structure cannot be visualized quickly, substantially affecting the areas involved in the production process that make use of them.

by not having a standard location for the storage of data required for design work and/or measurement of parts.

GENERAL OBJECTIVE

Implement an information system that controls the digital storage structure of projects generated in the production process of the company.

PWO de México S.A. de C.V., using the .NET Technology PowerShell command line interface.

SPECIFIC OBJECTIVES

- Analyze the information y requirements.
- Create a recursive folder structure based on the storage format preset.
- Set permissions using Microsoft's Access Control Lists (ACLs) on folders and folders files.
- Download files from a specific network drive that hosts required data by the company located at the head office in Germany (unit N:) via FTP and set the read permissions.
- Software testing in scheduled periods.

JUSTIFICATION

Due to a high demand in the creation of new part numbers produced by various areas involved in the company's production process, a storage structure for CAD data and part measurement programming must be in place. This structure must be updated, simulating a synchronization between existing data in the ERP and the data stored for design or measurement purposes. Since the data is not directly related to an ERP , the company has created a specific structure to ensure data security by using permissions assignment technology based on Access Control List (ACL) and Active Directory (AD).

Excessive time spent on creating directories, controlling the data storage structure and assigning access controls when creating new directories.

The automation of these activities is required in order to optimize the visualization of documents by the departments involved in the production process, as well as to encourage the people involved to focus on new projects and/or tasks. In this sense, the present work shows the implementation of a system for the control the storage structure.

digital through the use of .NET technology, making use of the PowerShell console interface. The contribution of this work will be manifested in the benefits resulting from the improvement actions: the optimization of the directory process and the activities that this entails. Likewise, the delays that have a direct impact on the document consultation times will be reduced, the activities will be carried out uniformly, in less time and in a more practical way.

THEORETICAL FRAMEWORK

PWO (Progress-Werk Oberkirch AG) is a supplier to the global automotive industry for the development and production of advanced metal components and subsystems using lightweight construction. The Group has developed unique know-how in metal forming and joining in the course of its 100-year history since its founding in 1919. With its expertise in cost-effective lightweight construction, it contributes to environmentally friendly driving and longer distances [1].

Windows PowerShell is a console interface (CLI) with the possibility writing and joining commands by means of scripts. It is much richer and more interactive than its predecessors, from DOS to Windows 7. This console interface is designed for use by system administrators to automate tasks or perform them in a more controlled manner. Originally named MONAD in 2003, its official name was changed to the current one when it was released to the public on April 25, 2006.
The standard used by PowerShell are Microsoft Office 365 cmdlets for Windows PowerShell to perform various administrative tasks from the command line. All Office 365 cmdlets are listed and described by common administrative task such as the

user control and domain administration. [2].

Windows PowerShell introduces the concept of the cmdlet ("command-let"). A cmdlet is a simple, single-function command-line tool included in the shell, whose purpose is to manipulate objects. You can recognize cmdlets by the format of their name: a verb and a noun.

separated by a hyphen (-), for example, Get-Help and Get-State. The verbs express specific Windows PowerShell actions, the names describe specific types of objects.

PowerShell cmdlets may have both mandatory and optional parameters. If a mandatory parameter is missing, the system will prompt you to enter it. If an optional parameter is missing, PowerShell will use the default value [3].

The .NET platform is an environment used to develop applications and run them on WIN32 operating systems. The development of applications on this platform does not depend on any language, that is to say, it can be developed in a multilingual way.

The .NET framework is the core part of the .NET platform, it is an infrastructure used to integrate and run Microsoft Windows-based applications. It is mainly composed of a set of compilers for various languages, a data library and a virtual machine [4].

Access control lists (ACLs) perform packet filtering to control the movement of packets across a network. Packet filtering provides security by limiting traffic access to a network, restricting user and device access to a network, and preventing traffic from leaving the network. IP access lists reduce the possibility of spoofing and denial-of-service attacks, and allow dynamic and temporary user access through a firewall [5].

Scrum is a process in which a set of best practices are applied on a regular basis to work collaboratively, as a team, and obtain the best possible result from a project. These practices support each other and their selection is based on a study of the way highly productive teams work.

