How do Colombian software companies evaluate software product quality?

Wilder Perdomo, Julia Prior, and John Leaney

School of Computer Science
Faculty of Engineering and IT
University of Technology Sydney

Objectives | Research Question

We present a study carried out with seven young software development companies in a developing country that examine:

1. How they evaluate software product quality (SPQ)?

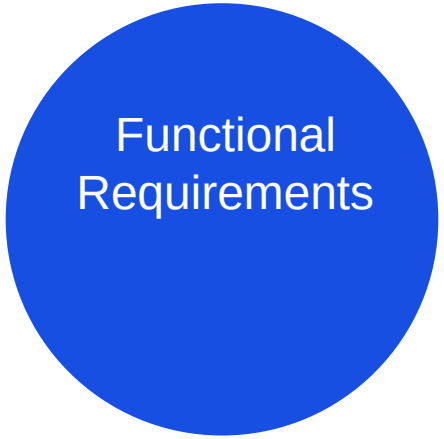
2. Which characteristics and measures they apply in their companies?

PROBLEM

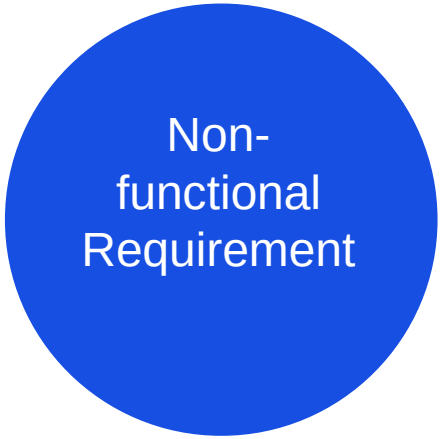
Low competitiveness and SPQ evaluation affects the young software development companies that represent 19% of the 5900 companies in the software and ICT industry in Colombia

Software Quality - Definition

'Capability of a software product to satisfy stated and implied needs when used under specified conditions' (ISO24765 2010).



Specify the capabilities that the solution must provide to its customers and the final users. It defines the functionality the solution needs to work (Inayat et al. 2015).



It is used to delineate requirements focusing on 'how well' software does something as opposed to functional requirements, which focus on 'what' the software does (Chung & do Prado Leite 2009).

SLR - Results

Some studies on issues of software product quality (SPQ)

Description	Author(s)	Year
Usability and Accessibility as Quality Factors of a Secure Web Product	Baquero et al.	2018
A New Model Based on Soft Computing for Evaluation and Selection of Software Products	Fernandez et al.	2018
Diagnostic on the Appropriation of Metrics in Software Medium Enterprises of Medellin City	Metaute and Serna	2016
Software Reliability Modeling Based on ISO/IEC SQuaRE	Febrero et al.	2016
A SQuaRE-Based Software Quality Evaluation Framework and Its Case Study	Nakai et al.	2016
Evaluation of Software Product Functional Suitability: a Case Study	Rodriguez et al.	2016
A Framework for Evaluating the Software Product Quality of Pregnancy Monitoring Mobile Personal Health Records	Idri et al.	2016
Assessment of Quality Factors in Enterprise Application Integration	Kumar et al.	2015
Improving Software Product Line Configuration: a Quality Attribute-Driven Approach	Guana and Correal	2013
A Model for Measuring Agility in Small and Medium Software Development Enterprises	Escobar and Linares	2012
Software Quality Modeling Experiences at an Oil Company	Lampasona et al.	2012
Certification Process and Product Quality: Route Colombian SME Manufacturing Software	Pelaez et al.	2011

SLR - Results

Measures to improve software product quality evaluation

Description	Author(s)
Functional suitability: Functional coverage, Functional correctness, Functional appropriateness of usage objective	Heck et al. 2010, Alvaro et al. 2007, Yahaya et al. 2008
Performance efficiency: Response time adequacy, Mean turnaround time User access capacity	Burger & Reussner 2011 Yahaya et al. 2008, 2010
Reliability: Fault correction, System availability, Redundancy of components	Febrero et al. 2016, Yahaya et al. 2008, 2010, Carvalho et al. 2009, Morris et al. 2001
Security: Data integrity, Authentication mechanism sufficiency, Authentication rules conformity	Izzat 2013, Mellado et al. 2010, Yahaya et al. 2010
Maintainability: Coupling of components, Cyclomatic complexity adequacy, Reusability of assets, Diagnosis function sufficiency, Modification correctness, Test function completeness	Duque et al. 2011, Rodriguez, Pedreira & Fernandez 2015, Serebrenik et al. 2010, Baggen et al. 2012, Izzat 2013, Marcos et al. 2008, Carvalho et al. 2009, Yahaya et al. 2010
Usability: Description completeness, User guidance completeness, Operational consistency, Monitoring capability, Appearance aesthetics of user interfaces	Bevan et al. 2016, Heck et al. 2010, Izzat 2013
Portability: System software environmental adaptability	Yahaya et al. 2010

