

MIREL CONCEPT-ALIGNED TOOL FOR ANALYSING EMPLOYEE PARTICIPATION IN PROJECTS

Bartosz SZCZEŚNIAK

Silesian University of Technology, Faculty of Organisation and Management, Institute of Economics and Informatics, bartosz.szczesniak@polsl.pl, ORCID: 0000-0002-9683-4629

Purpose: The main purpose of the study is to develop and present a spreadsheet-based tool aligned with the MiRel concept to support data processing process in the analysis of employee participation in project implementation by automating the generation of a required report. The process for which the tool is created is an actual process implemented in an organisation operating on the market.

Design/methodology/approach: The process in which a required report is generated has been analysed with particular attention to the data structure typical of the problem in question. The entities it contains along with the associated attributes have been identified. Connections between individual entities have been illustrated using an entity relationship model. The author has proposed methods for both data entry as well as entity representation in the spreadsheet-based tool. Mechanisms employed to transfer data from a form to tables representing individual entities have been discussed. The process in which the required report is created has been divided into a number of steps. For each of them, a target effect has been defined and specific mechanisms enabling the target to be attained have been proposed.

Findings: A spreadsheet provides an utterly sufficient environment to create a tool supporting the process subject to the research. Such a tool can be successfully created in line with the MiRel concept. Despite the fact that the required report in question is strictly defined in terms of its graphical form that does not match the charts and graphic elements available in MS Excel, it is possible to generate it automatically without having to use the VBA code. Its adequate graphical form can be obtained by means of conditional formatting provided that data have been correctly arranged in cells. Where the report is to be generated in a purely graphical form, the tool in question can be expanded with a relatively simple VBA procedure enabling such a report to be generated on the basis of its current form.

Practical implications: The solution proposed in the paper can be successfully applied in practice to support the process analysed.

Social implications: The solution proposed in the paper makes it possible to streamline the performance of the process analysed by relieving employees of the laborious and repeatable obligatory activities as well as reducing the risk of errors appearing in the report created in the process.

Originality/value: The concept described in the paper is the author's original solution.

Keywords: spreadsheet, database, data processing, improvement of information processes, project management.

Category of the paper: Case study and technical paper.

1. Introduction

At this day and age, information is one of the most precious resources of organisations, which is why one can successfully build competitive advantage by using it in a competent manner. With regard to the foregoing, effective implementation of processes in which information is generated, acquired, stored, processed, shared, interpreted, or utilised, or in other words – information processes (Oleński, 2001, 2002), is one of the key factors which determine the success of every business operating in the market.

Given the contemporary level of advancement of information technologies, a decided majority of information processes is implemented with adequate support from more or less sophisticated tools. And even though solutions of this kind are widely accessible, one can still come across processes where all or most of data processing is handled manually. The fact that the aforementioned processes continue to function nowadays may be attributable to the ever changing requirements pertaining to data logging and processing, introducing successive innovative data analysis concepts, the need to focus on a specific fraction of an enterprise in a more detailed manner than before, or the emergence of new and unprecedented issues. The necessity of handling information processes manually undoubtedly represents a serious problem, one which may potentially be solved by developing a dedicated IT tool on an in-house basis.

Needless to say that spreadsheets are definitely very universal and common tools, familiar to nearly all organisations. One can find numerous suggestions in the literature on the subject as to how they can be utilised in diverse operational areas. They prove particularly useful when employed to support information processes in matters such as finance, sales, or marketing management (Jackson, Staunton, 2004; Kuzstelak, 2020; Winston, 2019; Próchnicki, 2012), work time organisation (Szczęśniak, 2010; Zasadzień et al., 2017), or quality management (Knight, 2009; Szczęśniak, Molenda, 2013; Szczęśniak, 2021).

The high flexibility of spreadsheets makes them suitable for both ongoing processing of the data sets in disposal and for designing dedicated tools to automate such a processing operation. A solution which supports the creation of such tools is the author's signature concept known as MiRel (Szczęśniak, 2017, 2018a, 2018b, 2020a). Its main pre-assumption is that spreadsheets are used to develop IT tools based on an explicitly defined relational data model. An example of a tool created in line with this concept to support the process of analysis of employee participation in the implementation of individual projects has been discussed further on in this paper. The presented description is a case study for a real problem identified in an organization operating on the market.

2. Characteristics of the process analysed

An information process identified within an organisation whose core operation boils down to performing construction-related project activities consists in generating a report containing a list of employees who participate in the implementation of individual projects. This report is developed into a diagram modelled on the one provided in Figure 1.

