Back Office Automation Tools using Visual Basic for Applications in Microsoft Excel

Federico Brotons Gutiérrez
Gerard Garriga Turró
Pau Gallardo Campos
Pau Gómez-Rico Pérez

Tutor: Ernest Solé Udina
EME05
2018/2019
# Table of contents

1) Abstract  
2) Executive summary  
3) Contextualization  
   3.1) Macro perspective  
   3.2) Micro perspective  
4) The program  
   4.1) Delivery note and Invoice  
      4.1.1) Generate delivery note  
      4.1.2) Modify delivery note  
      4.1.3) Remove delivery note  
      4.1.4) Generate invoice  
   4.2) Mailing  
5) Quantitative analysis  
   5.1) Quantifying the value created by the program  
      5.1.1) Delivery note  
      5.1.2) Invoice  
      5.1.3) Mailing  
   5.2) Total savings  
   5.4) Conclusions and implications for SISNET 2003 S.L.  
6) Market competition  
7) Critical discussion  
8) Conclusion  
9) Annex  
10) Glossary  
11) Bibliography
1) Abstract
Large companies are starting to treat automation as a high-priority initiative to be implemented during the upcoming years. Yet, while great strides have been made in applying workplace technologies, it is argued that small businesses face more difficulties than big firms when it comes to a change in their operating system, as well as to exploit the returns to scale. This paper aims to demonstrate that Small and Medium Enterprises can also benefit from automation. Using Visual Basic for Applications, we created a program to automate certain tasks of an existent firm. In this way, we provide evidence that smaller firms could take advantage of a simple and feasible automation of processes.

Keywords
Automation of processes
Small and Medium Enterprises (SMEs)
Back Office Jobs
Visual Basic for Applications (VBA)
Cost optimization
Returns to scale
2) Executive summary

In the last century, automation has penetrated many sectors. Overall, workplace technologies have been designed to save labour from mechanical, repetitive or musculature-needed labour. Furthermore, it has revealed an opportunity to increase the firm’s value by reshaping the equilibrium between individuals, processes and technology. Following this line of inquiry, we believe that in our selected firm, automation would replace part of the repetitive tasks made by employees in such a way they could readapt their activities into a higher customer-oriented task, increasing the total value of the firm itself. That is, this transformation could contribute to substantial benefits for the firm and the economy as a whole, as well as in terms of satisfaction and quality of life of employees. To prove these claims, we displayed our analysis focusing specifically on a small cleaning services company, SISNET 2003 S.L, drawing on the close relationship we enjoy with its Chief Executive Officer, who has provided us with all the requested information.

Thus, the high potential benefits of automation has highly motivated our analysis. In this way, we will first proceed in section 3 by presenting the impact of automation on employment and employees’ quality of life (i.e. from a macro perspective) while then we will evaluate the benefits of automating certain processes from a firm’s profit point of view (i.e. from a micro perspective). At this stage, in section 4 and 5 we provide evidence of the firm efficiency gains- in terms of cost optimization- by developing a tool via Visual Basic for Applications (VBA) using Excel, and implementing it to our company, for the purpose of automating Back Office jobs. To the best of our knowledge, VBA Excel is the simplest and most appropriate tool for automating simple tasks that is within our grasp. In this regard, we decided to elaborate an explanatory video to present the functioning and scope of our program in a clearer and more appealing way.

As a final note, it is worth noting that we did not pretend to create a tool to be implemented on a widespread scale, but rather to demonstrate that Small and Medium Enterprises (SMEs) can also take advantage of automation by means of investing in technology that enables them to gain in productivity at an affordable price. We will further discuss this issue in Section 7.

---

1 Brotons, F., Gallardo, P., Garriga, G., Gómez, P. (Youtube), 2019. Delivery notes, invoices and mailing through VBA-Excel. [online] Available at: https://www.youtube.com/watch?v=TVInnX9pyhM
3) Contextualization

3.1) Macro perspective

Despite the success of technology as labour saviour, automation has not necessarily carried a reduction in aggregate employment. This fact is explained due to the existence of complementarities between human labour and technology in many jobs. For instance, machinery may replace repetitive processes, but still faces a lack of key values such as creativity or intuitive judgement. This is reflected in Polanyi's paradox, stating that the human knowledge of how the world functions is, in fact, beyond our explicit understanding, in such a way we cannot transmit this knowledge to a computer and therefore automate the task entirely. That is, jobs consist of a set of tasks and not all of these tasks can be automated easily or simply automated.

Workforce changes, for instance, are actually designing new and less repetitive or lack of judgement jobs, rather than merely eliminating occupations. In fact, through July 2018, manufacturing jobs over the past year grew at the fastest pace since 1995, adding 327,000 jobs. Besides, as the O-ring model points out, whenever one part of the production chain becomes more productive, in our case, due to technology, it implies an increase in the value of human labour. For instance, as suggested in an article from UBS, automation will allow us to save our employees from mundane tasks in a way they could focus on more creative and higher-added value services. In this regard, the largest documented impact derived from automation at work has been a decrease of the working hours as well as an improvement of the employees quality of life. In a nutshell, the human team could benefit from the time saved by employing it in what it does best; think strategically, act creatively and interact in a human way.

Yet, regarding the future effects of automation, a specific set of actions from different parties are said to be crucial to embrace the challenges that automation will provide us. Thus, in order to take full advantage of automation, a more precise analysis would require considering the expected actions from governments, private-sector leaders and innovators, which goes beyond the scope of our investigation. In this manner, we will base our study on assessing the gains of automation from the perspective of a concrete firm.

---


3.2) Micro perspective

The value of automation resides mainly in the efficiency it creates. This implies, amongst others, a reduction of costs, time saving and an increase of the quality of the product and/or service. While these benefits are well-known, it is unclear whether a company would be willing to assume the high-implied costs of implementing new technologies into the business process, especially if we refer to small and medium-sized enterprises (SMEs), which usually have a tighter budget and cannot profit from the economies of scale.

However, we found out that even the simple automation of processes - which presents lower costs - allows to eliminate errors, reduce biases and speed-up the transactions. These basic technologies have shown that, in comparison with human performance, it is possible to obtain a cost reduction up to 75% in repetitive tasks\(^6\). In this way, it is of our interest to see which tasks could require a simple automation. By doing so, the American Productivity and Quality Center (APQC) publishes a list of nearly 1,100 intersectoral activities that make up 300 basic business processes. These processes are organized into 70 process groups and into 13 high-level process categories. Using this framework, we could examine the average effort for each activity to identify which processes are more prone to be automated. Looking at the table, we see that the categories that reside at the top of the figure are actually the ones with the highest transactional component, such as the tasks that underpin the management of financial resources, the management of customer services and the delivery of physical products. On the other hand, the processes that are less easily automated are the most strategic ones, focusing on valuations, on activities such as the improvement of the strategic vision, as well as the management of external resources\(^7\).