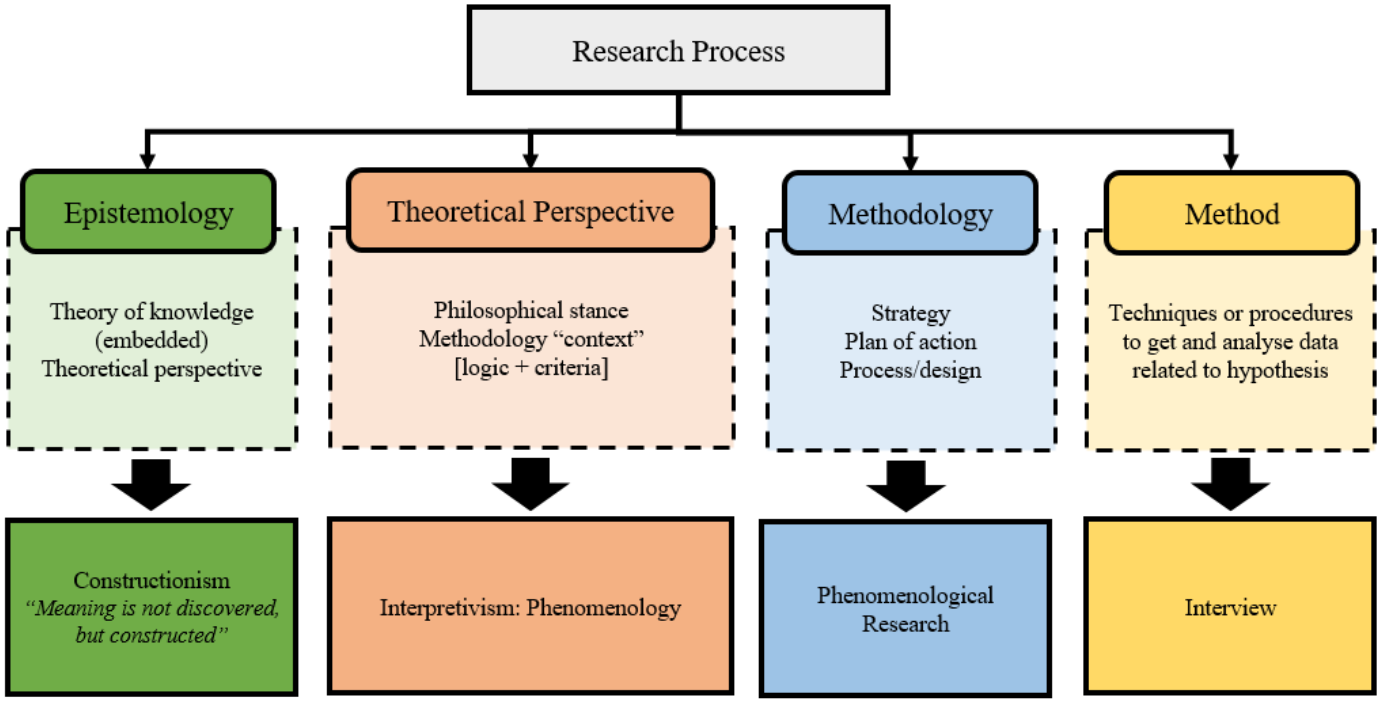
SLR - Results

Characteristics and attributes measured by Colombian software companies

Description	Author(s)	Year
Usability: Operational consistency; **Reliability:** Test coverage, System availability	Castellanos	2016
Compatibility: Co-existence with other products, Data formats exchangeability	Delgado et al.	2016
Maintainability: Cyclomatic complexity adequacy	Giraldo et al.	2014
Usability: Operational consistency; **Maintainability:** Coding rules conformity	Farah et al.	2014
Maintainability: Reusability of assets	Guana and Correal	2013
Functional suitability: Functional correctness, Functional appropriateness of usage objective; **Compatibility:** External interface adequacy; **Usability:** Appearance consistency; **Reliability:** Redundancy of components	Lampasona et al.	2012
Maintainability: Coding rules conformity	Pino et al.	2012
Maintainability: Coupling of components	Alba and Hurtado	2011
Usability: User guidance completeness; Maintainability: Cyclomatic complexity adequacy	Arciniegas et al.	2010
Usability: Functional customizability; **Portability:** Operational environment adaptability; **Maintainability:** Reusability of assets	Diaz et al.	2010

Research Design

1. Quantitative research approach (closed questionnaire)
   a. Pilot study (11 people, 4 companies)
   b. Results were not useful for the research
2. Qualitative research approach (semi-structured interview)
   a. We were interested in what they did understand and why? “new approach”
   b. 20 employees (developer, tester, project manager) from SMEs [young companies]



Research Process, adapted from Crotty (1998)

Validity

Data saturation

The depth of the data is expressed in terms of quality, as the richness of the data.