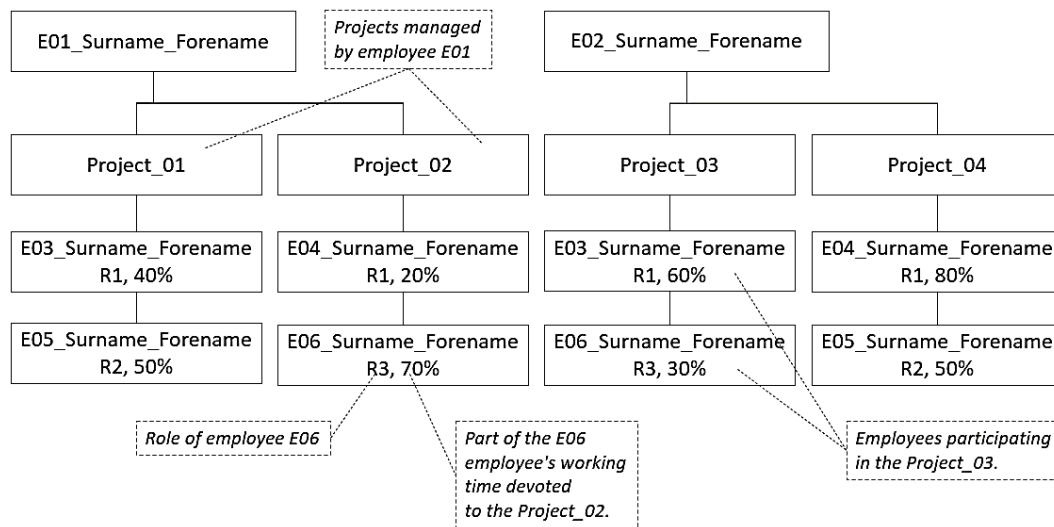


Figure 1. Structure of the report being generated.

It is developed with reference to data stored in a table whose layout has been shown in Figure 2.

Surname	Forname	Role	Projects				
			Project_01	Project_02	Project_03	Project_04	...
E01_Surname	E01_Forname	MN	1	1			...
E02_Surname	E02_Forname	MN			1	1	...
E03_Surname	E03_Forname	R1	0,4		0,6		...
E04_Surname	E04_Forname	R1		0,2		0,8	...
E05_Surname	E05_Forname	R2	0,5			0,5	...
E06_Surname	E06_Forname	R3		0,7	0,3		...
...

Figure 2. Structure of the table on which the report is based.

The report is created at the department level of the organization. One of the employees prepares data in the form of a table as shown in Figure 1. In this table, each project and each employee may appear only once. Each employee can play one specific role in the project. If it is a managerial role, this fact is simply marked in the table. If it is a different role, the table includes a number specifying how much of the employee's working time is spent on a given project. Based on the prepared table, the next employee prepares the final form of the report in accordance with Figure 2. In this report, individual roles should appear in a predetermined order. At the moment, the report is created manually. This activity is both

laborious and repeatable. Where this is the case, it seems to be a good solution to make sure that this process becomes fully automated, and to this end, a tool aligned with the MiRel concept and developed in a spreadsheet can prove particularly adequate.

3. Data structure applied in the tool

With regard to the problem in question, the following entities have been identified: role, project, employee, and employee-project. An instance of the *employee* entity represents an employee of a given organisation, while that of the *project* entity – a project implemented by the latter. An instance of the *role* entity refers to the role which an employee can perform under a project. An instance of the *employee-project* entity corresponds to the fact that a specific employee participates in a specific project. The links between individual entities as well as the attributes identified for them have been depicted in Figure 3.

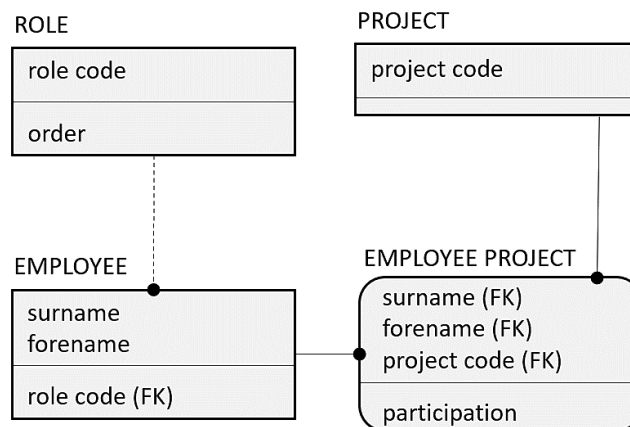


Figure 3. IDEF1x standard-compliant (Allen, 2006) model of inter-entity relations for the tool proposed.

The first of the attributes identified for entity *role* is *role code*. It is a two-letter identification code of the role an employee can perform under a project. The second attribute is *order*. The values it assumes are integers larger than 0. These numbers define the order in which an employee performing a given role will occur in the report being generated. The smaller the value, the sooner one occurs in the report. Exactly one role must take the value of 1 for attribute *order*. This role is considered as the main managerial role. Employees holding this role will occur in the upper section of the report. Other values of attribute *order*, i.e. those larger than 1, can reoccur. Another entity is that of *project*. For this entity, the only attribute which has been identified is a project-identifying code. For another entity, i.e. *employee*, three attributes have been identified. The first two of them are an employee's *surname* and *forename*. It has been assumed that a combination of *surname* and *forename* in the range analysed is unique, and consequently, these attributes form a composite primary key. The third attribute identified for this entity is *role code*, whose value defines the role a given employee performs under

projects. The last entity is *employee-project*, and four attributes have been defined for it. The first three are *surname*, *forename*, and *project code*. These attributes define the links between a given employee and a specific project, representing a composite primary key. The fourth attribute is that of *participation*. It takes numerical values larger than 0 and smaller than or equal to 1. The value of this attribute represents the fraction of a given employee's work which has been allocated to a given project. In the case of an employee assigned a role with the *order* attribute value of 1, the number 1 is entered irrespective of the number of projects which the given employee supervises.