At this stage, we proceed by creating a tool such that it permits automating certain processes - the most easily automated- of a specific firm- SMEs-, in order to identify the effects of automation. That is, we want to increase the radius of action of the automation of processes - beyond big firms and advanced or intelligent automation - by demonstrating that simple automation can lead to significant gains to SMEs. What is more, we want to do it by creating a simple tool which does not require either a high-level knowledge or a costly investment.


<table>
<thead>
<tr>
<th>Processes more easily automated</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process accounts payable and expenses</td>
<td>62</td>
</tr>
<tr>
<td>Process payrolls</td>
<td>56</td>
</tr>
<tr>
<td>Provide world-wide commercial services</td>
<td>53</td>
</tr>
<tr>
<td>Carry out de revenue accounting</td>
<td>52</td>
</tr>
<tr>
<td>Manage client service contracts</td>
<td>52</td>
</tr>
<tr>
<td>Manage the withdrawal of products and audits</td>
<td>52</td>
</tr>
<tr>
<td>Evaluate the service and the customer satisfaction</td>
<td>50</td>
</tr>
<tr>
<td>Produce, manufacture and deliver the product</td>
<td>50</td>
</tr>
<tr>
<td>Manage the logistics and storage</td>
<td>48</td>
</tr>
<tr>
<td>Reward and retain the employees</td>
<td>47</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Processes less easily automated</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sell assets</td>
<td>15</td>
</tr>
<tr>
<td>Implement solutions of communications technology</td>
<td>15</td>
</tr>
<tr>
<td>Develop knowledge management functions</td>
<td>16</td>
</tr>
<tr>
<td>Provide/support the communications technology services</td>
<td>16</td>
</tr>
<tr>
<td>Manage the relationships with the employees</td>
<td>17</td>
</tr>
<tr>
<td>Improve business resilience</td>
<td>17</td>
</tr>
<tr>
<td>Develop a client service strategy</td>
<td>18</td>
</tr>
<tr>
<td>Generate and define new products/services ideas</td>
<td>18</td>
</tr>
<tr>
<td>Redistribute and retire the employees</td>
<td>19</td>
</tr>
<tr>
<td>Establish governance strategies for the provision of services</td>
<td>19</td>
</tr>
</tbody>
</table>

Source: IBM Institute for Business Value – La evolución de la automatización de procesos

First, we considered to be a good feature the fact that VBA is the programming language of Excel and other office programs. Excel is, in fact, presented as the most important computer program in workplaces around the world which includes more than 750 million users worldwide\(^8\).\(^9\).

As discussed above, nowadays many sophisticated technologies can already deal with immense complexity. However, since the application of them requires having abundant resources, adapting these new processes ends up being economically unfeasible for SMEs. Nevertheless, these enterprises have actually business processes of lesser complexity\(^10\), so that a standard spreadsheet software could be a potentially beneficial and a feasible alternative.

---


Visual Basic for Applications (VBA) is such an add-on tool that has become popular in all fields of business\textsuperscript{11,12}. Compared to the other information system software or database software, VBA is not only more economical and thus more suitable for SMEs\textsuperscript{13}, but also can be used to address the usual problems a SME faces when dealing with automation; misunderstanding, complexity and low-user friendliness, reaching a state of invalid data\textsuperscript{14}. Furthermore, it has been proved that it can lead to an increase of the efficiency of the budgeting process\textsuperscript{15}. That is to say, VBA Excel can be the solution for SMEs to gain budgeting efficiency through automation in a more economical and simplistic way. In this manner, we apply VBA Excel to our SME, Sisnet 2003 S.L, to provide evidence of the firm’s efficiency gains by simple automation.

4) The program

At this point, our purpose is to improve, via automation, those small-medium firms that currently do not have or ignore the possible gains that they could obtain implementing automation on some of their daily routine tasks. Moreover, our radius of action would be firms which are established in Catalonia, due to territorial closeness. This fact explains why the most part of our program is in Spanish.

The program is composed by 4 sheets “Delivery Notes and Invoices”, “Mailing”, “Customers” and “Invoice History”\textsuperscript{16}, though the last two ones are simply databases which are protected to prevent other users from accidentally or deliberately, change or delete data. On the other hand, the other ones are mainly user interface, namely, where the tasks are certainly done.

In a nutshell, the core of the program is composed by 2 sections which are:

1. Delivery note & Invoice
2. Mailing

\textsuperscript{13} Hesse, R. and Hesse Scerno, D,. 2009. How electronic spreadsheets changed the world. Interfaces, 39 (2), 159-167.
\textsuperscript{16} “Delivery note and invoices” - Albaranes y facturas -, “Mailing” - Envio mail -, “Customer” - Clientes -, “Invoice History” - Historial de facturas -. 
4.1) Delivery note and Invoice

The sheet called “Delivery notes and Invoices” is used for managing the database entry as well as the issue of documents, both delivery notes and invoices. In the upper left part of the sheet, there are 4 command buttons called “Generate delivery note”, “Modify delivery note”, “Remove delivery note” and “Generate invoice”, each one associated to a specific macro, a piece of the program.

4.1.1) Generate delivery note

If we want to create a delivery note, the following userform will appear, which is divided into 4 areas:

In the upper left part, we find the client’s data, both the completion and the issue date and the delivery note number. The customer is selected using a drop-down list, which is connected to the sheet “Customers”, where it is found the customer database. The delivery note number always follows the patron: current month - year - xxxx. For instance, the number assigned to the first delivery note issued on June of 2019 would be: 620190001, and the following one 620190002.

Secondly, it is found the information related to the service offered, where mainly the type of it and the price charged are specified. There are two combo boxes loaded with information about the duration - whether it is punctual, daily, weekly, biweekly, monthly... - and another about the tasks made (home maintenance, window cleaning, community cleaning, office cleaning). Moreover, we also have textboxes where we can add additional information related to the specific service offered.

17 ”Generate delivery note” - Generar albarán -, “Modify delivery note” - Modificar albarán -, “Remove delivery note” - Eliminar albarán - and ”Generate invoice” - Generar albarán -.
Then, we decide the amount of hours per month and the price charged to the client. The price written is only an initial approximation and later on it can certainly be modified through the command button "modify delivery note". Since there is no equal service and client, there exists important variations in prices. This explains why we do not set a specific price for each type of the services offered, though it could be carried out in other types of companies with more fixed and less negotiable quotas.

At the top right, we add a non-compulsory section, just in case of the firm includes some product to the customer, regardless of the service offered, such as air freshener, industrial paper or degreaser. Again, there exists a combobox for selecting the desired product and a textbox where additional information can be added.

Finally, the firm can add some comments, which will not appear in the delivery note sent to the customer. It is a very useful tool for the firm, in case they needed to remark some characteristic of the customer with whom they are working with -i.e., risk of non-payment-.

Once the delivery note is finished, the user must click on the accept button, and automatically a Word or a Pdf document will be fulfilled with the data written before and saved on the client’s folder of the computer.

