A Requirement Elicitation Process for BI Projects

Danielle A. Menéndez and Paulo Caetano da Silva

Abstract—To elicit requirements is one of the most important activities in the Software Engineering process. To understand and correctly specify what the user wants is an indispensable condition to ensure that the software meets its intended purpose. For BI projects, the elicitation of requirements is a task of great importance, since BI projects are based on service-oriented systems for the strategic areas of organizations, and so errors and flaws can produce serious consequences. Thus, this paper aims to discuss methods and techniques developed for the requirements elicitation activity. Therefore, a process of requirements elicitation focused on BI projects is presented. The proposed process defines stages and activities, the interactions between them, a set of input and output documents, as well as the most appropriate elicitation techniques for BI projects. In addition, a case study is presented. It was performed to validate the stages and activities of REP-BIP and to verify if the artifacts produced are consistent with the expectations for requirements elicitation for BI projects.

Index Terms—Software engineering, requirements elicitation, business intelligence, software process.

I. INTRODUCTION

To make assertive decisions in today’s competitive corporate environment can make the difference between achieving and maintaining leadership and losing market share. Decision makers can take action and follow paths from intuition; however, the use of automated tools becomes essential due to the exponential increase of data from several and dispersed sources. To organize and display this data to meet the perspective of executives is the purpose of Business Intelligence (BI) solutions. Turban et al. [1] claim that companies that fail to properly implement BI solutions are placed at a competitive disadvantage.

The BI solutions in Software Engineering are treated as information systems that have all their development cycle very well defined. Thus, the BI solutions also suffer the same problems related to the failure of projects that occur in other information systems. For over 10 years, half of BI projects had flaws [2] and this problem persists until more recent times [3]. In the developing process of BI solutions projects, especially in the phase of Requirements Engineering, many questions remain unanswered, resulting in projects that do not match the desired goals [4]. Projects of this nature require stakeholders with great critical/analytical skills and a technical team with good planning capacity and a strong power of identification and specification of requirements [5].

According to Sommerville [6], the process of discovering, analyzing, documenting and verifying what will make a determined system, as well as its restrictions, is called Requirements Engineering. Many are the critical factors for the success of software development projects; however, to elicit requirements correctly is the first activity to be performed so that a software system meets the needs of its users. In the Software Engineering literature many methods and techniques for requirements elicitation are found [6]-[8]. However, the information required for the specification of requirements is usually obtained from interviews, although there is good evidence that interviews are not the best way to extract user needs [9].

This work proposes a process of elicitation and management of requirements, which could be used by requirements engineers in their interactions with users, for the development of BI projects. The proposed process is an adaptation of Sommerville’s [6] process, which is one of the Software Engineering projects that are best accepted by the software community. The main difference between the two processes relates to the fact that Sommerville’s [6] process is generic and the process proposed for this work adds elements exclusively of BI projects, as the definition of data sources, data load process, Data Mart, among others.

II. REP-BIP

A. Influence of Related Work

REP-BIP (Requirement Elicitation Process for BI Projects), which can be seen in Fig. 1, had a strong influence after analyzing related works. The selected works indicated that there should be a defined process of requirements elicitation. To define the process, this work was based on the Software Engineering literature [10], [6] and [8] and on the analysis of the related work. Below a discussion on the related work found on this paper is presented.

Sutcliffe and Sawyer [11] indicate that prototypes are visual and intuitive artifacts that allow the user to simulate how the final application will be, influencing directly on the understanding of the requirements. To [12], the use of prototypes allows the handling of unstable conditions with more efficiency. However, [13] states that in some situations, such as projects whose requirements are volatile, prototypes should be elaborated from the beginning of the project, as it occurs in agile software development.

To both Carrizo et al. [9] and TDWI [14], the requirements must be appropriate for each type of project that is conducted. Carrizo et al. [9] discusses how different attributes, such as the experience of the person who identifies requirements and availability of user time on different factors (e.g. managing the problem, the process used, elicitor, etc.) lead to the use of a particular technique of requirements elicitation. TDWI [14]
presents 10 elicitation techniques and relates them to a set of project types (high level uncertainty, innovators, quick projects, etc.), indicating what techniques can be used in which project types.