Validity of Analysis

For all the patterns, we considered saturation had been reached when the authors agreed that no new or relevant data emerged regarding any patterns.

Method validity

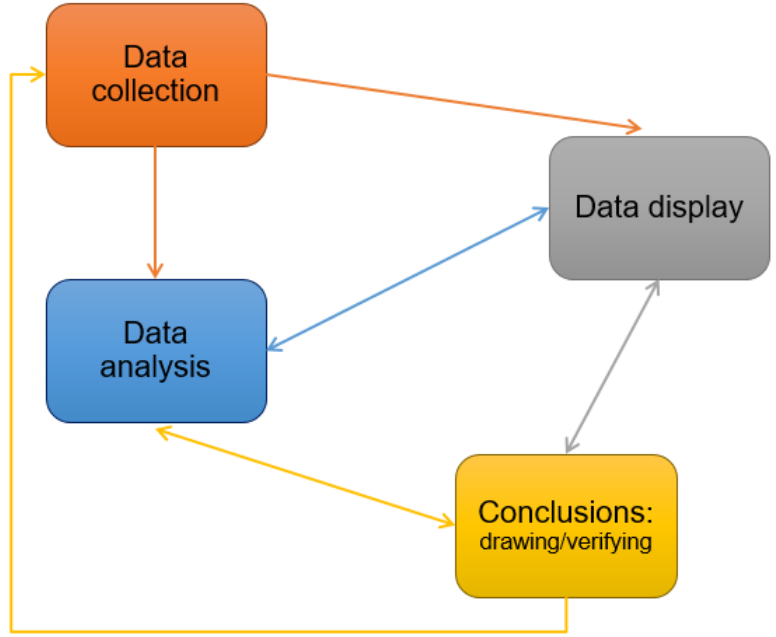
We adopted a well-researched iterative method. This uses iterations between the collection, data analysis and conclusions drawn.

Iterative Method Validation

1st author did the initial coding, and preliminary patterns. The coding and patterns were reviewed and refined by 2nd and 1st author. The 3 authors reviewed the patterns and developed the overarching theme together.

Data Collection

Participants classification	
Size	Small (2 years of creation) Medium (3-6 years of creation)
Role	Developer, Tester, and Project Manager
Market	Local, National, and International
Industry	Services, financial, agroindustry, education, automotive, food, manufacturing, private sector, government, aeronautics, energy, telecommunications, agriculture, all types of industries



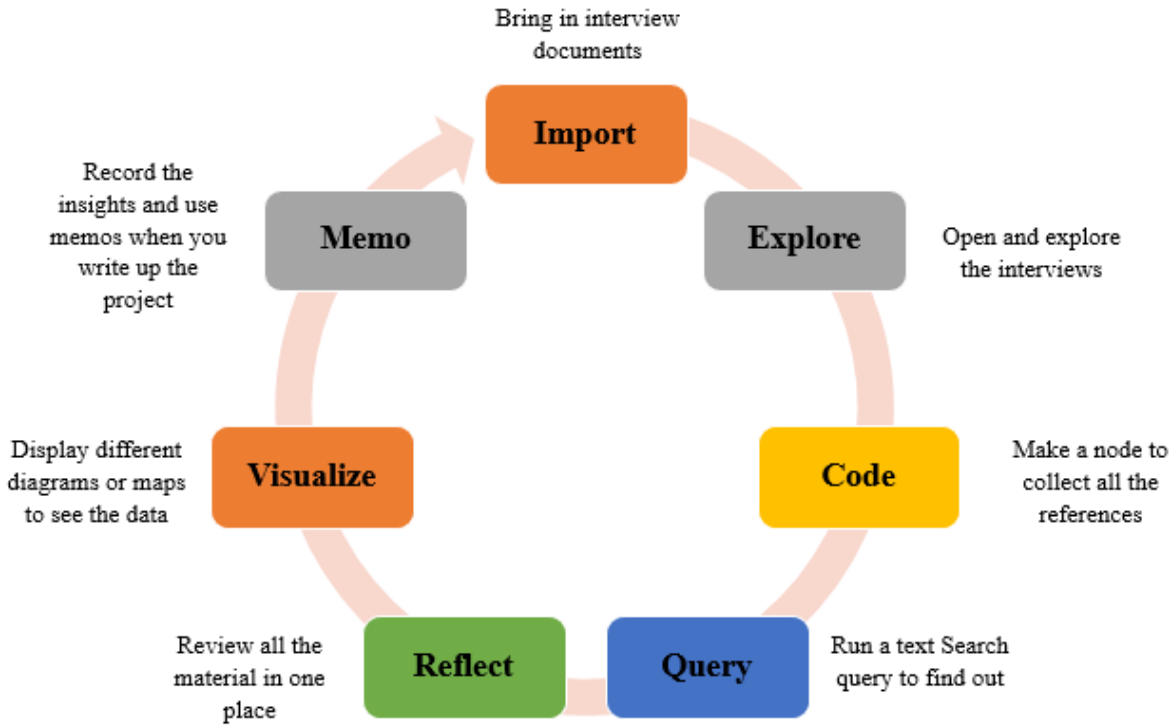
Components of data analysis. adapted from (Bazeley 2013, Miles et al. 2014)