It could be said that SCRUM is based on a certain controlled chaos, but it establishes certain mechanisms to control this indeterminacy, manipulate the unpredictable and control flexibility.

In Scrum, regular partial deliveries of the final product are made, prioritized by the benefit they bring to the project recipient. For this reason, Scrum is especially suitable for projects in

complex environments, where results need to be obtained soon, where requirements are changing or poorly defined, where innovation, competitiveness, flexibility and productivity are essential [6].

STATE OF THE ART

Olmeda de la Casa's work implements a base image cloning system with the operating system (Windows 10 Enterprise 64 bits) together with the software and utilities applied in the computer classrooms of a university department for the realization of teaching practices. In this way, an identical configuration is available in the classrooms, without detriment of computer equipment with different hardware, facilitating the teaching of independent practices of the assigned classroom. By means of the proposed solution, it is possible to carry out efficient and simple maintenance of the computer equipment in the event of future software updates, probable increases in the number of computers in the classrooms or hardware or software failures.

A series of scripts have been developed using the Powershell tool for the automation of routine tasks periodically executed in the administration of computers that make up the computer classrooms. This facilitates the management of operating systems in a simple and safe way, minimizing possible errors as opposed to the tedious repetition of tasks of choosing maintenance equipment by equipment. In addition, it takes advantage of the fact that it is on computers with an active directory. -AD- configured with group settings in policies (GPOs) to complement and facilitate the system layout of the hardware [7].

Huera Paltan presents a research on Microsoft PowerShell generated scripts and graphical management to administer Windows Server.

2008, allows selecting the optimal management in the Microsoft academy of the Escuela Superior Politécnica de Chimborazo.
The comparative analysis is based on the scientific method, with techniques of direct observation and experimentation, through the implementation of software tools: Powershell 2.0, PowerGUI and system monitor.

The administration methodology under investigation, were integrated in Powershell and graphical mode, to automate server systems, allowing to determine which has better performance, usability and portability, giving as a quantitative result a difference of 59% indicators for graphical management vs. 66% for PowerShell management [8].

DEVELOPMENT OF THE RESEARCH

The development of the project was carried out using the agile Scrum methodology, so different iterations were carried out in short time blocks (1 to 4 weeks) and at the end of each block a preview of the project was delivered to the client who proposed to add, remove or modify certain features to the project.

Figure 1 shows a scheme that represents the Scrum methodology applied to the project; however, in subsequent paragraphs each developed stage is described.

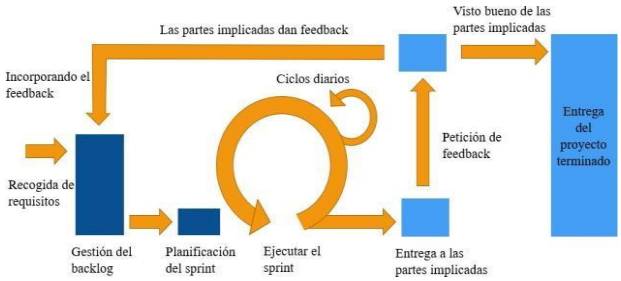


Figure 1. Agile Scrum Methodology Diagram Source: Own elaboration.

Gathering of information and requirements (meeting with the work team)

The client introduces the members of the project team, recording the prioritized requirements. The objective to be followed for the development of an information system that automates the creation and control of the

The company also established part number directories for each component within the departments involved in the company's production process, as well as the deliverables at the end of the first Sprint.

Execution of iterations (Sprints).

Considering the scrum methodology, the diagram in figure 2 shows the objectives achieved by each sprint executed in the development of the project.

Figure 2. Development of the agile Scrum methodology with its Sprints and results.
Source: Own elaboration.

File restructuring and content extraction

For the first sprint of the project development the objective achieved was the restructuring of complementary files in the process, which are essential to execute functionalities in the project, an example of these is a file named "Acl_config.ini" (figure 3), which will be read by the software to obtain accurate data on groups and access permissions.