Individual entities in the tool created as proposed are represented in conformity with type I of entity representation, as assumed under the MiRel concept, namely in the form of tables, as in a typical relational database. Each table has been placed in a separate worksheet whose name corresponds to the entity name. Their layout has been shown in Figure 4. One of the premises underlying the solution in question is that it can contain data of no more than 10 roles, 50 employees, and 20 projects. Consequently, it has been further assumed that, in worksheet Roles, data will be contained in rows 2 through 11, in worksheet Projects – in rows 2 through 21, while in worksheet Employees – in rows 2 through 51. What has also been envisaged is that no more than 500 employee assignments to projects can be entered in the tool, and therefore, in worksheet Employees_Projects, data can be contained in rows 2 through 501.

Worksheet: Roles		Projects		Employees			Employees_Projects						
	A	B		A	B	C		A	B	C	D		
1	RoleCode	Order	1	ProjectCode	1	Surname	Forename	Role	1	Surname	Forename	Project	Participation
2	MN	1	2	P01	2	SN01	FN01	MN	2	SN01	FN01	P01	1
3	R1	4	3	P02	3	SN02	FN02	R1	3	SN01	FN01	P04	1
4	R2	3	4	P03	4	SN03	FN03	MN	4	SN01	FN01	P06	1
5	R3	2	5	P04	5	SN01	FN02	R1	5	SN02	FN02	P01	0,3
6	R4	5	6	P05	6	SN02	FN03	R2	6	SN02	FN02	P03	0,3
7			7	P06	7	SN03	FN01	MN	7	SN02	FN02	P05	0,4
8			8		8	SN01	FN03	R3	8	SN03	FN03	P03	1

Figure 4. Representation of entities in the tool in question.

Where the tool in question is used, only the data in worksheet Roles are entered in the layout provided in Figure 4. All other data are entered in the form shown in Figure 5, conforming with the form currently filled while performing the process in question in a routine manner, and then, by way of suitable formulas, they are transferred to corresponding tables.

	A	B	C	D	E	F	G	H	I	J	X
1											
2				C	Project						
3		Surname	Forname!	Role	P01	P02	P03	P04	P05	P06	
4		SN01	FN01	MN	1,00			1,00		1,00	
5		SN02	FN02	R1	0,30		0,30		0,40		A
6		SN03	FN03	MN			1,00		1,00		
7		SN01	FN02	R1		0,50		0,20		0,30	
8		SN02	FN03	R2	0,40	0,10	0,10	0,20	0,10	0,10	
9		SN03	FN01	MN		1,00					
10		SN01	FN03	B	0,60				0,20	0,20	
53											

Figure 5. Layout of cells in worksheet Form with the highlighted ranges corresponding to individual tables.

This form is to be found in the worksheet titled Form. It is where one can identify the ranges corresponding to the tables shown in Figure 4. Range A corresponds to table Projects, range B corresponds to table Employees, and range C – to table Employees_Projects. The data layout within ranges A and B matches the data layout in the target tables, and so it suffices to employ simple formulas for data transfer purposes. In the case of range C, the form’s data layout is compatible with type II of in-worksheet representation, and therefore, in order to transfer the data to the appropriate table compatible with type I of representation, one of the solutions proposed in the literature has been applied (Szczęśniak, 2018a). Under this solution, auxiliary table T1 has been added to worksheet Form and three auxiliary tables to worksheet Employees_Projects. Both the auxiliary column and its formula have been shown in Figure 6. In terms of how formulas are presented in the paper, it has been assumed that their form, as provided, is to be considered representative of the top left cell of the range in which they occur. For formula FA1 in auxiliary table T1, it is the form to be found in cell AA4.

	A	B	C	D	E	F	G	H	I	J	X	AA	AB	AC	AD	AE	AF	AT
1																		
2					Project													
3		Surname	Forname!	Role	P01	P02	P03	P04	P05	P06		T1						
4		SN01	FN01	MN	1,00			1,00		1,00		0			3		5	
5		SN02	FN02	R1	0,30		0,30		0,40			20		22		24		
6		SN03	FN03	MN			1,00		1,00					42		44		
7		SN01	FN02	R1		0,50		0,20		0,30			61		63		65	
8		SN02	FN03	R2	0,40	0,10	0,10	0,20	0,10	0,10		80	81	82	83	84	85	
9		SN03	FN01	MN		1,00							101					
10		SN01	FN03	R3	0,60				0,20	0,20		120				124	125	
53																		

FA1 =IF(E4<>"";(COLUMN(A1))-1+COLUMNS(\$E\$3:\$X\$3)*(ROW(A1)-1);"")

Figure 6. Cell layout and formula in an auxiliary table in worksheet Form.

Individual cells of the auxiliary table correspond to the form cells where the numbers representing the employee participation in the project implementation are entered. The number which conveys information about the row and column in the form where a value previously entered is to be found occurs in each auxiliary table cell if this value exists in the corresponding form cell. This number is entered according to formula 1:

$$n = a - 1 + d(b - 1) \quad (1)$$

where:

n – number entered in auxiliary table T1,

a – number of the current column within range E4:X53 in the form,

b – number of the current row within range E4:X53 in the form,

d – total number of columns in range E3:X3 in the form, representing the maximum number of projects which the tool may comprise.

The auxiliary columns in worksheet Employees_Projects along with their respective formulas have been shown in Figure 7.