Paim and Castro [15] and TDWI [14] works have influenced this work because they indicated that the elicitation of requirements should be directed to certain project types. These works influenced the decision that the classical process of requirements engineering should be adapted to BI projects.

B. The REP-BIP Process

The proposal of this process was built based on the identification of methods and techniques for requirements elicitation found in literature and they should be used to obtain satisfactory results, i.e. which meet the needs presented by stakeholders to implement BI solutions.

To Sommerville [6], the process of elicitation requirements is divided into four phases, which are: 1) viability study; 2) elicitation and analysis of requirements; 3) specification of requirements and 4) validation of requirements. These phases were used as the basis for the development of the REP-BIP process, but with special features, to adequate to BI projects.

To Paim and Castro [16], in BI projects the elicitation of requirements should always find out: i) facts and their properties; ii) the relationship between dimensions and facts; and iii) the sources which are providers of information. In addition, the requirements engineer must be aware of the efficient and rapid control of the change in requirements and provide a high quality of the documentation produced. Below, an REP-BIP model is presented with its phases, activities and tasks.

The ERP -BIP was divided into three stages: Planning, Development and Management of Requirements Changes.

C. The Planning Stage

The REP-BIP Planning stages consist of two activities, Viability Study and Elicitation.

To perform the Viability Study activity, the necessary inputs are user needs and the documentation of the field of application for which the BI solution will be oriented, e.g. entity-relationship diagrams and diagrams of the use cases of the systems in operation. This activity allows the identification of the hardware infrastructure and software that are needed, the allocation of resources, the initial estimative term and the possibility of delivery of the products requested by the user.

To White [17], four aspects should be part of the viability study in information systems projects: 1) operational; 2) technical; 3) schedule and 4) economic. The set of viability analyses of these four aspects is what will indicate whether it is viable to initiate the project or if there are alternatives that can be explored. REP-BIP adopted this set of aspects and its result will compose the Viability Document of the project.

The second activity of this phase is elicitation. REP-BIP adapted the requirements elicitation activity of the process model proposed by Sommerville [6]. The goal of this activity is to identify the initial requirements which are desired to develop a BI environment. The elicitation activity was divided into four tasks that are performed in a cycle that is ended when stakeholders accept the list of requirements and the non-functional prototypes created by requirements engineers.

The tasks of this activity are: 1) Discovery: the requirements engineer, taking as an input the documentation of the application domain which must find out the project needs. From these needs a list of necessary items for the
project will be produced and for each need the linked requirements; 2) Classification and Organization: it checks the duplicity of requirements, additional requirements or requirements that are not tied to a need in order to organize and classify the requirements; 3) Identification of Priorities: it sets the order that the requirements will be implemented and delivered; and 4) Creation of prototypes: they should be created to validate the correct understanding of elicited requirements.

D. Development Stage

The REP-BIP had its Development stage divided into two activities: Specification and Approval of Requirements. In the specification activity, the final version of the requirements document of the BI solution for the technical area and the functional prototypes are produced. However, in the Requirements Approval activity, users validate the prototypes built and the requirements engineer specifies the final version of the requirements document for the user area, which is an agreement between the parties containing the solution requirements.

The Specification activity consists of ensuring that the requirements defined are coherent with what was proposed by stakeholders, if there are no competing requirements or overlaps, or if there are new requirements that were not discovered in the Elicitation activity of the Planning stage.

In addition to specifying and discovering new possible requirements, the Specification activity develops functional prototypes to be used in visualizing the future implementation by stakeholders. Unlike the prototypes produced in the Elicitation activity, the prototypes created in this activity are functional, i.e. the user can visualize fictional data, navigate, simulate Drill-down or Roll-up operations, access menu options, among other features, illustrating, thus, the application in operation.

At the end of the Specification activity, two documents must be produced: the User Requirements Document and the System Requirements Document, each directed to a different audience.