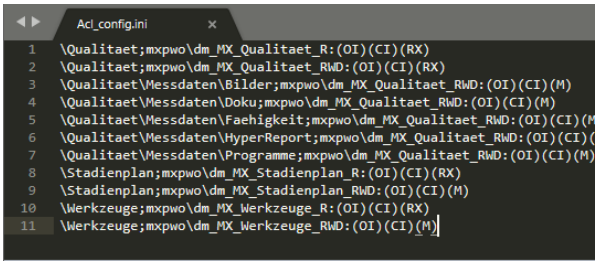


Figure 3. Structure of the "Acl_config.ini" document
Source: Own elaboration.

We also worked with the extraction of information from the documents on which the execution of the software is performed, these documents contain important information, such as the part number of the component to be created, the plant on which it is created and additional information. An example of this document is shown in figure 4.

Figure 4. .erl file structure Source: PWO documentation.

Creating directories and assigning permissions

In the second Sprint, the corrections and new requirements provided in the previous review were made, and the process of creating directories, internal structure and assigning access permissions to the different groups that exist in the company's production process was developed. Table 1 shows the internal structure that each part number must contain when created.

Root	Level 1	Level 2	Level 3
P04141	Qualitaet	Messdaten	Bilder
			Doku
			Faehigkeit
			HyperReport
			Program
	Stadienplan		
	Werkzeuge		

Table 1. Internal structure of a part number directory
Source: Own elaboration,

Once the directory structure was created, access permissions were assigned to the different groups involved in the production process. Figure 5 shows a fragment of code written for the assignment of ACL permissions; in it, four sentences can be observed.

The first one is in charge of obtaining the ACL information included by default in the directory.

In the second statement, a new ACL is created where the group of users that will have access, the type of access it has and the "Allow" command to approve the assignment are specified. In the third statement, the ACL information obtained from the first statement of the previously created permission is added to the ACL information.
Finally, the last line stores information generated from the previous sentences, specifying the path to the directory on which the permissions will be applied and the object that includes the information.

ISSN: 2594 - 2905

```
#Getting default Acl from directory
$Acl = Get-Acl "$routepn\$partnumber"
#Set default group in directory and access type
$Ar = New-Object System.Security.AccessControl.FileSystemAccessRule($groupAD, "ReadAndExecute", "
$Acl.SetAccessRule($Ar)
#Applying access to directory
Set-Acl -Path "$routepn\$partnumber" -AclObject $Acl
```

Figure 5. Code fragment for ACL permissions assignment.
Source: Own elaboration.

Execution of validations

In the third iteration of the project development, the creation of validations was addressed, which are a fundamental part of the optimal functioning of the software due to their risk reduction at the moment in which the data enters the information processing through the code and will ensure the accuracy and reliability of the data that enters. Some of the most important validations are:

- Verify that the part number is of type 'P'.
- Verify that the plant on which the part number was created belongs to Puebla (plant 5000).
- Verify length of part numbers.

The internal nomenclature of the part numbers indicates that, if they are longer than 6 digits, it means that in the document from which they are extracted they contain a space, being replaced by a hyphen under '_' at the time of creating the directory.

Code optimization and validations

The fourth Sprint in the development of the project is aimed at optimizing the code, identifying all possible cases in which errors and/or exceptions may arise, thus managing to control them and prevent the process from stopping. It is of utmost importance to identify these cases in time,

The system is designed to handle them and avoid the creation of "junk" files that do not comply with the quality standards initially specified.

Table 2 shows two examples of exception handling at two important stages of the process: the creation of directories and the assignment of ACL permissions to them.

Process	Exception	Message
Creation of directorio	System.IO.IOException	"Directory already exists"
	Others	"an error occurred"
Assignment of permits	System.Security	"you may not have enough permissions"
	Others	"an error occurred"

Table 2. Internal structure of a part number directory
Source: Own elaboration.

Completed project

The development of the fifth iteration is oriented towards local testing, which will lead to testing on client servers and run-time error correction.

They start software testing locally, creating dummy groups on the test computer, modifying some of the essential files and running the software to test its effectiveness.

One of the files edited to carry out the testing phase locally was the document "Acl_config.ini" (Figure 6), in which the groups belonging to the company are replaced by the fictitious groups created in the equipment.