	A	B	C	D	F	G	H
	FB4	F54	FB6	FB7	FB1	FB2	FB3
1	Surname	Forename	Project	Participation	C1	C2	C3
2	SN01	FN01	P01	1	0	1	1
3	SN01	FN01	P04	1	3	1	4
4	SN01	FN01	P06	1	5	1	6
5	SN02	FN02	P01	0,3	20	2	1
6	SN02	FN02	P03	0,3	22	2	3
7	SN02	FN02	P05	0,4	24	2	5
8	SN03	FN03	P03	1	42	3	3
501							

FB1	=IFERROR(SMALL(Form!\$AA\$4:\$AT\$53;ROW(A1));"")
FB2	=IF(F2<>"";ROUNDDOWN(F2/COLUMNS(Form!\$E\$3:\$X\$3);0)+1;"")
FB3	=IF(F2<>"";MOD(F2;COLUMNS(Form!\$E\$3:\$X\$3))+1;"")
FB4	=IF(F2<>"";INDEX(Form!\$B\$4:\$B\$53;Employees_Projects!G2;1);"")
FB5	=IF(F2<>"";INDEX(Form!\$C\$4:\$C\$53;Employees_Projects!G2;1);"")
FB6	=IF(F2<>"";INDEX(Form!\$E\$3:\$X\$3;1;Employees_Projects!H2);"")
FB7	=IF(F2<>"";INDEX(Form!\$E\$4:\$X\$53;Employees_Projects!G2;Employees_Projects!H2);"")

Figure 7. Layout of columns in worksheet Employees_Projects with the formulas applied.

In consecutive rows, formula FB1 returns consecutive numbers retrieved from auxiliary table T1. Based on these numbers, formula FB2 and formula FB3 identify the numerical designation of the form's row and column, respectively, from which corresponding data should be acquired. The row and column numbers are established according to formulas 2 and 3:

$$r = \left\lfloor \frac{n}{d} \right\rfloor + 1 \quad (2)$$

$$c = \text{mod} \left(\frac{n}{d} \right) + 1 \quad (3)$$

where:

r – number of the row in range E4:X53 in the form,

c – number of the column in range E4:X53 in the form,

n – numerical value returned by formula FB1 from auxiliary column C1,

d – total number of columns in range E3:X3 in the form, representing the maximum number of projects which the tool may comprise.

With reference to the row and column numbers previously identified, formulas FB4, FB5, FB6 and FB7 transfer the values of attributes Surname, Forename, Project, and Participation from the form to table Employees_Projects.

4. Generating the report

Based on the data collated in the layout corresponding to the one shown in Figure 4, the report envisaged in the analysed process is generated, as schematically depicted in Figure 1. The report generation process can be divided into a number of steps. The first one consists in identifying the employees who supervise projects as well as the projects assigned to these employees. In this step, auxiliary columns have been added in worksheets Employees and Employees_Projects. These columns, along with their respective formulas, have been shown in Figures 8, 9, and 10, respectively.

	A	B	C	E
1	Surname	Forename	Role	FC1 C1
2	SN01	FN01	MN	SN01-FN01
3	SN02	FN02	R1	SN02-FN02
4	SN03	FN03	MN	SN03-FN03
5	SN01	FN02	R1	SN01-FN02
6	SN02	FN03	R2	SN02-FN03
7	SN03	FN01	MN	SN03-FN01
8	SN01	FN03	R3	SN01-FN03
51				
FC1	=IF(A2<>"",CONCAT(A2;"-";B2);"")			

Figure 8. Auxiliary column in worksheet Employees.

Formula F1 to be found in the auxiliary column in worksheet Employees returns a code which constitutes a combination of the values of the attributes belonging to the primary key of the table. An identical code is generated in the auxiliary column in worksheet Employees_Projects by means of formula FD1. Formula FD2 returns the number of the row in worksheet Employees containing the code from the current row in worksheet Employees_Projects. It is from that row that formula FD3 retrieves the corresponding code of the employee role under a given project. Formula FD4 returns a number which specifies how many times the current code from auxiliary column C21 has occurred in the preceding rows of the worksheet as well as in the current one. Next, formula FD5 returns the value of 1 in the rows where the given code in column C21 occurs for the first time and the code in auxiliary column C23 is the code assigned the order of 1. These values are then totalled by formula FD6, whereby consecutive natural numbers occur in auxiliary column C26 in the rows containing data of the project-supervising employees, and they are the rows in which these data occur for the first time. Additionally, based on the code acquired from column C21, an extended code is generated in auxiliary column C27, containing a number that specifies which consecutive occurrence of the code from column C21 it is.