Criteria to select participants:
i) they had to be employees for a small (11-50 employees) or medium (51-200 employees) enterprise (SME), and
ii) they had to have at least two years of professional experience working in a software development area, i.e. “the experience of developing software”.

Thematic Analysis

We performed both a manual analysis and a semi-automatic analysis using NVivo.

- Coding
- Codes, nodes, and pattern creation
- *Participant names were anonymous, and their names are unidentified*
- 12 patterns identified initially, which were merged to form 10 patterns



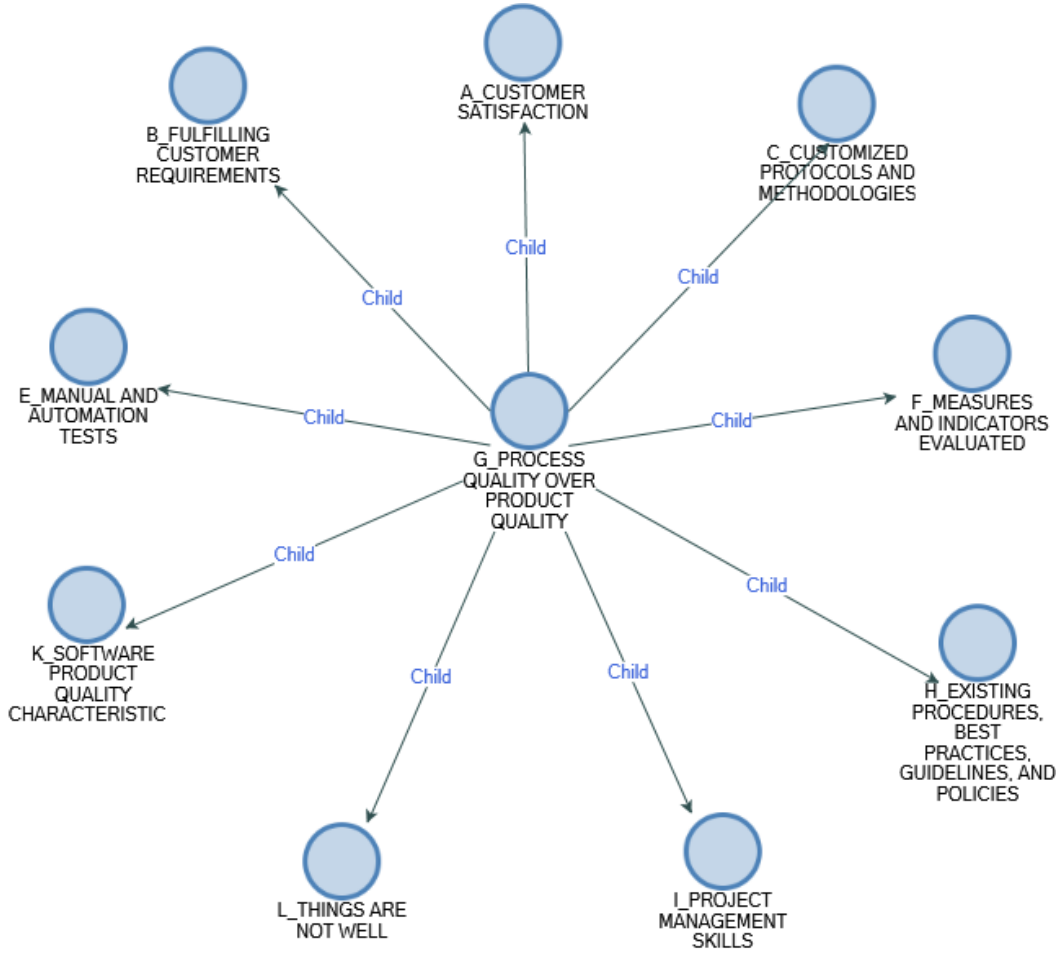
Path for exploring data in NVivo
(Alase 2017, Pietkiewicz & Smith 2014)

Findings and discussion

- 80% of the interviewees understand that they need to attend to quality in their companies but are focused only on process quality.
- Only a few of the interviewees appear to have a good understanding of SPQ.
- Half of the 20 participants confuse product quality with process quality. These participants ‘overlaid’ the idea of product quality with process quality, i.e. they talked about product quality as if it was process quality.
- As a consequence of this confusion, they do not recognize that they are not, in fact, measuring product quality.