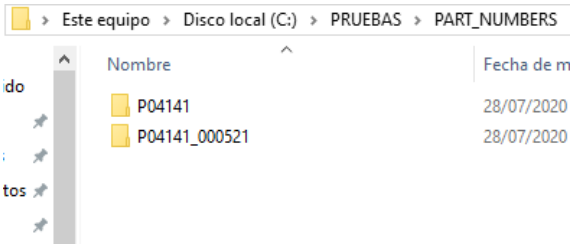
Acl_configp.ini

```
\Qualitaet;DESKTOP-G94N9IJ\grupo_prueba_rx:(OI)(CI)(RX)
\Qualitaet;DESKTOP-G94N9IJ\grupo_prueba_rx:(OI)(CI)(RX)
\Qualitaet\Messdaten\Bilder;DESKTOP-G94N9IJ\grupo_prueba_m:(OI)(CI)(M)
\Qualitaet\Messdaten\Doku;DESKTOP-G94N9IJ\grupo_prueba_m:(OI)(CI)(M)
\Qualitaet\Messdaten\Faehigkeit;DESKTOP-G94N9IJ\grupo_prueba_m:(OI)(CI)(M)
\Qualitaet\Messdaten\HyperReport;DESKTOP-G94N9IJ\grupo_prueba_m:(OI)(CI)(M)
\Qualitaet\Messdaten\Programme;DESKTOP-G94N9IJ\grupo_prueba_m:(OI)(CI)(M)
\Stadienplan;DESKTOP-G94N9IJ\grupo_prueba_rx:(OI)(CI)(RX)
\Stadienplan;DESKTOP-G94N9IJ\grupo_prueba_m:(OI)(CI)(M)
\Werkzeuge;DESKTOP-G94N9IJ\grupo_prueba_rx:(OI)(CI)(RX)
\Werkzeuge;DESKTOP-G94N9IJ\grupo_prueba_m:(OI)(CI)(M)
```

Figure 6. Structure of the "Acl_config.ini" file for tests
Source: Own elaboration.

Once the documents were edited for the corresponding tests, the satisfactory results derived from the local simulations were approved to continue with the testing phase on the client's servers.

Figure 7 presents two examples of part numbers generated by the software, each with a different format from the two possible for generated part numbers.



Part numbers created in the testing phase Source: Own elaboration.

Figure 8 shows the access permissions to the main directory of the part number that is selected from the "test_group_1" group assigned by software.

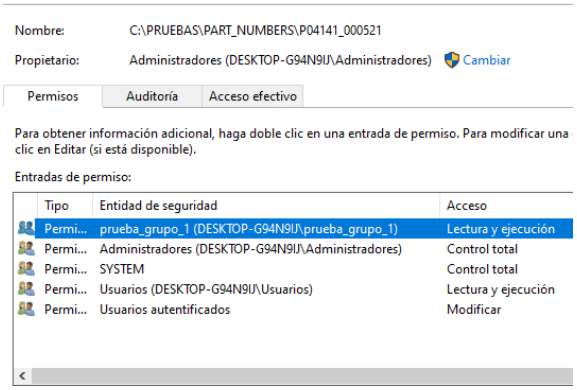


Figure 8. Dummy group assigned in the root folder of a part number
Source. Own elaboration.

Figure 9 shows the "Qualitaet" directory, set as internal within part number structure, the group "test_group_rx" is assigned to it with its access permissions.