	A	B	C	D	J	K	L	M	N	O	P
1	Surname	Forename	Project	Participation	FD1 C21	FD2 C22	FD3 C23	FD4 C24	FD5 C25	FD6 C26	FD7 C27
2	SN01	FN01	P01	1	SN01-FN01	1	MN	1	1	1	SN01-FN01-1
3	SN01	FN01	P04	1	SN01-FN01	1	MN	2	0	1	SN01-FN01-2
4	SN01	FN01	P06	1	SN01-FN01	1	MN	3	0	1	SN01-FN01-3
5	SN02	FN02	P01	0,3	SN02-FN02	2	R1	1	0	1	SN02-FN02-1
6	SN02	FN02	P03	0,3	SN02-FN02	2	R1	2	0	1	SN02-FN02-2
7	SN02	FN02	P05	0,4	SN02-FN02	2	R1	3	0	1	SN02-FN02-3
501											

FD1	=IF(A2<>"";CONCAT(A2;"-";B2);"")
FD2	=IF(A2<>"";MATCH(J2;Employees!\$E\$2:\$E\$51;0);"")
FD3	=IF(A2<>"";INDEX(Employees!\$C\$2:\$C\$51;Employees_Projects!K2;1);"")
FD4	=IF(A2<>"";COUNTIF(\$J\$2:J2;J2);"")
FD5	=IF(A2<>"";IF(AND(L2=INDEX(Roles!\$A\$2:\$A\$11;MATCH(1;Roles!\$B\$2:\$B\$11;0);1);M2=1);1;0);"")
FD6	=IF(A2<>"";SUM(\$N\$2:N2);"")
DF7	=IF(A2<>"";CONCAT(J2;"-";M2);"")

Figure 9. First group of auxiliary columns of the report generating mechanism in worksheet Employees_Projects.

Using the data to be found in auxiliary columns C21 and C26, formula FE1 creates a list of the codes which identify the project supervisors. For each of the employees thus identified, formula FE2 determines the number of projects they supervise, while formula FE3 calculates a cumulative sum based on this number.

	A	B	C	D	R	S	T	U	V	W	X	Y
1	Surnan	Forena	Projec	Partic	FE1 C28	FE2 C29	FE3 C30	FE4 C31	FE5 C32	FE6 C33	FE7 C34	FE8 C35
2	SN01	FN01	P01	1	SN01-FN01	3	3	1	SN01-FN01	1	SN01-FN01-1	1
3	SN01	FN01	P04	1	SN03-FN03	2	5	1	SN01-FN01	2	SN01-FN01-2	2
4	SN01	FN01	P06	1	SN03-FN01	1	6	1	SN01-FN01	3	SN01-FN01-3	3
5	SN02	FN02	P01	0,3				2	SN03-FN03	1	SN03-FN03-1	7
6	SN02	FN02	P03	0,3				2	SN03-FN03	2	SN03-FN03-2	8
7	SN02	FN02	P05	0,4				3	SN03-FN01	1	SN03-FN01-1	18
21	SN01	FN03	P05	0,2								

FE1	=IFERROR(INDEX(\$J\$2:\$J\$501;MATCH(ROW(A1);\$O\$2:\$O\$501;0));"")
FE2	=IF(R2<>"";COUNTIF(\$J\$2:\$J\$501;R2);"")
FE3	=IF(R2<>"";SUM(\$S\$2:S2);"")
FE4	=IF(ROW(A1)<=MAX(\$T\$2:\$T\$21);XMATCH(ROW(A1);\$T\$2:\$T\$21;1);"")
FE5	=IF(U2<>"";INDEX(\$R\$2:\$R\$21;U2;1);"")
FE6	=IF(U2<>"";COUNTIF(\$V\$2:V2;V2);"")
FE7	=IF(U2<>"";CONCAT(V2;"-";W2);"")
FE8	=IF(U2<>"";MATCH(X2;\$P\$2:\$P\$501;0);"")

Figure 10. Second group of auxiliary columns of the report generating mechanism in worksheet Employees_Projects.

Formula FE4 enters the subsequent natural number as many times as there are projects supervised by a given employee. For instance, the first employee identified by the code SN01-FN01 supervises three projects, which is why the number 1 occurs in three consecutive rows in auxiliary column C31, while the second employee identified by the code SN03-FN03 supervises two projects, hence the number 2 occurring in two consecutive rows in auxiliary column C31. Based on the numbers to be found in auxiliary column C31, formula FE5 returns a code which identifies a given employee in as many consecutive rows as there are projects supervised by this employee, while formula FE6 calculates how many times the given code occurs. Based on the values in auxiliary columns C32 and C33, formula FE7 generates a code which contains both an employee-identifying code and a number specifying which consecutive occurrence of the given code it is. The structure of the extended code conforms with the structure of the previously generated code to be found in auxiliary column C27. Based on the extended code, formula FE8 specifies the number of the row in table Employees_Projects from which the code of the project supervised by a given employee should be acquired. For instance, in table Employees_Projects, the consecutive codes of the projects supervised by an employee identified by the code SN01-FN01 are to be found in rows 1, 2 and 3, while for an employee identified by the code SN03-FN03, the consecutive codes of the projects they supervise are to be found in rows 7 and 8.

The second step towards generating the report is to prepare and set the order of the contents which the report will contain as information about individual employees. To this end, the third group of nine auxiliary columns have been added in worksheet *Employees_Projects*, which have been shown in Figure 11 along with their respective formulas.