Patterns discovered

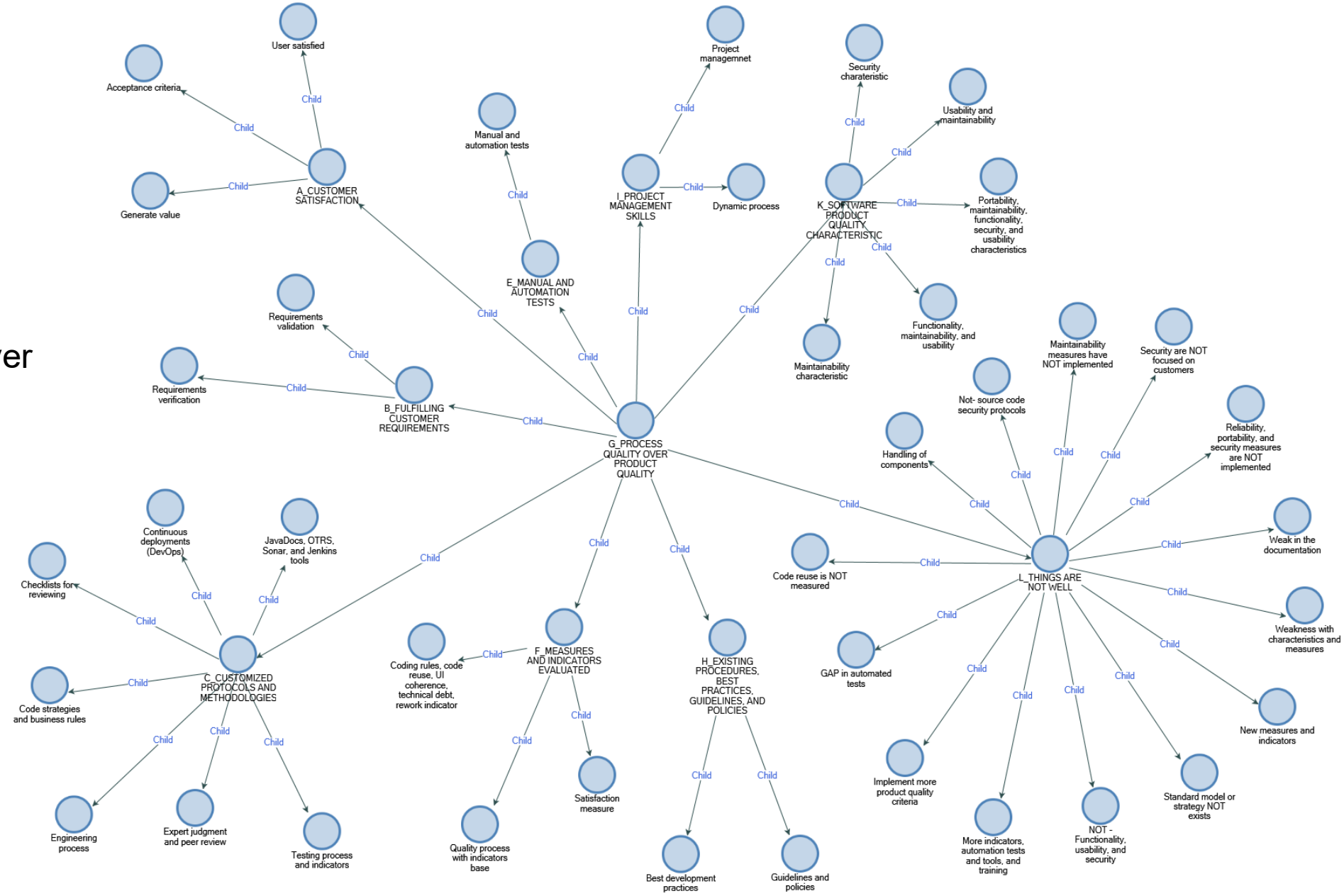
List of patterns	
Customer satisfaction	Process quality over product quality
Fulfilling customer requirements	Existing procedures, best practices, guidelines, and policies
Customized protocols and methodologies	Project management skills
Manual and automation tests	Software product quality characteristics
Measures and indicators evaluated	The quality environment is not well