The User Requirements Document is a high-level specification of requirements elicited with stakeholders, and a set of functional prototypes. This document does not contain technical details such as the dimensional model, setting of tables, and other aspects which are important to the development of BI solutions, which are not relevant to the understanding of the solution by users. The System Requirements Document is the one which contemplates the technical aspects of the project. This document serves as the basis for the other phases of the software life cycle, such as the design, the implementation and tests. The System Requirements Document is divided into eight sections: 1) introduction; 2) the system architecture; 3) corporate dimensional model; 4) specification of the Data Mart identified; 5) requirements specification; 6) non-functional requirements; 7) ETL process; and viii) glossary.

The second activity of the Development stage is the Requirements Approval. This activity aims to confront what was produced by the requirements engineer with initial requests from stakeholders. To approve the compliance of requirements, the artifacts produced in the Specification activity are used. The User Requirements Document is verified by identifying requirements faults of understanding. For each functional requirement the prototype developed must be approved. In addition to approving the requirement, the prototype will allow the observation of behavior, navigability and usability of what will be developed.

The non-compliance with one or more requirements, both in the User Requirements Document, as in the working prototype, generates an output document called Request for Change. This action will lead to a resumption of the Specification activity. This Specification-Approval of Requirements is ended when there are no non-compliances, when they do not generate changes or when they are accepted by the requirements engineer and stakeholders.

After the agreement between the parties involved, the requirements engineer should complete both the User Requirements Document and the System Requirements Document, ending the model of the requirements elicitation process.

E. Requirements Change Management Stage

The third stage of ERPM-BIP is Requirements Change Management. It is composed of three activities: Change Specification, Change and Cost Impact and Requirements Change. In this stage the monitoring of changes in the system requirements is performed, verifying the impacts brought by a new scenario modified by stakeholders.

The monitoring is performed within the tasks of the Elicitation activity and the Requirements Approval activity, in the Development stage. These are the times when it is checked whether the requirements produced represent the needs of the project. However, changes in requirements may be requested by stakeholders at any time, including poster phases of the life cycle of the software, for example, in the encoding. These events trigger the Requirements Change Management stage and allow the requirements to continue to be monitored even after the completion of the elicitation of system requirements.

Below a breakdown of each activity in the Requirements Change Management stage is presented.

The Change Specification activity starts with the identification of non-compliance with any requirement or a specific proposal for change from stakeholders. During this activity an analysis of the problem or the proposal is made, seeking to obtain details of the change and its viability.

After the analysis, the Requirements Change Document is produced. This document contains: 1) the specification of the original requirement; 2) requested changes; and 3) new needs and related requirements.

The activity of Impact and Cost of Change has as an input the Requirements Change Document, produced in the Change Specification activity. Such activity aims to verify the impacts that may occur in the project due to the requested changes.

The effect of the proposed change should be assessed using tracking information, as well as with its own requirements already elicited from the system. The initial schedule should be revised to reflect the scenario produced by the change, and the cost should be estimated based on the changes in both requirements documents and in projects in more advanced stages of the software life cycle, also in the design and
implementation of the system. After completing the analysis of the impact and costs, the findings contained in the Requirements Change Document must be presented to stakeholders to decide whether to approve the change, taking into account the change of scope, time and cost of the project.

F. Artifacts Produced in REP-BIP

The artifacts produced are, with the exception of the Questionnaire, documents describing their purpose and their use within the process. Below each of these artifacts is discussed:

- **Questionnaire**: basic set of questions to assist the requirements engineer in the interaction with users;
- **Project Viability Document**: a document that aims to make an analysis of the setting for the project and to check the possibility of its implementation;
- **Project Vision Document**: a document that provides a comprehensive overview of the project;
- **BI Solution Requirements Document**: a document with the set of requirements elicited in the Elicitation activity;
- **User Requirement Document**: a document produced in the Specification activity, with the review of all requirements of the project and which will constitute the contract between the technical area and the users;
- **System Requirements Document**: a technical document with the aim of providing information to the other stages of the software life cycle;
- **Requirements Change Request**: a document that requests changes to Requirement changes, produced in the Elicitation activity;
- **Requirements Change Document**: a document produced in the Requirements Management stage, due to the need for change in requirements that may bring impacts to the project;
- **Prototypes**: prototypes of interfaces for initial discussion with users, produced in the Elicitation activity;
- **Functional prototypes**: prototypes made with the development of software tools to show the functionality of the software, produced in the Specification stage.