	A	B	C	D	AA	AB	AC	AD	AE	AF	AG	AH	AI
					FG1	FG2	FG3	FG4	FG5	FG6	FG7	FG8	FG9
1	Surnan	Forena	Projec	Partic	C38	C39	C40	C41	C42	C43	C44	C45	C46
2	SN01	FN01	P01	1	SN01 FN01	1	1	1	1	P01	SN01 FN01	1	P01-1
3	SN01	FN01	P04	1	SN01 FN01	1	1	2	2	P04	SN01 FN01	1	P04-1
4	SN01	FN01	P06	1	SN01 FN01	1	1	3	3	P06	SN01 FN01	1	P06-1
5	SN02	FN02	P01	0,3	SN02 FN02R1: 30%	2	4	16	7	P03	SN03 FN03	1	P03-1
6	SN02	FN02	P03	0,3	SN02 FN02R1: 30%	2	4	17	8	P05	SN03 FN03	1	P05-1
7	SN02	FN02	P05	0,4	SN02 FN02R1: 40%	2	4	18	18	P02	SN03 FN01	1	P02-1
501													

FG1	=IF(A2<>"";CONCAT(A2;" ";B2;IF(L2<>INDEX(Roles!\$A\$2:\$A\$11;MATCH(1;Roles!\$B\$2:\$B\$11;0);1);CONCAT(CHAR(10);L2;" ";D2*100,"%");"")));"")
FG2	=IF(A2<>"";MATCH(L2;Roles!\$A\$2:\$A\$11;0);"")
FG3	=IF(A2<>"";INDEX(Roles!\$B\$2:\$B\$11;AB2);"")
FG4	=IF(A2<>"";RANK.EQ(AC2;\$AC\$2:\$AC\$501;1)+COUNTIF(\$AC\$2:AC2;AC2)-1;"")
FG5	=IF(A2<>"";MATCH(ROW(A1);\$AD\$2:\$AD\$501;0);"")
FG6	=IF(A2<>"";INDEX(\$C\$2:\$C\$501;AE2;1);"")
FG7	=IF(A2<>"";INDEX(\$AA\$2:\$AA\$501;AE2;1);"")
FG8	=IF(A2<>"";COUNTIF(\$AF\$2:AF2;AF2);"")
FG9	=IF(A2<>"";CONCAT(AF2;"-";AH2);"")

Figure 11. Third group of auxiliary columns of the report generating mechanism in worksheet *Employees_Projects*.

The values to be found in individual rows in auxiliary columns C38 through C41 correspond to the values to be found in these rows in table *Employees_Projects*. The values in auxiliary columns C42 through C46 are organised in the order in which they are to occur in the report.

Formula FG1 in auxiliary column C38 creates the actual content which should be contained in the report. For the persons who supervise project implementation, this content is limited to their forename and surname. In the case of all other persons, besides their forename and surname, this content comprises the role performed under a project as well as a number, expressed in per cents, which specifies the employee participation in the project. The role and participation should be displayed in the subsequent row of text. Next, formula FG2 returns a number which designates the row in table *Role* where the role performed by a given employee is defined, while formula FG3 returns a number which defines the order according to which the person performing this role should appear in the report. According to this number, formula FG4 generates a number which determines the position in the range containing sorted values at which the values from the current row of the unsorted range will be stored. The following is taken into account when establishing this position: the number from auxiliary column C40 and the order

in the unsorted range. Formula FG5 uses that number to return the numerical designation of the row in the unsorted range from which the corresponding values are to be transferred to the given row in the sorted range. In the sorted range, formula FG6 displays the relevant project code, while formula FG7 – the text to appear in the report. Additionally, formula FG8 counts the successive occurrences of the project code in the sorted range. Using that number as well as the project code to be found in auxiliary column C43, formula C46 creates a code according to which the right text is obtained and transferred to the corresponding cell in the report.

The proper report is generated in worksheet Report, in the cells of range B8:AN52. It can comprise up to 20 projects, while besides the supervisor, up to 21 employees can be designated under a single project. There are two sections in the report. One lists the employees who supervise projects as well as the projects they have been assigned. The formulas applied in this section have been entered in individual rows. The other section presents all the persons participating in the implementation of individual projects in the right order. There is a single uniform formula which has been applied in this section. On account of the report formatting, across its entire range, the relevant values occur in every second row and every second column. Both the cell layout and the formulas applied have been shown in Figure 12.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1													
7		1		2		3		7		8		18	
8		SN01 FN01	SN01 FN01	SN01 FN01		SN01 FN01		SN03 FN03		SN03 FN03		SN03 FN01	
10		P01		P04		P06		P03		P05		P02	
12		SN01 FN03	SN02 FN03	SN01 FN03		SN02 FN03		SN01 FN03		SN02 FN03		SN02 FN03	
		R3: 60%	R2: 20%	R3: 20%		R2: 10%		R3: 20%		R2: 10%		R2: 10%	
14		SN02 FN03	SN01 FN02	SN02 FN03		SN02 FN02		SN02 FN03		SN02 FN03		SN01 FN02	
		R2: 40%	R1: 20%	R2: 10%		R1: 30%		R2: 10%		R2: 10%		R1: 50%	
16		SN02 FN02		SN01 FN02				SN02 FN02					
		R1: 30%		R1: 30%				R1: 40%					

FF1	=IF(ISEVEN(COLUMN(B1));INDEX(Employees_Projects!\$Y\$2:\$Y\$21;COLUMN(B1)/2);"")
FF2	=IFERROR(INDEX(Employees_Projects!\$AG\$2:\$AG\$501;MATCH(CONCAT(B\$10;"-";1);Employees_Projects!\$AI\$2:\$AI\$501;0);1);"")
FF3	=IFERROR(INDEX(Employees_Projects!\$C\$2:\$C\$501;B7;1);"")
FF4	=IFERROR(INDEX(Employees_Projects!\$AG\$2:\$AG\$501;MATCH(CONCAT(B\$10;"-";ROW(A4)/2);Employees_Projects!\$AI\$2:\$AI\$501;0);1);"")

Figure 12. Layout of cells and formulas in the report.