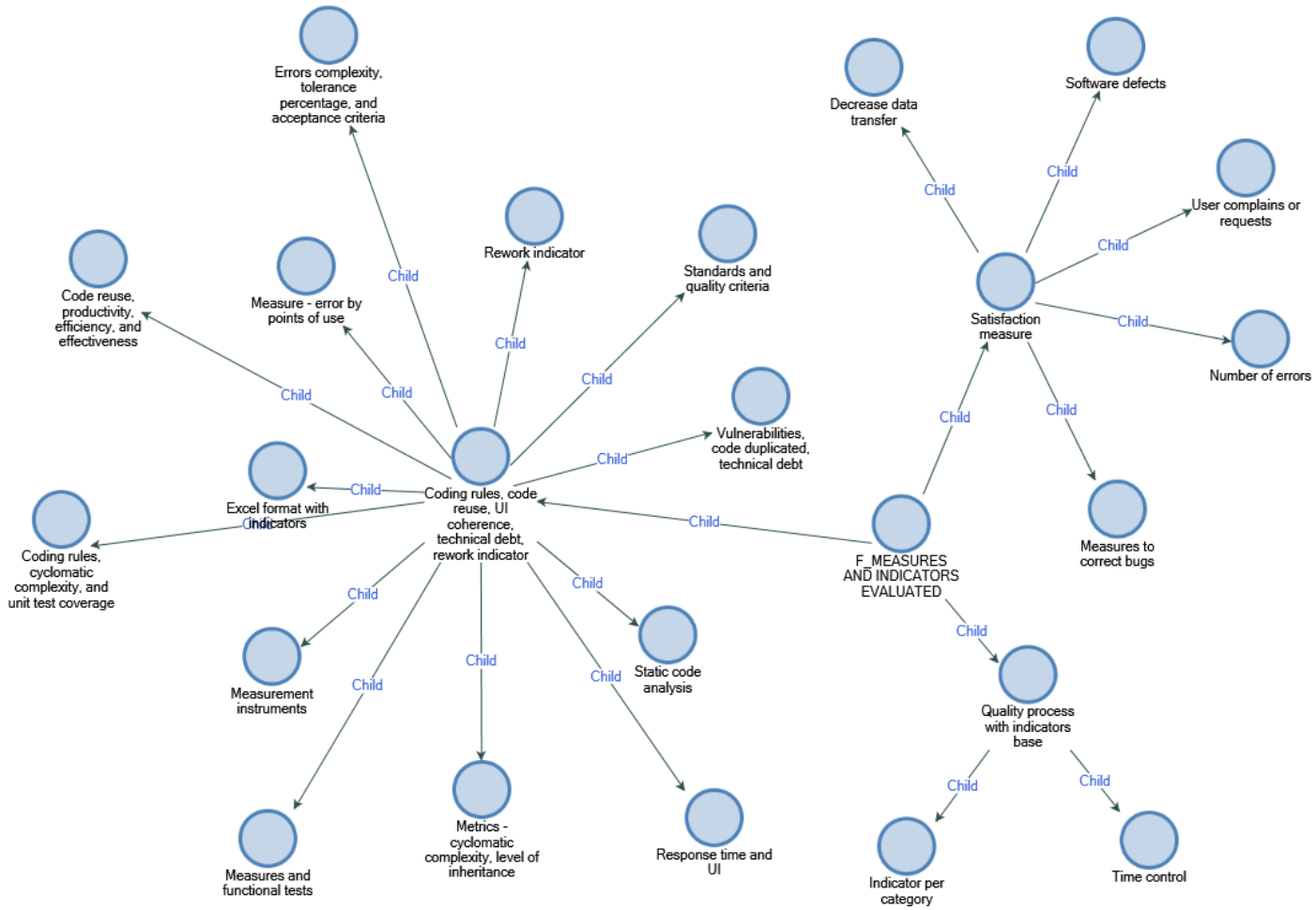
Patterns discovered using NVivo

Theme

"Process Quality over Product Quality"