4.1.2) Modify delivery note
The second command button found in the “Delivery note and Invoices” sheet, is called “Modify delivery note”, which is used for modifying delivery notes. If it is needed to change some input, the following userform will appear:
Clicking on the drop-down list button “Delivery note number”\(^\text{18}\), it appears all the delivery notes’ numbers, previously created. Thus, we choose the delivery note that we want to modify and automatically all the information related to the client will appear. Then, we will simply have to change what we consider appropriate, tap the accept button and the corresponding entry will be updated, both in the Excel database and in the Word or the Pdf document stored previously in the client’s folder.

The modification of the delivery note through Excel itself is blocked. Delivery notes can only be modified accessing to the command button of "modify delivery note". In this way, we ensure that the possible information changed is also being updated in the delivery note Word document.

4.1.3) Remove delivery note
The third command button found in “Delivery notes and Invoices” sheet, is called "Remove delivery note". It allows the user to break up the selected delivery note in the active cell of the sheet. By putting us anywhere in the line corresponding to a specific delivery note number and clicking on the undo button, a message box will pop up asking if we are sure we want to delete this entry line. After clicking on accept, the delivery note line will disappear, as well as the Word or the Pdf document stored previously in the client’s folder. The message box mentioned that will appear is the following:

\[\text{¿Estás seguro que quieres eliminar este albarán?}\]

\[\text{Sí} \quad \text{No}\]

---

\(^\text{18}\) “Delivery note number” - número de albarán -.

\(^\text{19}\) Are you sure that you want to remove the selected delivery note?
4.1.4) Generate invoice

Finally, the fourth command button, found in the top left of the “Delivery notes and Invoices” sheet, is called "Generate invoice". Generally, this task will be undertaken at the end of the month, when the delivery notes need to be converted into invoices. After clicking on this button, it appears a userform which is shown below.

In this userform, it is listed all the delivery notes which are pending to be converted into invoices. This tool has been programmed to pick all the delivery notes at the same time, though the user can select as many delivery notes as he needs, either clicking on the interested clients’ entry or using the search button located at the top of the user form.

When all the interested delivery notes have been selected, the user has to click on the “Bill” button; simultaneously, all the selected entries will move from “Delivery note and Invoices” sheet to “Historical invoices” sheet where all the generated invoices are stored. At the same time, a Word or Pdf document for each client will be fulfilled using the database of the client and individually saved on the computer in the respective client’s folder.

---

20 “Bill” button - Botón de facturar
21 “Historical invoices” - Historial de facturas
4.2) Mailing

The sheet called “Mailing” is used for choosing the client to whom we want to send a specific mail, with the option to attach as many documents as we want. Once we select the client, a mail is automatically created with the personal data related to it.

![Excel sheet with columns for client, address, and documents]

Above it is shown the “Mailing” sheet. Notice that there are two types of Excel cells; on the one hand, cells with a white background - independent ones, which can be modified by the user- and on the other hand, ones with a grey background - dependent ones, which are protected and provide information linked to the “Customers” database sheet-. In this way, depending on which customer we select, the grey background cells will be different.

When we click on the command button located in the upper left - Send Mail - we proceed to run the code, consisting of connecting Excel to Outlook, which will cause that Outlook will be automatically opened. Afterwards, the user will be able to either confirm the sending of the mail and/or -when necessary- add or modify some aspects.

Through the code and using Hypertext Markup Language (HTML), we create a message body with some underline or bold text, bullet points to list the documents attached as well as the company’s logo.

---

22 Send Mail - Enviar Mail
Moreover, several types of email templates have also been created, with the aim of providing to the user a wide range of email options. For instance, to celebrate “The international day of cleaning” or the “Sant Jordi’s day”.
5) Quantitative analysis

As presented above, our main purpose is to offer a program to those small and medium-sized enterprises (SMEs) that usually face significant budget constraints and have, to this day, some tasks that could be automated. Due to some economic and legal restrictions, we did not aim to create a complete and transferable program for firms or invite them to incorporate VBA Excel by themselves. Yet, we decided that the best way for presenting ourselves is as if we were Excel-specialists consultants.

Thus, in this section we focus on quantifying economically the effect of our program on firms. To do this, we applied our tool to the firm Sisnet 2003 S.L., since it meets the requirement of being an SME with a restricted budget, as well as being a company very close to us. Knowing the firm has enabled us to estimate more precisely the optimal price that could be charged for offering them our tool. More precisely, to determine the optimal price, we decided to carry out two studios; on the one hand, studying the reduction in labour costs that our product could cause and on the other hand, analyzing the market and the possible competitors.

5.1) Quantifying the value created by the program

In this first analysis, it is of our interest to estimate how much resources - in the form of salaries, effort and time - both the firm and the workers invest in performing those potentially automated tasks.

In this way, we looked at the collective labour agreement for the workplace to know which is the salary that a worker should receive. We decide to use the “Convenio colectivo oficinas y despachos Catalunya 2017-2018”23, since the possible firms with whom we could interact (i.e., willing to require our service) are found in this area. Considering the low-capabilities that employees are required to have to complete these tasks - elaboration of delivery notes and invoices and mailing tasks - it is reasonable to think that an adequate annual salary would be around 18,000€24, having and extra cost for the firm of 540€ (social security - 30% of the total amount). Consequently, assuming that the worker works 40 hours per week, the firm would have to afford, per employee, a cost of 9.66€ per hour.

The next step is to observe the benefits of our program, relating the above-mentioned salary with the time-saving that it would produce. As it is said, the program enables to speed up the three-step procedure in a given Back Office Job: delivery note, invoice and mailing.

---

23 “Catalunya office group 2017-2018”
5.1.1) Delivery note

In this case, we consider the task of automating the delivery notes the firm issues, which we considered as being one of the tasks more prone to be automated by our tool. To compute the gains in efficiency in terms of reduction of labour costs, we have considered our firm Sisnet 2003 S.L., plus two hypothetical firms, named A and B, with a higher amount of invoicing, which are able to produce 20% and 50% more, respectively, than Sisnet 2003 S.L.

By automating certain tasks that are done by employees, our program fosters efficiently the reduction of time, which is captured by the throughput. Thus, in order to compute this time saving, it is fundamental to know how much time employees spend on these tasks. Since we apply our program to Sisnet 2003 S.L., it was considered that the most precise way to know how much time workers need to elaborate a delivery note is to directly observe them. Taking advantage of being close to the firm, during May of 2019 we met the company. By doing so, we found that the throughput is around 5 minutes, in other words, with the current methodology, the employee needs, on average, 5 minutes to fill out a delivery note.

Even though we did not have the time to set up the application in the firm, we have measured how would be the average throughput if a given employee deals with this type of task using the tool. Thanks to the program, the throughput is reduced from 5 to 1 minute. Before, the worker needed to use a special sheet and then fulfill, by hand, all the customer details - ID, name, street, etc. After the implementation of the program, this process is done automatically, using a drop-down list so that the user has access to the client's database. Once the customer is selected, everything is fulfilled, thus reducing to zero a hypothetical human error.

The following table shows the total output of delivery notes, which, on average, each company receive per month, the gross salary per hour earned by the employee on Sisnet 2003 S.L., the old throughput without using the platform, the new throughput using the platform and the hypothetical savings that automation would carry to the firm.