G. Difference between REP-BIP and Generic Processes of Requirements Elicitation

As mentioned in the introduction, the REP-BIP was based on the generic processes of Software Engineering, mainly Sommerville's [6] and Pressman's [8]. However, there are several points targeted exclusively for BI projects:

- Check if the defined architecture meets the requirements, as well as if the development tool has the necessary resources for the implementation of what was requested;
- The analysis of the source system(s) documentation to find out which sources of data together with the ETL process is a factor explored only in BI projects;
- Defining the Data Mart, usually inserted in the software design stage, was brought to the requirements elicitation stage because it is important to determine if all requirements can be met with the specified Data Mart. In addition, the integration of the Data Mart with the corporate Data Warehouse helps prevent the duplication of the dimensions of the model;
- The definition of aggregation rules helps the subsequent stages of the life cycle to know which and how attributes can be aggregated. Equivalences of attributes also helped the ETL processes.

All these elements were incorporated into the REP-BIP System Requirements Document with the goal of helping the design and implementation stages, since in these stages it is possible to discover, later, that some requirements will fail to be delivered as requested, due to, for example, the lack of source data, or the ETL processes were badly dimensioned and the schedule will have to be modified.

With these differences to other processes, it is expected that the REP-BIP may not just be assertive in the requirements elicitation, but that it may also be decisive for the other stages of the project.

III. CASE STUDY

A. Monitoring Stages

Aiming to evaluate the REP-BIP proposal, a case study applying the specification process and requirements management of a BI project for a language school was performed.

The project aims to create an academic database for the decision making of the managers of the school, allowing actions to be performed for some critical areas such as financial, academic and marketing. The school hired a company specialized in BI solutions and the requirements engineer used the REP-BIP in the requirements elicitation.

The strategy used by the contractor was to divide the project into short sections, with fast delivery to the customer. This decision ended up affecting the phase of requirements elicitation, since a smaller set of needs was elicited, specified and managed. The school decided that the first part of the project would be directed to the evolution of the number of students enrolled in academic periods. This evolution includes: 1) the retention of students; 2) attracting new students and 3) the student’s decision to cancel the registration or even leave the school.

To initiate the project the stakeholders involved were brought together with the technical staff of the contractor and the head of the IT staff of the school. At this meeting it was explained how the elicitation and requirements management with the REP-BIP model would be and what the steps of the project, implementation and implantation of the BI solution would be like. After this first meeting, the team was put together: the academic director, the requirements engineer, the IT manager of the school. The responsibilities of each member was defined as well.

With all the necessary supplies, the Viability study began. The outputs of the study activity were: Project Viability Document, Vision Document and Initial Set of Needs. All outputs were produced from the templates defined in the REP-BIP.

To begin the Discovery task, the interview technique was used with the academic director, applying a questionnaire of 21 questions, which are present in the REP-BIP template. The output of the Discovery task was a List of Elicited Needs, which served as input for the classification and organization tasks, where the requirements were grouped. This set of requirements was presented to the academic director, in the
Priority Identification task, to perform their verification and prioritization. The next task was the creation of prototypes, where non-functional prototypes were built with a tool called Balsamiq Mockups (http://balsamiq.com/products/mockups/). This set of requirements and prototypes was presented to the academic director who validated the work realized, requesting not very significant modifications, so the requirements engineer did not find it necessary to start the requirements management for change control. Table I shows one of the requirements elicited by the academic director.

The output of the Elicitation activity was an Initial List of BI requirement, which served as input to Development stage. In the development stage a cycle between Specification activities and Requirements Approval was carried out. In these two activities, the IT manager at the language school participated in the process by providing various inputs related to the relational model and the documentation of the academic system.