A single auxiliary row has been added directly above the report. It contains formula FF1 which transfers values from worksheet Employees_Projects, from auxiliary column C35, according to which, formula FF3 retrieves the project code from the corresponding row in table Employees_Projects. Based on the project code, formula FF2 generates the corresponding code, determines its position in auxiliary column C46 in worksheet Employees_Projects, and then retrieves the text to be displayed in the report for the person who supervises the given project from the corresponding row in auxiliary column C44. It is the layout of the formulas applied in the auxiliary columns in worksheet Employees_Projects which guarantees that the

projects supervised by a single employee will be displayed next to one another in the report. Functioning similarly to formula FF2, also with reference to the project code, formula FF4 acquires consecutive texts envisaged to be displayed in the report for all the employees participating in the project implementation.

The right graphical form of the final report has been obtained by way of conditional formatting. In the first place, for the entire report, white has been defined as the fill colour for all cells. For a row which contains information about the supervisors, i.e. for range $\$B\$8:\$AN\8 , white has been defined as the font colour, followed by the application of three formatting rules:

- rule 1 – top and bottom cell edge is visible, and Blue, Accent 5, Lighter 80% is defined as the cell fill colour for the cells which meet the condition $=\text{OR}(\text{B8}<>""; \text{AND}(\text{B8}="" ; \text{A8}<>"" ; \text{C8}=\text{A8}))$,
- rule 2 – right cell edge is visible for the cells which meet the condition $=\text{AND}(\text{COUNTIF}(\text{B8}:\$A\$8;\text{B8})=1; \text{B8}<>"")$,
- rule 3 – right cell edge is visible and black is defined as the font colour for the cells which meet the condition $=\text{AND}(\text{COUNTIF}(\$B\$8:\text{B8};\text{B8})=1; \text{B8}<>"")$.

For all the remaining cells in the report, i.e. for range $\$B\$9:\$AN\52 , the following two formatting rules have been applied:

- all cell edges are visible for the cells which meet the condition $=\text{B9}<>""$
- left cell edge is visible for the cells which meet the condition $=\text{AND}(\text{B9}="" ; \text{B8}<>"" ; \text{B10}<>"")$

It is for the application of the above formulas that the form demonstrated in Figure 13 has ultimately been obtained.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1													
8		SN01 FN01						SN03 FN03				SN03 FN01	
9													
10		P01		P04		P06		P03		P05		P02	
12		SN01 FN03 R3: 60%		SN02 FN03 R2: 20%		SN01 FN03 R3: 20%		SN02 FN03 R2: 10%		SN01 FN03 R3: 20%		SN02 FN03 R2: 10%	
14		SN02 FN03 R2: 40%		SN01 FN02 R1: 20%		SN02 FN03 R2: 10%		SN02 FN02 R1: 30%		SN02 FN03 R2: 10%		SN01 FN02 R1: 50%	
16		SN02 FN02 R1: 30%				SN01 FN02 R1: 30%				SN02 FN02 R1: 40%			

Figure 13. Ultimate form of the report being created.

In the form proposed in this paper, the tool in question uses only built-in spreadsheet functions as well as the conditional formatting mechanism. In cases where the report should be generated in a graphical form, the tool can be expanded with an additional, yet relatively simple, code developed in Visual Basic for Applications, as demonstrated in Figure 14.

```

Public Sub CreateReport()
Dim sL As Single, sT As Single, sW As Single, sH As Single, strP As String
Dim shtI As Worksheet, shtO As Worksheet
Dim rngC As Range, shpC As Shape, shpP As Shape, conC As Shape
sL = 10: sT = 20: sW = 100: sH = 50
Set shtI = Worksheets("Report")
Set shtO = Worksheets("ReportGR")
For Each shpC In shtO.Shapes
    shpC.Delete
Next shpC
For iC = 1 To 20
    For iR = 1 To 23
        Set rngC = shtI.Cells(2 * iR + 6, 2 * iC)
        If rngC = "" Then Exit For
        If iR > 1 Or rngC <> strP Then
            Set shpC = shtO.Shapes.AddShape _
                (msoShapeRectangle, sL + (iC - 1) * (sW + sL), sT + (iR - 1) * (sH + sT), sW, sH)
            shpC.TextFrame2.TextRange.Text = rngC
            If iR = 1 Then
                strP = rngC
                Set shpM = shpC
            End If
        End If
        If iR = 1 Then
            Set shpP = shpM
        Else
            Set conC = shtO.Shapes.AddConnector(msoConnectorElbow, 10, 10, 20, 20)
            conC.ConnectorFormat.BeginConnect shpP, 3
            conC.ConnectorFormat.EndConnect shpC, 1
            Set shpP = shpC
        End If
    Next iR
Next iC
End Sub

```

Figure 14. Code of the procedure to generate the report in a graphical form.

The CreateReport() procedure, as provided in the figure above, generates the report in a graphical form in worksheet ReportGR by making use of the report's current form. This report is composed of rectangles whose width and height is defined by variables sW and sH, respectively, while the horizontal and vertical spaces between them are defined by variables sL and sT, respectively.