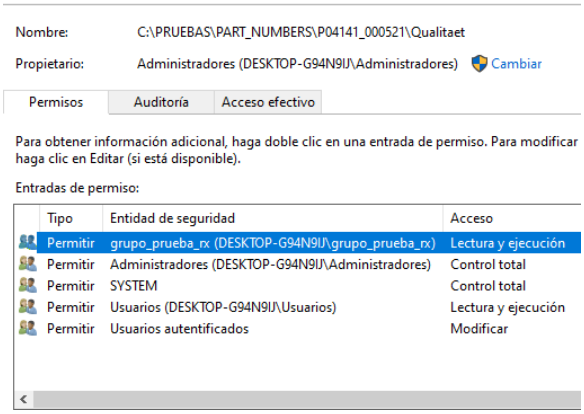


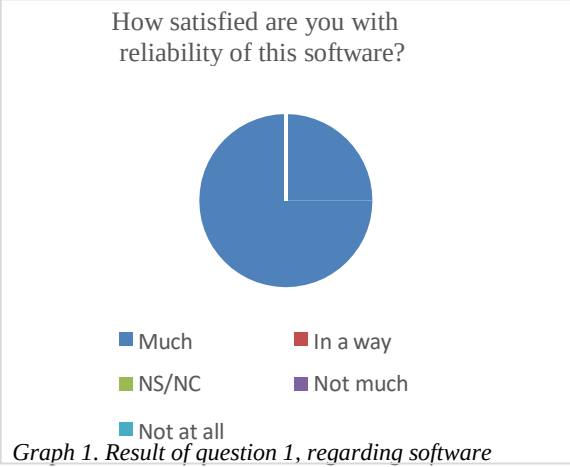
Figure 9. Dummy group assigned in the Qualitaet folder of a part number
Source: Own elaboration.

DISCUSSION AND ANALYSIS OF RESULTS

Once the software was executed in the company's servers, it achieved the initially expected objectives and was executed in real time in production, complying with each and every one of the requirements requested by the company. Finally, a satisfaction survey was applied to the Organization and Information Technologies (OI) department team to evaluate the functionality of the software Each item follows the Likert scale model, which contemplates 5 possible answers that represent the degree of satisfaction in each indicator (these answers are represented by colors in the graphs shown in graphs 1, 2, 3 , 4, 5, 6 and 7), in the department work 3 people, thus making up 100% of the data collection in each question, each member represents 33% of each answer in the research instrument. Analysis of the results:

Question 1: This question is focused on the Reliability dimension of the software performance, for example, the correction of bugs that arose during development and are now resolved.

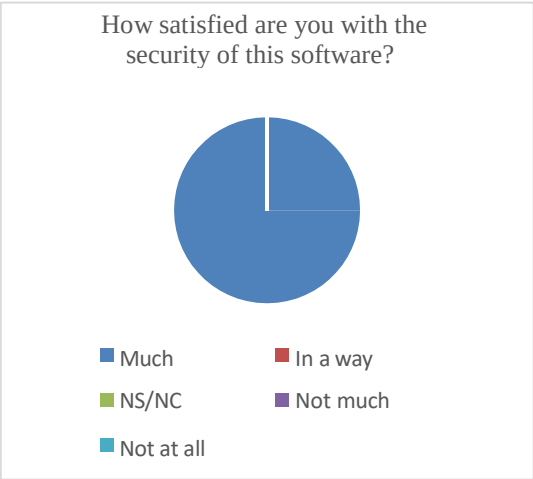
IPSUMTEC
ISSN: 2594 - 2905



Graph 1. Result of question 1, regarding software reliability.
Source: Own elaboration.

100% of the respondents are satisfied with the reliability of the software, due to the tests performed beforehand, and the corrections made during these tests.

Question 2: Question number 2 of the survey, dimension Customer safety with handling exceptions that may arise during the execution of the software in production.

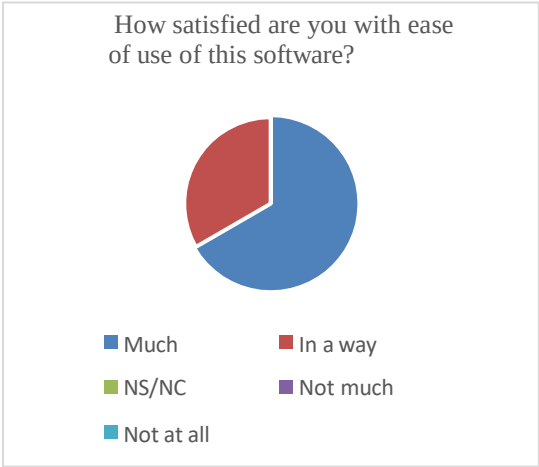


Graph 2. Result of question 2, regarding software security
Source: Own elaboration.

The security of the software is supported by its programming, including all the cases in which it could be insecure, thus avoiding problems that could occur during its use.

operation. This is how all respondents agree that the software is secure.