### Table I: Requirement Discovered in the Elicitation Activity

<table>
<thead>
<tr>
<th>Name</th>
<th>Priority</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Show the evolution of the students enrolled over the past five years.</td>
<td>High</td>
<td>Functional</td>
<td>Show the number of students enrolled in the school in the last five years, allowing the application of filters.</td>
</tr>
</tbody>
</table>

#### Additional resources:
- The system must allow grouping or detailing the information by pre-established criteria.
- The system must allow the exchange between rows and columns of the data matrix.
- The system must allow filtering the data by day, week, month, semester and year.

The output of the Elicitation activity was an Initial List of BI requirement, which served as input to Development stage. In the development stage a cycle between Specification activities and Requirements Approval was carried out. In these two activities, the IT manager at the language school participated in the process by providing various inputs related to the relational model and the documentation of the academic system.

### Table II: Location of the Extraction of the Stage Dimension

<table>
<thead>
<tr>
<th>Attribute Source</th>
<th>Origin Attr.</th>
<th>Register file</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cd_stage</td>
<td>Stages</td>
<td>Stages.txt</td>
</tr>
<tr>
<td>Description</td>
<td>Stages</td>
<td>Ds_stage</td>
</tr>
<tr>
<td>Type_student</td>
<td>Stages</td>
<td>Tp_category</td>
</tr>
<tr>
<td>Class</td>
<td>Classes</td>
<td>Cd_class</td>
</tr>
</tbody>
</table>

In the loading process the periodicity of each dimension and of the fact table was set. The Stage and Motive dimensions had a single periodicity; the teacher dimension was defined as biannual load, the student dimension was set to monthly periodicity and time and fact dimensions have weekly load.

For the User Requirement Document, the requirements engineer had to set the requirements with a higher level of detail, and develop, together the IT department, functional prototypes, and an example can be seen in Fig. 3.

Although REP-BIP does not define, in the Planning stage, the order to build the User Requirements Document and the System Requirements Document, also able to be performed simultaneously, in practice it proved to be better to create the System Requirements Document initially, since it was possible to discuss, with the IT staff, the requirements elicited together with the non-functional prototypes, to elaborate the conceptual model of the Data Mart, which can be seen in Fig. 2. The Data Mart was called Evolution and beyond the fact table, the following dimensions were identified: 1) time; 2) stage; 3) student; 4) teacher and 5) motive.

Besides the Data Mart, the ETL process (Extract, Transform and Load) was also defined. Table II shows the source from which the data were extracted for the stage dimension. Filters needed for the process were also specified. Specifically for the stage dimension, only the stages with status = "A" (active) are the ones that should be extracted. The extracted data will be stored in a text file called stages.txt.

The Transformation process of the case study was simplified, since the database was migrated in the change of the academic system, which took place less than two years ago, and there were not overlaps, mappings, missing data or the equivalence of attributes.

### Table III: Details of the Requirement Performed in the Elicitation Activity

The details of the same requirement in Table I, performed in the Elicitation activity, may be seen in Table III, this time to compose the User Requirements Document.

In addition to functional requirements, some non-functional requirements were elicited together with stakeholders, such as:
- The system must be web;
- The system must be able to export the data displayed to
Excel spreadsheets and PDF files;
• The time for each query must not exceed 10 seconds.

The stakeholders of the project requested a few changes in requirements, and, for those which were made, there was no need to start the Requirements Change Management, since they were only changes in layout and the placement of columns in the tables of the functional prototype, which would not generate impacts to the schedule and the budgeting of the project.

TABLE III: APPROVED REQUIREMENT IN THE REQUIREMENTS APPROVAL ACTIVITY

<table>
<thead>
<tr>
<th>Name</th>
<th>Show the evolution of the students enrolled over the past five years.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority</td>
<td>High</td>
</tr>
<tr>
<td>Type</td>
<td>Functional</td>
</tr>
<tr>
<td>Description</td>
<td>The system must show a chart with the evolution of students enrolled for the last five years, grouped by semester. It must also show the data in a chart. The user can click to see the details of a given semester. The detailing illustrates the number of students enrolled in each stage of the course in that particular semester. It will be also possible to filter the data by day, week, month and year, indicating the start and end period. The user must have permission to access the system.</td>
</tr>
<tr>
<td>High-level UML Diagram</td>
<td><img src="http://www.evolutiao.matriculas.html" alt="Diagram" /></td>
</tr>
</tbody>
</table>

The Specification activity was terminated with the delivery of its outputs to the project stakeholders: 1) User Requirements Document; 2) System Requirements Document and 3) the sources of functional prototypes. The completion of the Specification activity ended the elicitation of the requirements of REP-BIP.