<table>
<thead>
<tr>
<th>Firm</th>
<th>Output (u/mth)</th>
<th>Salary (€/h)</th>
<th>Old Throughput (min/u)</th>
<th>New Throughput (min/u)</th>
<th>Total savings – Delivery notes (€/mth)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sisnet</td>
<td>150</td>
<td>9.66</td>
<td>5</td>
<td>1</td>
<td>TS = 120.75 - 24.15 = 96.60</td>
</tr>
<tr>
<td>Firm A</td>
<td>180</td>
<td>9.66</td>
<td>5</td>
<td>1</td>
<td>TS = 144.90 - 28.98 = 115.92</td>
</tr>
<tr>
<td>Firm B</td>
<td>225</td>
<td>9.66</td>
<td>5</td>
<td>1</td>
<td>TS = 181.13 - 36.23 = 144.91</td>
</tr>
</tbody>
</table>
It is important to emphasize that in this particular step of the process the unit benefit created by the tool is constant, regardless of the volume of output the company has. This can be easily demonstrated computing the average savings per unit in the three companies, which in all cases remains constant at 0.644 € per unit\textsuperscript{25}. In other words, if the program is successfully implemented, for each delivery note issued by the company its costs will be reduced by 0.64€.

\subsection*{5.1.2) Invoice}

The invoices are ordinarily generated at the end of the month by our program, which is able to create and save them automatically in one easy click. Once the customer has agreed on the conditions written in the delivery note, the invoice can be done. Yet, on average, 2% of the total amount of delivery notes are not successfully converted into invoice.

Again, in order to determine the impact of our program in terms of labour cost reduction, we applied our tool to Sisnet 2003 S.L. To this day, a firm’s worker needs approximately 9h each month to deal with the 150 delivery notes, which means a throughput of 3.67 minutes for each invoice done. At the present time, the company is using a specific program for elaborating them.

As displayed in the video, due to the program it is possible to create several invoices in a matter of seconds (the program’s rapidity will also depend on the laptop’s processor). In the table below, we have considered that the total amount of time needed to perform the 147 invoices is 20 seconds. In that time, the program creates the whole invoices and saves them to each client’s folder. Since the throughput using the program is 0.13 seconds\textsuperscript{26} per unit- practically zero- we have considered the total cost of using the tool as being redundant.

\begin{table}[h]
\centering
\begin{tabular}{|c|ccc|c|c|}
\hline
Firm & Output (u/mth) & Salary (€/h) & Old Throughput (min/u) & New Throughput (sec/u) & Total savings - Invoice (€/mth) \\
\hline
Sisnet & 147 & 9.66 & 3.67 & 0.13 & TS = 86.94 - 0* = 86.94 \\
Firm A & 176 & 9.66 & 3.67 & 0.13 & TS = 105.09 - 0* = 104.09 \\
Firm B & 220 & 9.66 & 3.67 & 0.13 & TS = 130.11 - 0* = 130.11 \\
\hline
\end{tabular}
\end{table}

\* The total costs using the program is approximately zero. Notice that as output increases the total costs are nearer to zero.

\textsuperscript{25} Total savings divided the output is equal to the unit benefit, i.e. 96.60/150 = 115.92/180 = 0.644€ per delivery note.

\textsuperscript{26} New throughput: 20 seconds / 147 invoices = 0.13 sec/u
Different from the previous procedure, this one generates more value as the total output increases, a phenomenon widely known as returns to scale. Although it is not observed in the above table, since the unit benefit is in all scenarios 0.59€ per invoice, we can proudly state that the impact of our program would mitigate the costs that appear once the company’s clients’ portfolio increases (i.e administrative costs).

In other words, the program can perform using the same time -just a few seconds- either creating one invoice or hundreds of them.

5.1.3) Mailing

Mailing service is the core of the program. During last years, customers increasingly ask for a better and closer service, in such a way we consider this issue of relevance to be taken into account when the program was made. Both when the delivery note and invoice are sent, the program automatically offers a personalized email where the documents are attached depending on the client’s name.

Following the literature, we realized that workers spend too much time writing emails. For instance, a research made by HubSpot\textsuperscript{28} -a developer of software- showed that marketers spend an average of 3.48 hours a week creating and sending emails. Another report, done by Smartsheet\textsuperscript{29}-a software as a service application- states that over 40% of workers spend at least a quarter of their time, doing manual repetitive tasks that involves mailing, data collection and data entry, amongst others. That is to say -presented in the report-, that if those repetitive tasks were automated, almost 60% of workers could invest 6 hours per week focusing on other higher-added value for the firm.

Before analysing the results, it is important to clarify some issues, presented as follows. Firstly, the tool has been set up to deal one by one client -though it is possible, through a loop, to create and send with just one click hundreds of emails to clients-. Moreover, although the program contains different types of emails (i.e. Sant Jordi greeting)- which can be easily switched through the platform- in the cost analysis we will only focus on the documents attachment case, which retain the highest value-added amongst the email types for the company.

\textsuperscript{27} Total savings (TS) / Output = Unit benefit -i.e. 86.94 €/147u = 0.59€/u -. 
Even though setting up the application in the firm remained beyond our reach, we measured how would be the average throughput if a given employee dealt with this type of task using the tool. A Sisnet 2003 S.L's employee spends firstly, on average, 11 minutes to send the delivery note. Once it has been signed and received, it is spent, on average, 7 minutes to send his respective invoice.

The time spent in each type of mail have been found following the next rationale. In order to send a delivery note, it is assumed a worker needs 2 minutes for finding the paper and 1 minute for checking the data written. Then, 2 minutes for scanning, 1 minute for saving the delivery note made by hand and 30 seconds more for saving the digital one. Finally, it is needed 30 seconds for opening the mail and 4 extra minutes for writing and attaching the document. Regarding the invoices, the reasoning has been the same, yet the worker would avoid the first 4 steps (finding the paper version, checking, scanning and saving the documents), while he would need 30 extra seconds for finding the digital one. We must remember that before the implementation of our program, the company was already using an alternative program for doing the invoices so that they were already saved in their computers.

Thanks to the literature and after the assumptions that are made, it is noted that mailing is one of the most time-consuming tasks. To this day, Sisnet 2003 S.L. sends around 150 mails per month for all the delivery notes issued, and later, once the order is accepted, around 147 mails with the invoice attached.

In order to determine the time spent on this task it is necessary to compute the weighted average, since the number of delivery notes is always bigger than invoices.

\[
Weighted\ average_{\text{Sisnet}} = \frac{150}{147 + 150} \times 11\, \text{min} + \frac{147}{147 + 150} \times 7\, \text{min} = 9.02
\]

With this reasoning, we obtain that the time saving would be around 8.02 minutes per order. In the following table, it has been resumed the impact of our tool in terms of total savings.