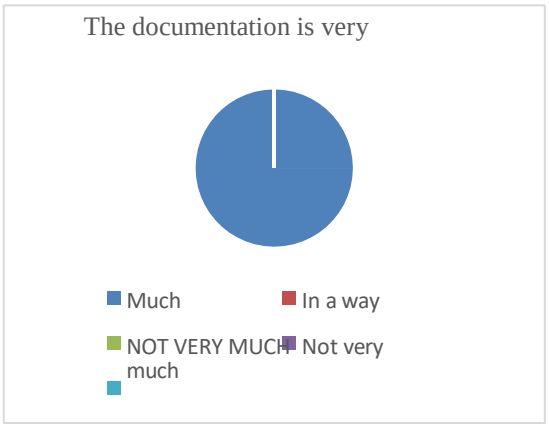
Question 3: The use of the software is automated, through a script that is executed from time to time, however, it can also be executed manually, for this it requires a previous knowledge of the programming language used.



Graph 3. Result of question 3, respondents' satisfaction with the ease of use of the software.
Source: Own elaboration.

In this item, 67% of the respondents agree on a simple interaction with the software. Since it is an automation, involvement with the code is unnecessary. The reason why 33% are satisfied is because it is not necessary to have knowledge of the language, unless changes or deep maintenance is required, prior knowledge of the programming language is necessary.

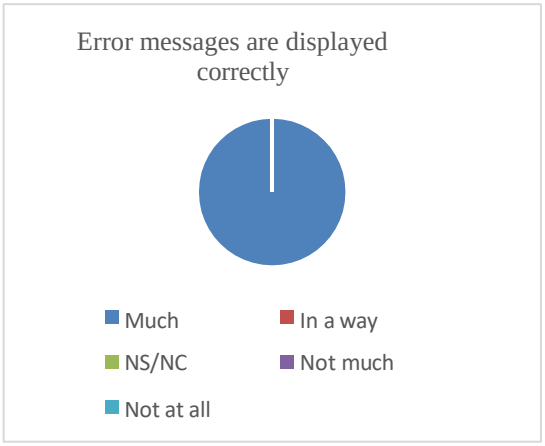
Question 4: Documentation is as important as the software itself, as it is the basis for bug fixes and future maintenance, delivered as a user manual.



Graph 4. Result of question 4, on documentation delivered to the company.
Source: Own elaboration.

Due to the extensive documentation provided to the OI team in the user's manual on how the software works and how to use it, the survey yielded a 100% satisfaction rate from the respondents.

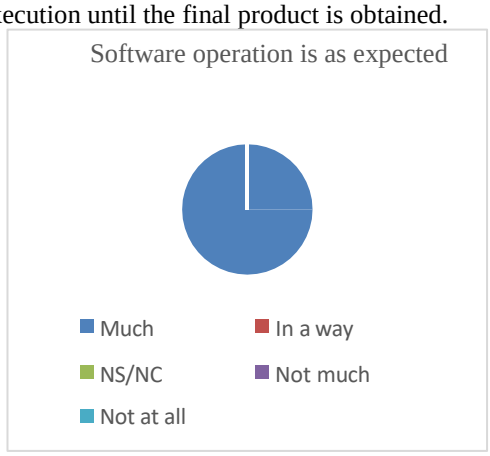
Question 5: Exceptions are runtime errors that occur when a process is not performed correctly, which can slow down the execution of the software, when they appear, they must be handled and the user must be notified that this has occurred.



Graph 5. Result of question 5, error with exception handling
Source: Own elaboration.

100% of respondents are satisfied with the representation of error messages when they appear.

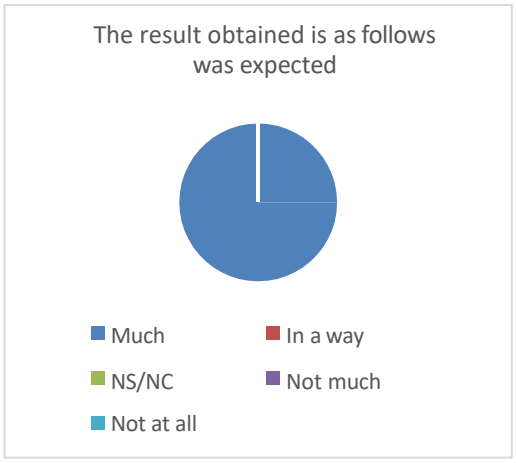
Question 6: When the software is executed, different tasks are performed, all of which contain necessary validations to be able to continue with execution until the final product is obtained.