B. Benefits of the REP-BIP for the Case Study

This work followed the project in the requirements elicitation and project stages, and also a part of the implementation stage, since the project had not yet been fully implemented before the completion of this work.

During these steps, it was possible to note that the REP-BIP realized what was expected, specifying the requirements and showing stakeholders what the application would be like after its implementation and, in addition, the fact that the academic director and the IT staff of the school were satisfied with what was produced by REP-BIP infers that the proposed method succeeded in elicitation of system requirements in its relationship with stakeholders.

On the other hand, analyzing the benefits generated by the documentation for the technical area, the requirements engineer was present in technical meetings of the company hired to build the software and could see that the System Requirements Document was enough for the IT staff to develop all the part referring to the dimensional model, that is, the Data Mart and the ETL process, besides the development of the features specified in the conditions, also implying that the REP-BIP was successful in the specification produced for the technical area.

The requirements engineers also accompanied the development of the implementation of elicited features and found that there was no need for further consultations with stakeholders, which is a fairly common practice in software development projects, when doubts occur in the elicited requirements, which implies the elicitation process was successful.

Thus, the use of the REP-BIP in the case study was decisive in the following points:
• Better interaction between requirements engineers and stakeholders;
• Verification by the stakeholders of the appearance and functionality of the application;
• Correct understanding of the technical needs of the area described by the stakeholders.

IV. CONCLUSIONS

This study aimed to define a requirements elicitation process, which is a recurring activity in software development. The related studies found were the basis for the definition of the ERP BIP process, for the elicitation of requirements that are exclusively BI solutions-oriented. This process consists of three stages, seven activities and four tasks. The proposed process REP-BIP is composed of questionnaires, interviews and prototyping techniques, which will be used in the Viability Study, Elicitation and Specification activities.

With the definition of the REP-BIP process, it is hoped that the elicitation of requirements for BI projects will be more faithful to the concerns of stakeholders, reducing the difference between what is envisioned by the user and what is delivered by the development team. With the development of more appropriate techniques, it is expected to facilitate the work of requirements engineers in their interactions with users. Furthermore, the definition of templates for each of the proposed documents also helps to standardize what is produced in the proposed model.

The case study helped consolidate the proposed model, since interactions among stages, activities and tasks were checked and minor adjustments were made in the templates defined to improve the readability of the document, as well as incorporating common elements in the industry that are not routinely found in the academic environment.

To consolidate the proposed process it is necessary to apply it in other case studies of different sizes, which have other data sources or that have greater volatility of requirements. Another future work would be to extend the REP-BIP to act in other stages of software development, defining a complete process of developing BI projects.

REFERENCES


TDWI, “TDWI requirements gathering - Getting correct and complete requirements for BI systems,” 2013.


Danielle A. Menéndez is pursuing her master’s degree in Universidade Salvador, Unifacs. She is a member of the Research Groups Software engiNeering and its applications of Universidade Salvador and Research Group Education, Information Technology and Cyberculture (GPECTIC/CNPq).

Paulo C. da Silva has a Ph.D. in computer science, Universidade Federal de Pernambuco. He is currently a professor in the master's program in computing and systems. He is an systems analyst at Central Bank of Brazil and a member committee of XBRL Certification Board, XCB, U.S. He is an author of the book “XBRL – Conceitos e Aplicações”, 2006, co-author of the chapter “A framework for modeling and generation of web service oriented context aware systems”, 2011, and author of the chapter “Multidimensional data analysis based on links: Models and languages”, 2014.