<table>
<thead>
<tr>
<th>Firm</th>
<th>Output (u/mth)</th>
<th>Salary (€/h)</th>
<th>Old Throughput (min/u)</th>
<th>New Throughput (min/u)</th>
<th>Total savings - Mailing (€/mth)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sisnet</td>
<td>297</td>
<td>9.66</td>
<td>9.02</td>
<td>1</td>
<td>TS = 431.43 - 47.82 = 383.61</td>
</tr>
<tr>
<td>Firm A</td>
<td>356</td>
<td>9.66</td>
<td>9.02</td>
<td>1</td>
<td>TS = 517.13 - 57.32 = 459.82</td>
</tr>
<tr>
<td>Firm B</td>
<td>445</td>
<td>9.66</td>
<td>9.02</td>
<td>1</td>
<td>TS = 642.42 - 71.65 = 574.77</td>
</tr>
</tbody>
</table>
It is in this step where the reduction of time is more meaningful. The reason lies in the length of the service, by far the longest among all the services that the program provides. While without using the software, the Sisnet’s employee lasts, on average, nine minutes to deal with a client. In contrast, by employing the program the throughput is significantly reduced, since to cope with that task it is only necessary one minute per client. However, as with delivery notes task, there is no empirical evidence that the program generates returns to scale. In other words, no matter the size of the company, the unit benefit is constant, 1.29€\(^{30}\) per client.

5.2) Total savings

After analysing each step of the program, the below table sums up the savings that could generate the implementation of our platform. The process that enables to Sisnet 2003 S.L. to reduce its cost is the third one, which represents two thirds of the total savings. The first and the second steps represents 17.03% and 15.33%, respectively. Notice that as a result of the lack of returns to scale, the weighted savings in each task remains constant no matter the volume of the firm’s output.

<table>
<thead>
<tr>
<th>Firm</th>
<th>Delivery notes (€/mth)</th>
<th>Invoice (€/mth)</th>
<th>Mailing (€/mth)</th>
<th>Total savings (€/mth)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sisnet</td>
<td>96.60 (4 min)</td>
<td>86.94 (3.67 min)</td>
<td>383.61 (8.02 min)</td>
<td>567.15 (15.69 min)</td>
</tr>
<tr>
<td>Firm A</td>
<td>115.92 (4 min)</td>
<td>104.09 (3.67 min)</td>
<td>459.82 (8.02 min)</td>
<td>679.83 (15.69 min)</td>
</tr>
<tr>
<td>Firm B</td>
<td>144.91 (4 min)</td>
<td>130.11 (3.67 min)</td>
<td>574.77 (8.02 min)</td>
<td>849.79 (15.69 min)</td>
</tr>
</tbody>
</table>

*Note: In parentheses the total amount of time saved for each client thanks to the program.*

Numbers speak for themselves. When the employee of Sisnet 2003, S.L. deals with clients without using the program, the time spent, on average, per client is 17.69 minutes, while using the program only takes 2 minutes. Therefore, we can conclude that the program reduces in a 88.69% the time needed to deal with a client without losing quality.

\(^{30}\) Total savings divided the output is equal to the unit benefit, i.e. \(383.61/297 = 459.82/356 = 1.29\)€ per client.
5.4) Conclusions and implications for SISNET 2003 S.L.

The aim of this section is to figure out, in its measure, the impact that the program has had in Sisnet 2003 S.L and, at the same time, to argue which price should be charged to our future customers, once the potential competitors are considered.

The time and money saved with the use of the program provide a profitable opportunity either to minimize its cost structure or to focus on other higher added-value activities such as marketing. After meeting and exchanging this view with the CEO of the company, both parties agreed that there are major market niches yet to be exploited.

Assuming that for each month, on average, Sisnet 2003 S.L. manages all his clients, this can be numerically translated as follows. Before the implementation of the program, to fill out and send by email 150 delivery notes and to generate and send 147 invoices, was translated into 66.14 hours of work. Using the program, for the same volume of clients it is only needed 7.45 hours. As it may be anticipated, if the program is well-implemented, the customer-per-employee ratio or the sales-per employee ratio will improve notoriously.

Moreover, the simple automation of processes -which presents lower costs- allows to eliminate human errors which in general causes significant financial losses as well as a bad reputation for the company brand. Those benefits, also, will be extended to the employee who will be able to accomplish his paperwork from a lean approach, ridding off tedious and repetitive tasks, that would ultimately mine his quality of work.

On the other hand, the implementation of the platform implies, firstly, an important investment in learning how the program works, and then, a time spent to modify some little parts of the code in order to readapt future potential needs.

Nevertheless, we undoubtedly opt for a technology advance, due to the fact that the benefits will clearly outweigh the costs.

6) Market competition

Besides, another way to determine the price that we must charge to customers is by studying the market. That is, by looking for the prices that our competitors are demanding for doing similar tasks.

Nowadays, there exists a huge amount of programs and firms offering a wide variety of automated services, such as accounting, legal, taxes or labour. However, despite of the enormous supply, there are companies that do not have enough money to pay for them or they simply ignore all the benefits that they could get through automation.
When trying to determine a price, we must take into account the fact that our program would offer automating 3 specific tasks; creation of delivery notes, invoices and mailing services, making it as whole quite special. Thus, the uniqueness of our product makes the situation -establishing a price- difficult to be compared. In other words, as expected, we found no similar program that could offer exactly the same services that our tool provides. Yet, we tried to estimate a price by splitting our program into the different services it offers and thus comparing each part with an existent product.

In the following table we observe the possible competitors that we could face:

<table>
<thead>
<tr>
<th>Firm</th>
<th>Service offered</th>
<th>Price charged</th>
</tr>
</thead>
<tbody>
<tr>
<td>XB Consultores Excel Online</td>
<td>Macros &amp; Excel specialist (remote control)</td>
<td>99€/month <strong>the cheapest fee</strong>31</td>
</tr>
<tr>
<td>Midas Gestión (alternative program)</td>
<td>Elaboration of invoices &amp; delivery notes, stock’s control, cash movements, make your logo...</td>
<td>29,90€/month or 399€ (single payment)32</td>
</tr>
<tr>
<td>Anfix (alternative program)</td>
<td>Elaboration of invoices and budgets, connection with banks, accounting, App Android/iOS...</td>
<td>41€/month33</td>
</tr>
</tbody>
</table>

Studying the market, we realized that competitors and our program differ on some characteristics.

First, a wide majority of competitors develop their own program, which it is something that should be taken into account in the future. Moreover, we should find out whether it is because customers ask for it or because competitors ignore all the benefits that they could get from VBA Excel and its macros.

On the other hand, as expected, almost all of them offers financial services. In this sense, we take note of that, since it is something that we could also incorporate in the future.

Summing up, determining the prices via competitors is not as straightforward as it was studying the reduction in labour costs. In this second case, there are a lot of variables which are out of our control, as well as many characteristics that remain incomparable due to major differences in ages between other programs and our product.

7) Critical discussion
Though our program was tested into an existing firm, a word of caution is in order. That is, while the project was developing, a whole series of reasonable doubts and impediments appeared. The aim of this section is to allow the reader to understand the difficulties that we had to deal with and, at the same time, explain how we approached them.