5. Conclusions

The solution proposed and discussed in the paper proves that one can successfully use the spreadsheet environment to create tools capable of providing full automation of data processing. Such tools can be created in line with the MiRel concept, assuming that all the tool's data are stored in tables whose structure is compatible with the relational data model. The tool proposed by the author significantly streamlines the process of analysing employee participation in the implementation of projects by eliminating the necessary, yet laborious, manual report generation activity. The required report can therefore be generated automatically once the input data have been entered. In its basic form, the tool does not require using the VBA code, and as such, it raises no doubts as to safety aspects. All of data processing, in terms of both transferring them to appropriate tables and generating the final report, is executed by means of formulas which make use of standard built-in functions available in the MS Excel spreadsheet.

The selection of the functions applied to this end is relatively narrow, comprising AND(), COLUMN(), COLUMNS(), CONCAT(), COUNTIF(), IF(), IFERROR(), INDEX(), ISEVEN(), MATCH(), MAX(), MOD(), OR(), RANK.EQ(), ROUNDDOWN(), ROW(), SMALL(), SUM(), and XMATCH(). The ultimate form of the report has been obtained using the applicable conditional formatting rules. However, in cases where the report is to be generated in a purely graphical form, the tool can be expanded with a relatively uncomplicated VBA procedure capable of generating that report on the basis of its current form.

Acknowledgements

The research presented in the article was supported by statutory work BK-274/ROZ1/2023 (13/010/BK_23/0072).

References

1. Allen, S. (2006). *Modelowanie danych*. Gliwice: Helion.
2. Jackson, M., Staunton, M. (2004). *Zaawansowane modele finansowe z wykorzystaniem Excels i VBA*. Gliwice: Helion.
3. Knight, G. (2006). *Analyzing Business Data with Excel*. Sebastopol: O'Reilly Media.
4. Kusztełak, P. (2020). *Microsoft Excel w pracy finansisty. Analiza i modelowanie danych finansowych*. Warszawa: PWE.
5. Oleński, J. (2001). *Ekonomika informacji. Podstawy*. Warszawa: PWE.
6. Oleński, J. (2002). *Ekonomika informacji. Metody*. Warszawa: PWE.
7. Próchnicki, W. (2012). *Zastosowanie Excels w pracy analityka finansowego, specjalisty ds. controllingu i analityka sprzedaży*. Gliwice: Helion.
8. Szczęśniak, B. (2017). Microtools Based on the Relational Data Model - representation of entities in a spreadsheet. *17th International Multidisciplinary Scientific GeoConference SGEM 2017, 17(21)*, ISBN: 978-619-7408-01-0, 29 June - 5 July, Albena, Bulgaria, pp. 447-454.
9. Szczęśniak, B. (2018a). Microtools Based On The Relational Data Model - transition from type II to type I of entity representation. *118th International Multidisciplinary Scientific GeoConference SGEM 2018, 18(2.1)*. ISBN: 978-619-7408-39-3, 02-08 July, Albena, Bulgaria, pp. 553-560.
10. Szczęśniak, B. (2018b). Microtools based on the relational data model - conformity between foreign and primary keys. *XV International Conference Multidisciplinary Aspects of*

- Production Engineering (MAPE)*, 1(1). ISBN: 978-83-65265-25-8, 5-8 September, Zawiercie, Poland, pp. 491-497.
11. Szczeńśniak, B. (2020a). Microtools based on the relational data model for improvement of information processes - multiplicity of relationships, and mandatory nature of relationships and attributes. In: K.S. Soliman (Ed.), *Education Excellence and Innovation Management: a 2025 Vision to Sustain Economic Development during Global Challenges*. Proceedings of the 35th International Business Information Management Association Conference (IBIMA), 1-2 April 2020 Seville, Spain. pp. 10537-10544.
 12. Szczeńśniak, B. (2021). MiRel concept-conforming tool for supporting service quality measurement by the SERVQUAL method. In: K.S. Soliman (Ed.), *Innovation management and information technology impact on global economy in the era of pandemic*. Proceedings of the 37th International Business Information Management Association Conference (IBIMA), 30-31 May 2021, Cordoba, Spain. pp. 12207-12218.
 13. Szczeńśniak, B., Molenda, M. (2013). *Spreadsheet application supporting the x-r control chart*. Conference Proceedings of the 22th Conference Modern Mathematical Methods in Engineering (3mi), ISBN: 978-80-248-3234-0, June 3-5, Horni Lomna, Czech Republic, pp. 128-134.
 14. Szczeńśniak, B. (2010). Arkusz kalkulacyjny w doskonaleniu procesu układania planu zajęć w szkole specjalnej. In: R. Knosala (Ed.), *Komputerowo zintegrowane zarządzanie, vol. II* (pp. 525-537). Opole: Oficyna Wydawnicza Polskiego Towarzystwa Zarządzania Produkcją.
 15. Winston, W.L. (2019). *Analiza marketingowa. Praktyczne techniki z wykorzystaniem analizy danych i narzędzi Excela*. Gliwice: Helion.
 16. Zasadzien, M., Szczeńśniak, B., Skotnicka-Zasadzien, B. (2017). Implementation of maintenance employees' work time scheduling. In: P. Limpaphayom, G. Huang (Eds.), *Proceedings of the Second International Conference on Economic and Business Management (Febm 2017), Shanghai, October 21-23, 2017*. Atlantis Press, pp. 226-231.