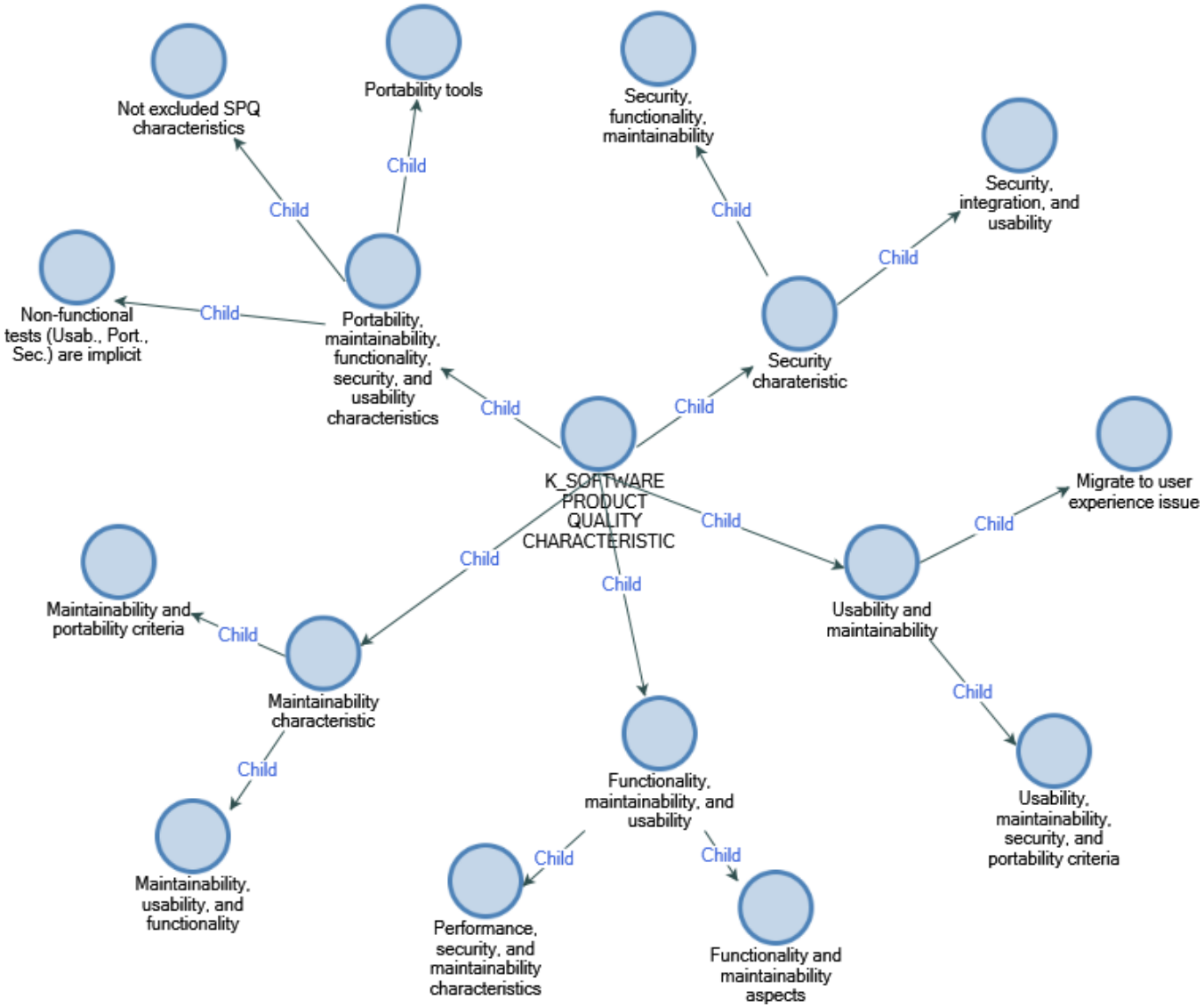


Measures | Indicators	
Effectiveness of defect elimination	Product
Deliveries without critical bugs	
% Protocol delivery deviation	
Response time for the performance case in different scenarios: peak loads or basic usage load of the application	
Percentage of application airtime in a specified time period	
Findings related to security (number of problems found in the application)	
Indicator of duplicate lines of code	Code Quality
Indicator of unit tests	
Indicator of the source code complexity	
Periodic behaviour of maintenance cases	



Pattern. Measures and indicators evaluated

Measures | Indicators	
% Changes without commitment guidance	Process
% Reverted changes	
Efficiency of peer reviews	
Quality of installers	
Rework cost	
Volatility of requirements	
Quality of requirements	
Effectiveness of planning	
Effectiveness of the development test	
Actual percentage of progress	
Prioritization of critical bugs	
% Changes without responsibility	
% Changes without tracking	
% Effort deviation in protocol development	
Cost of reworking protocols	
% Bugs rejected	
Stability of requirements	
% Bugs reported in regression	