Firstly, we are aware that our program has not already been tested to other small and medium enterprises that differ from our firm in some relevant characteristics (region, resources, activities). However, one of the strengths of this application is the fact that it can be well-adapted to practically all types of sectors, since the core of the program is the same. In this way, once the initial investment -elaborating the code and the program-, which is the most time-consuming task, is made, transferring to other firms that perform on different sectors, is, a priori, more straightforward.

Secondly, since we are definitely not software developers, in a way, we may be unaware of the possible problems that our tool might have, as well as the solutions that firms integrating our program may need. In addition, we are well-conscious of the VBA Excel is not the most user friendly language since it has not been upgraded since 201034, whereas other programs have continuously evolved (i.e RStudio or Python). Yet, the great advantage lies in the immense number of users that daily make use of Excel. Another relevant constraint is the impossibility of working using the cloud, such as other tech-companies, which provides “Google Docs”, a fact that hinders team working.

Lastly, with these cautions in mind, we believe we have been able to demonstrate that even SMEs can benefit from a simple automation of processes, since we, being undergraduate students, have been able to create a tool, apply it to the real world and obtain promising results. That is, through the achievement of our initial objective we have been able to automate routine tasks and improve the company's organization, thus avoiding, to a great extent, human errors.

8) Conclusion

As time passes, the process of automation is disrupting the business world and altering its processes and tasks - so far - performed by employees, as well as the equilibrium between them. Yet, it is believed the business world - as a whole - have still not exploited the full potential of automation, especially those firms restricted to a tight budget, usually included under the Small and Medium Enterprise (SME) term. In this way, we postulate in this analysis that the radius of action of automation can be expanded by creating a simple - economically feasible and beneficial user interface, in the sense that it would increase the efficiency of SMEs, in both economic and time resources terms.

With regard to the quantification part of our analysis, and with the intention to figure out which would be the hypothetical price charged to acquire the program, we firmly state that the most appropriate approach would be selling ourselves as consultants, instead of selling the product as such. The reason resides on the fact that the program requires to be partially readapted to meet client’s needs. Thus, assuming that the program will be fully set up after 5-15 hours of work - which mainly depends on the wide range features required by the customer -, and in order to be strategically well-positioned in the market, we have considered to charge 20€ per hour, thus between 100€ up to 300€ to get the whole user interface settled.

As commented in Section 5, the next challenge will be, once the program is set, to observe in detail how the firm performs. Needless to say, it is not an easy task and we can not expect to obtain a specific outcome at first glance. In any case, we must keep in mind that applying our tool does not only translates into a hypothetical gain in monetary terms but also into a qualitative impact which is difficult to measure (i.e. workers would not have to perform daily routine tasks anymore).

Taking a step further, it is important to raise awareness of the following issue; over the years, as long as we create different kinds of programs to cover different clients’ requests, it would be possible to reduce substantially our fees charged, since the set up would be more easily transferable. Another alternative way we have deeply studied consisted of creating a more user-friendly interface outside the Excel, thus patenting the invention and selling it as a license. However, we considered that this option would face, firstly, a fierce market competition and secondly, a loss of the essence of the cornerstone of this project, i.e., the user accessibility.

Finally, in the near future, it is of our interest to be able to improve both the current version and the extent of the program - to different companies and sectors -, what will mean applying certain and specific changes in the composition of the application. For instance, generating, through the information registered in the invoices, the financial accounting.

As a final note, we are pleased to state that the initial goal has been achieved, based on creating an application that allows to speed up the Back Office paperwork in a substantial way, as well as, in turn, demonstrating that SMEs can also harness the opportunities that automation brings to the business world.
9) Annex

To carry out our program that we have developed, it is necessary to use the Excel code, called Visual Basic for Applications (VBA), which it is accessible through the code “ALT + F11” command or through the “Developer -> Visual Basic” toolbar.

Once we are in the program, in the left side we find the following panel:

The program is divided into three sections: Microsoft Excel Objects, Userforms (Formularios) and Modules (Módulos).

- Microsoft Excel Objects, where it is found the Excel Worksheets created. There, we have all the customers data, sheets for elaborating and modifying delivery notes and invoices among others.

- Userforms. They are custom-built dialog box that facilitates the data entry for elaborating and modifying delivery notes and select those that we want to invoice. We can place text boxes, labels, combo boxes, command buttons, etc. Thus, in our program, it has been used three userforms, as it is showed in the panel (albaran, factura, modificar_albaran).

- Modules, where it is found the most part of the computer code. The whole set of them is what is called a VBA project or program. In case the lector is not familiarized with programming language, the best way to understand how it is structured is thinking that the project is a book and each module represents a chapter. In each chapter -module-, characters -variables, arrays, objects and functions are defined and all of them perform a scene - run an action or a set of actions-. 
Having said that, the technical explanation will focus on the four modules itself. As it is shown in the above picture, there are 4 types of modules:

(i) Formulas

This module is composed by 3 functions: “Generate Folders”, “Get_files”, “SplitingWordFactura”.

The first one generates a folder, for each customer name, in the database sheet “Clients”-only used when the program is set up for the first time.

```vbnet
Sub GenerateFolders()
Dim i As Integer
For i = 2 To 162
   MsgBox "C:\Users\U122308\Desktop\TFG\Clientes" & Worksheets("Clientes").Range("B" & i)
Next i
End Sub
```

The second one, “Get_files”, serve to get all the paths of documents available inside the customer’s folder.

This function is used in “Send email” sheet, in order to create an email, using Outlook application, with documents attached on it.

```vbnet
Sub Get_files()
Dim objFSO As Object
Dim objFolder As Object
Dim objFile As Object
Dim i As Integer

'Create an instance of the FileSystemObject
Set objFSO = CreateObject("Scripting.FileSystemObject")

'Get the folder object
Set objFolder = objFSO.GetFolder("C:\\Users\\U122308\\Desktop\\TFG\\Clientes" & Worksheets("Enviar_mail").Cells(7, 6))

For i = 1 To objFolder.Files.Count
   objFolder.Files(i).ClearContents
   'loop through each file in the directory and prints their names and path
   For Each objFile In objFolder.Files
      'print file name
      Worksheets("Get_files").Cells(i + 1, 1) = objFile.Name
      'print file path
      Worksheets("Get_files").Cells(i + 1, 2) = objFile.Path
   Next objFile
Next i
End Sub
```

Finally, the third one is a technical solution which allows to split a given “Mail Merge” Word document to several new ones. That type of file is composed by different sheets, usually as many rows as there are in the Excel sheet linked (in our case the clients invoiced), which are divided and stored independently in their respective folders. For instance, if the Mail Merge document, 1 file, is composed by 100 sheets, the formula cuts it in 100 files of one page each one.