Graph 6. Result of question 6, software operation.
Source: Own elaboration.

100% of the OI team agrees that the operation of the software works as initially designed.

Question 7: The result of the software is a directory structure with access permissions to different groups of the company involved in the production process.



Graph 7. Result of question 7, results obtained with the execution of the software.
Source: Own elaboration.

According to the results obtained in the first executions of the software in the production phase, the respondents converge in the fact that the projected results were obtained.

The survey data indicate that the performance of the software meets the requirements requested by the customers through the results obtained in each item of the applied research instrument.

CONCLUSIONS

From the results obtained in the implementation of the software, it is concluded that the operation meets the requirements initially raised by the customer, its operation and implementation have marked a significant and considerable time savings to OI staff, thanks to the process of creating directories and part number structure demanded a time of approximately 40 minutes, with the implementation of the software, the time to create the structure is below 60 seconds, thus achieving a savings of 97.5%, it should be considered that the times vary depending on the number of new documents to be processed.

It is mentioning that savings time belongs to the time invested by OI personnel, therefore, other departments of the company are indirectly benefited from the savings. The creation process was triggered at the request of OI, which meant a state of urgency to have the structure and initiate a process, originating from the production area or some other under a vital corporate symbiosis. With the implementation of the software, it has been possible to avoid the repetition of this process, since it is executed in pre-established periods of time, so that when a new structure has to be created, the process is executed.

In summary, the implementation of the project in PWO S.A. de C.V. represents a reduction in the creation and waiting times, directly to the members of OI department and, as a consequence, to the rest of the departments involved in the production process, improving the efficiency of the production processes.

REFERENCES

[1] Progress-Werk Oberkirch AG (2020). *WHO WE ARE & WHERE WE ARE LOCATED*. Retrieved from https://www.progress-werk.de/en/investors-press/about-pwo/

[2] Wilson, E. (2008). *Windows Powershell Scripting Guide.* Washington, USA: Microsoft Press.

[3] CA ARCserve (2012). *PowerShell command guide.* Iceland, New York: CA Technologies. Retrieved from https://documentation.arcserve.com/Arcserve-RHA/Available/R16/ENP/Bookshelf_Files/PDF/XO_POWER_W_ESN.pdf

[4] Ball, D. (2005). *Migration of PEMEX debt information system software applications to .NET technology.* Mexico City: Universidad Nacional Autónoma de México. Retrieved from http://132.248.9.195/ptd2005/00327/0348907/Index.html

[5] Cisco Systems, Inc (2015). *Security Configuration Guide: Access Control Lists, Cisco IOS XE Release 3S.* San Jose: Cisco Systems, Inc. retrieved from. from https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/sec/sec_data_acl/configuration/xe-3s/sec- data-acl-xe-3s-book.pdf

[6] Maida, E. G., & Pacienzia, J. (2015). *Software development methodologies.* Buenos Aires: Catholic University of Argentina. Retrieved from http://bibliotecadigital.uca.edu.ar/repositorio/tesis/methodologies-software-development.pdf

[7] Olmeda de la Casa, F. M. (2019). *Deployment of operating system images in departmental classrooms with OpenGnsys and subsequent process automation using Powershell.* Escuela superiror de ingeniería informática. Valencia, Spain: Universidad Politécnica de Valencia. Retrieved from https://riunet.upv.es/bitstream/handle/10251/127678/Olmeda%20-%20Despliegue%20de%20im%c3%a1genes%20del%20sistema%20operativo%20en%20aulas%20departamentales%20con%20OpenGnsys%20y%20posterior%20automatizaci%c3%b3n%20de%20procesos%20mediante%20Powershell.pdf

[8] Huera Paltán, B. P. (2014). *Script generation with Microsoft Powershell for Microsoft server administration of Microsoft espoch academy Microsoft espoch.* Faculty of computer science and electronics. Riobamba, Ecuador: Escuela superior politécnica de chimborazo. Retrieved from http://dspace.espoch.edu.ec/bitstream/123456789/3316/1/18T00543.pdf