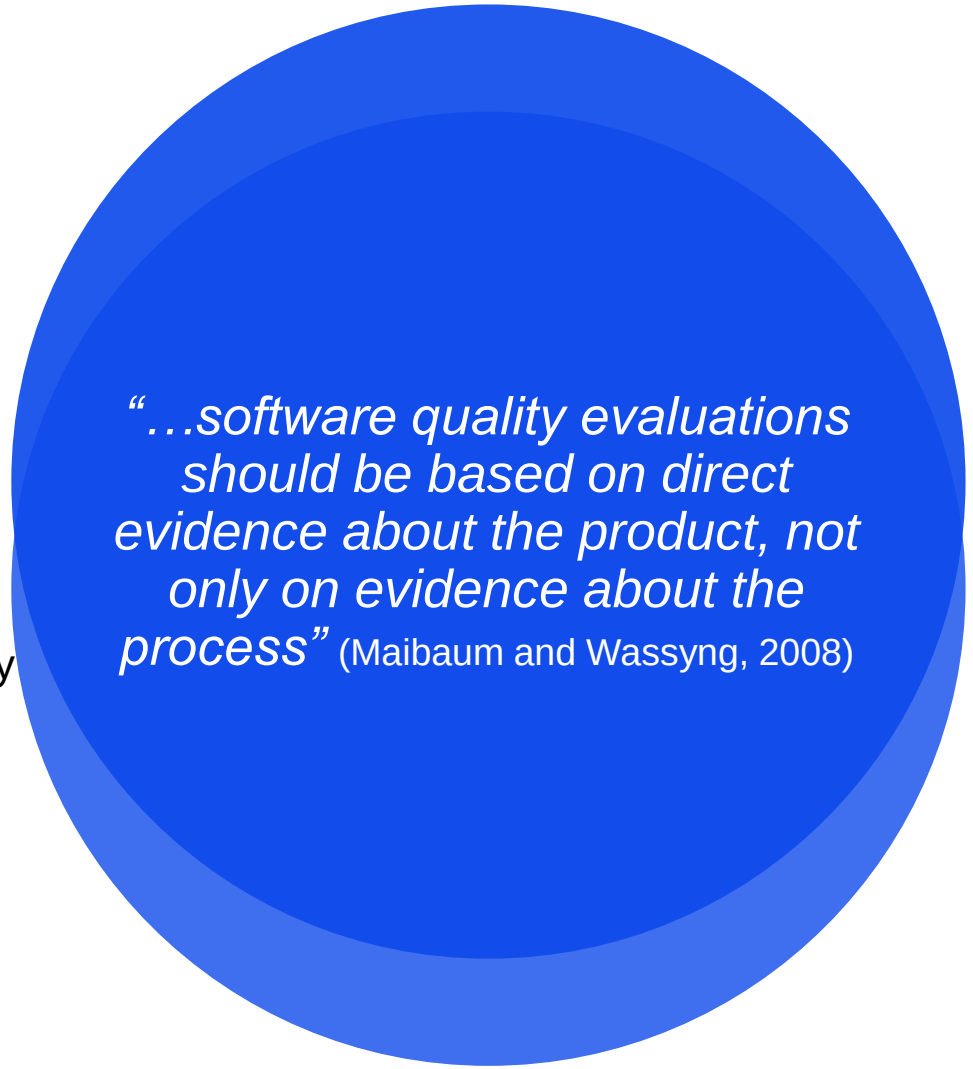
Pattern. Software product quality characteristics

Consequences and Benefits

Consequence...	Benefits…	
Colombian companies are not able to take advantage of the advances in software product quality (SPQ) because they do not understand that SPQ is different from software process quality	The candidate has gained an understanding of the first consequence and can deliver guidelines, education and community groups that can be set up to improve this situation	Companies need to focus on SPQ measures to ensure customer satisfaction
Developers need to establish a reviewing process iteratively for each SPQ criteria during the development stages (*reducing it helps reducing time, cost, and resources invested)*	A standard and economical model is required for having an environment with flexible measures and procedures and avoiding such confusion *(technical concepts: measures/characteristics/indicators)*	Training education program for specific companies to understand the differences between SPQ and software process quality

Conclusions

- Understanding how young software development companies evaluate SPQ and which measures they apply in their companies to do so
- Using phenomenological research to investigate participants' experiences in software quality
- Demonstrating that young software companies in Colombia focus more on process quality than software product quality
- Software developers in these companies confuse product quality with process quality and even conflate the former with the latter
- Companies mistakenly believe that they are evaluating software product quality, when, in fact, they are not
- Implications for quality assurance of these companies' software products, and the companies' competitiveness and success



Future work

Propose an economical and flexible model for evaluating software product quality in young software development companies

Build communities of practice to share the meanings and criteria to consider SPQ as a key factor for software development projects

Replicate this study with other significant samples of young software development companies to do comparative studies on SPQ

THANK YOU

For more details.
Contact: Wilder Perdomo
wilder.perdomo@uts.edu.au

UTS CRICOS 00099F