---

35 i.e. “C:\Users\U122308\Desktop\TFG\Clientes\ADN\Albarán_620190007”,
This formula has been used when the “Generate Invoices” command button is tapped.

```vba
Sub SplittingWordLetters()
    Dim CopyRange As String
    Dim Low As Integer
    Dim sth, sth As Worksheet
    Set sth = Worksheets("Tables Mail Merge Factors")
    Set sth = Worksheets("Factors")
    Low = sth.Range("A:500!").End(xlDown).Row
    For i = 2 To Low
        CopyRange = "A" & i & ":" & "AB" & i
        sth.Range("Al:A2") = sth.Range(CopyRange).Value
        Call RunMergeFactors
    Next i
    End Sub
```

(ii) Mail_Merge_Albaran

The actions of this module are several; transfer the information introduced in the userform “Delivery note” or “Modify delivery note” to a Word or Pdf file called “Albarán.docx” - a general template, create an independent Word document with the name “Albarán_XXXXXXXX36”, print it, save it to the client’s folder and close it. The function name used to call this piece of code is “RunMerge”.

```vba
Sub RunMerge()
    Dim wd As Object
    Dim wshObject As Object
    Dim stWorkbookName, stFolderName, stObjectName As String
    Dim sth As Worksheet
    Set sth = Worksheets("Tables Mail Merge")
    ' Word constants
    Const wdOpenFormatAuto = 4
    Const wdOpenFormatAuto = 4
    Const wdDefaultFirstRecord = 1
    Const wdDefaultLastRecord = -14
    stFolderName = sth.Range("C2")
    stObjectName = "Albarán_" & sth.Range("A1")
    On Error Resume Next
    Set wd = GetObject("Word.Application")
    If wd Is Nothing Then
        Set wd = CreateObject("Word.Application")
    End If
    On Error GoTo 0
    Set stwSource = wd.Documents.Open("C:\Users\UI2230\Desktop\TFG\Albarán.docx")
    ' Cambiar el directorio según la ubicación del documento Albarán.docx
    stwSource.MailMerge.SendDataSource = stWorkbookName & ":" & stFolderName & ":" & stObjectName & ":TPC, else path
    stwSource.MailMerge.MailDocumentType = wdFormLetters
    stwSource.MailMerge.SendDataSource = stwSource.MailMerge.SendDataSource & ":UPDATEBOOKMARKs = False,
    stwSource.MailMerge.SendDataSource = stwSource.MailMerge.SendDataSource & ":RecordSet = wdOpenFormatAuto,
    With stwSource.MailMerge
    .Destination = wdMailToEveDocUnsentMail
    .SupressDefaultLines = True
    With .DataSource
        .FirstRecord = wdDefaultFirstRecord
        .LastRecord = wdDefaultLastRecord
    End With
    .ExecutePreview = False
    End With
    ActiveDocument.SaveAs2 Filename = "C:\Users\UI2230\Desktop\TFG\Albarán.docx" & ":FolderName = " & stFolderName & ":" & stObjectName & ":TPC, else path
    'ActiveDocument.PrintOut
    ActiveDocument.Close
    wd.Visible = False
    stwSource.MailMerge.SendDataSource = False
    Set stwSource = Nothing
    Set wd = Nothing
    End Sub
```

36 The first X is the current month, the next four X’s are the current year and the last four X’s are a number of four digits, the whole represents the delivery note number.
(iii) Mail_Merge_Factura

Similar to the “Mail_Merge_Albaran” module, the aim this chunk of code, is to transfer the information introduced, this time, in the userform “factura” to a Word or Pdf file called “Factura.docx” -a general template-, create an independent Word document with the name “Factura_XXXXXX”, print it, save it to the client’s folder and close it.

The function name used to call this piece of code is “RunMergeFactura”.

(iv) Mailing

The last module is one of the most important parts of the program. The goal of it is to create a customized email for a client; specifying the sender account, the email recipient, the email sender, the email cc (Carbon Copy) field, the body message and so on, using Outlook application. This piece of code is used in the “Send email” sheet, when clicking the button “Enviar Mail”.

---

37 The first X is the current month, the next four X’s are the current year and the last four X’s are a number of four digits, the whole represents the delivery note number.
The function name used to call this chunk of code is “send_email_complete”.

```
Sub send_email_complete()
    Dim outlookApp As Outlook.Application
    Dim myMail As Outlook.MailItem
    Dim source_file As String
    Dim i, j As Integer

    Set outlookApp = New Outlook.Application
    Set myMail = outlookApp.CreateItem(0)
    j = 1
    Do While j < 14
        j = j + 1
        source_file = C:"";
        Worksheets("Envio mail").Range("F2") = "Documentación" Then
            Do While myMail.RichText > 0
                j = j + 1
        End If
    Loop
    Worksheets("Envio mail").Cells(6, i) = ""
    Worksheets("Envio mail").Cells(6, 6) = ""
    myMail.Subject = Worksheets("Envio mail").Range("F2")
    myMail.Body = Worksheets("Envio mail").Range("F2")
    myMail.To = Worksheets("Envio mail").Range("F2")
    myMail.Cc = Worksheets("Envio mail").Range("F2")
    myMail.Bcc = Worksheets("Envio mail").Range("F2")
    myMail.Body = Worksheets("Envio mail").Range("F2")
    Worksheets("Envio mail").Cells(6, i) = ""
    Worksheets("Envio mail").Cells(6, 6) = ""
    myMail.Display = False
End Sub
```

Notice that behind the userform objects as well as in some worksheets, there is an important part of the program, mostly related to the Excel data compilation. For instance, when we click on the button “Accept” in the “Delivery Note” userform, all the data that has been entered must be shifted to a specific row in a specific sheet. Therefore, we have considered not necessary to post it, although if the lector is interested to see that kind of details, he can find the rest of the code entering to the VBA editor.
### Cliente: ADN

<table>
<thead>
<tr>
<th>Cliente: ADN</th>
<th>N.I.F. B1254911</th>
</tr>
</thead>
</table>

### Dirección: Plaça de la Mercè, 12

<table>
<thead>
<tr>
<th>Dirección: Plaça de la Mercè, 12</th>
<th>Código Postal: 8301</th>
</tr>
</thead>
</table>

### Municipio: Mataró

<table>
<thead>
<tr>
<th>Municipio: Mataró</th>
<th>Tel.: 931214522</th>
</tr>
</thead>
</table>

### Operario: Pau Gómez Rico

<table>
<thead>
<tr>
<th>Operario: Pau Gómez Rico</th>
<th>Correo: <a href="mailto:sisnet@sisnet2003.es">sisnet@sisnet2003.es</a></th>
</tr>
</thead>
</table>

### FECHA: 6/3/2019

### Tipo de servicio

<table>
<thead>
<tr>
<th>Tipo de servicio</th>
<th>Cantidad (h)</th>
<th>Precio (€/h)</th>
<th>Importe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Puntual - limpieza de comunidades</td>
<td>24</td>
<td>€14,00</td>
<td>€336,00</td>
</tr>
<tr>
<td>8h durante 3 días</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Otras observaciones:** parking

### Tipo de productos

<table>
<thead>
<tr>
<th>Tipo de productos</th>
<th>Cantidad (h)</th>
<th>Precio (€/h)</th>
<th>Importe</th>
</tr>
</thead>
<tbody>
<tr>
<td>ambientador</td>
<td>2</td>
<td>€3,00</td>
<td>€6,00</td>
</tr>
</tbody>
</table>

### FECHA DE REALIZACIÓN: 5/5/2019

**IMPORTE TOTAL:** €342,00
FACTURA N°: F-620190003  
Fecha de realización: 43590

<table>
<thead>
<tr>
<th>Descripción</th>
<th>Cantidad</th>
<th>Precio</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Puntual: limpieza de comunidades parking</td>
<td>24</td>
<td>14</td>
<td>€336,00</td>
</tr>
<tr>
<td>- 8h durante 3 días</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>- ambientador</td>
<td>2</td>
<td>3</td>
<td>€6,00</td>
</tr>
</tbody>
</table>

Base Imponible: €282,640  
% IVA: 21,00  
IVA: €59,360  
Total factura: €342,00

Forma de Pago: Domiciliación bancaria  
Vencimiento: 31/05/2019  
Observaciones:

IBAN: 0081 0032 44 0001037111
10) Glossary

Visual Basic for Application vocabulary

(i) Types of objects used in the program
- Workbook: is a collection of several worksheets, the whole set is the Excel file.
- Worksheet: is a single page in a workbook file.
- Folder: also called directory, or catalog is a way to organize computer files, i.e. “Clients’ folders”.
- Userform: is a window or dialog-box within Excel user interface, i.e. “Delivery note” userform. Inside the userform it has also been used:
  - ComboBox: combines a text box with a list box to create a drop-down list.
  - Command Button: is a clickable object used to trigger a macro or function, i.e. “Send email”.
  - TextBox: is an empty field where the user can write either a text or an integer.
  - ListBox: is a list from where a user can pick an item, i.e. “Type of service”.
- Message Box (MsgBox): It displays a dialog box, that presents a message to the user. It is a modal window that blocks other actions in the application until the user closes it, i.e. “Are you sure you want to delete this record”, used in the “Delivery note and Invoice” sheet.
- Hyperlink: is a reference to data that the user can directly follow by clicking on it, i.e. used in the index sheet.

(ii) Types of variables used in the program

Numeric data types
- Integer: storage is equal to 2 bytes and the range is from -32,768 to 32,767.
- Double: storage is equal to 8 bytes and the range is from $-1.80e+308$ to $4.94e-324$ for negative values and from $4.94e-324$ to $1.80e+308$ for positive values.

String data types
- String: text variable, storage is equal length of string and the range is from 1 to 65,000 characters.
- Boolean (ByVal): a dummy variable, storage is equal to 2 bytes, range is True (1) or False (0).
- Date: storage is equal to 8 bytes, range is from January 1, 100 to December 31, 9999.
- Range: represents a cell or multiple cells in a given Excel worksheet.
- Array: is a type of variable which enables to store several values of the same typology.
(iii) Types of loops used in the program
- For loop: executes a sequence of statements that is continually repeated until a certain condition is reached, i.e. For i = 1 to 150 -used to create clients’ folders-. 
- Do while: executes a sequence of statements as long as the condition is true, i.e. used to get all the files path in a folder’s client in order to attach the documents into the email.

Loop control statements
- Exit For: Terminates the “For loop” statement when a certain criteria is reached, i.e. when the “counter” is reached a given value.
- On Error Go Next: this statement is used to ignore a given error and resume the execution with the next line of code, i.e. used in the “Mail Merge” module when create delivery notes and invoices.

(iv) Conditional statements used in the program
- If (...) ElseIf (...) Else (...) End If: executes a block of code if a specified condition is true. If the condition is false, another block of code can be executed.
- If (...) Then: executes a block of code if a specified condition is true, i.e. used to identify when a cell is not empty; If A1 <> “” Then ... (statements) ... End If

(v) Types of functions used in the program
- Sub + “the name of the function” (...) End Sub: Is a function procedure in VBA code that after be executed, returns a value or an array of values. It can be used anywhere in the Workbook using “Call + the name of the function” command, since is defined as “Public”.

(vi) Other applications used in the program
Because Microsoft Outlook and Microsoft Word supports automation, you can control them from any program that is written with Microsoft Visual Basic (VBA). Also, the printer has been set up to be controlled through Microsoft Excel program.
Formulas used in quantifying calculations

(i) **Delivery notes**

\[
\text{Without using the program} \left( \frac{e}{mth} \right) = \text{output} \left( \frac{u}{mth} \right) \cdot \text{throughput} \left( \frac{h}{u} \right) \cdot \text{salary} \left( \frac{e}{h} \right)
\]

\[
\text{Using the program} \left( \frac{e}{mth} \right) = \text{output} \left( \frac{u}{mth} \right) \cdot \text{throughput} \left( \frac{h}{u} \right) \cdot \text{salary} \left( \frac{e}{h} \right)
\]

\[
TS \left( \frac{e}{mth} \right) = TC[\{\text{Delivery notes} \mid \text{Without the program}\} - \{\text{Delivery notes} \mid \text{Using the program}\}] \left( \frac{e}{mth} \right)
\]

(ii) **Invoice**

\[
\text{Without the program} \left( \frac{e}{mth} \right) = \text{output} \left( \frac{u}{mth} \right) \cdot \text{throughput} \left( \frac{h}{u} \right) \cdot \text{salary} \left( \frac{e}{h} \right)
\]

\[
\text{Using the program} \left( \frac{e}{mth} \right) = \text{output} \left( \frac{u}{mth} \right) \cdot \text{throughput} \left( \frac{h}{u} \right) \cdot \text{salary} \left( \frac{e}{h} \right)
\]

\[
TS \left( \frac{e}{mth} \right) = TC[\{\text{Invoice} \mid \text{Without the program}\} - \{\text{Invoice} \mid \text{Using the program}\}] \left( \frac{e}{mth} \right)
\]

(iii) **Mailing**

\[
\text{Without the program} \left( \frac{e}{mth} \right) = \text{output} \left( \frac{u}{mth} \right) \cdot \text{throughput} \left( \frac{h}{u} \right) \cdot \text{salary} \left( \frac{e}{h} \right)
\]

\[
\text{Using the program} \left( \frac{e}{mth} \right) = \text{output} \left( \frac{u}{mth} \right) \cdot \text{throughput} \left( \frac{h}{u} \right) \cdot \text{salary} \left( \frac{e}{h} \right)
\]

\[
TS \left( \frac{e}{mth} \right) = TC[\{\text{Mailing} \mid \text{Without the program}\} - \{\text{Mailing} \mid \text{Using the program}\}] \left( \frac{e}{mth} \right)
\]

(iv) **Total savings**

\[
TS \left( \frac{e}{mth} \right) = TS_{\text{Delivery note}} \left( \frac{e}{mth} \right) + TS_{\text{Invoice}} \left( \frac{e}{mth} \right) + TS_{\text{Mailing}} \left( \frac{e}{mth} \right)
\]
11) Bibliography


