IBIMA Publishing
Communications of International Proceedings
https://ibimapublishing.com/p-articles/44SE/2024/4439724/
Vol. 2024 (24), Article ID 4439724

Practices of the Software Industry in Chile: An Exploratory Study*

Andrés ALFARO and Dante CARRIZO

University of Atacama, Copiapó, Chile

Correspondence should be addressed to: Andrés ALFARO, andres.alfaro@uda.cl

* Presented at the 44th IBIMA International Conference, 27-28 November 2024 Granada, Spain

Abstract

This article presents a comprehensive investigation on the practices of the software industry in Chile, aimed at gaining a deep understanding of its processes, tools, and methods, as well as the factors influencing success, risk, and failure in software product development. In response to the scarcity of detailed studies in this field, the study employs an ethnographic approach, gathering first-hand data through a series of short stays in various software development companies across the country. The findings show that companies apply a set of best practices in their projects, with active client involvement highlighted as a critical factor for successful development. Additionally, common practices are identified, such as project estimation based on expert judgment, development teams generally consisting only of project managers and developers, and the use of meetings as the primary method for requirements gathering. The study also reveals the adoption of agile methodologies and the implementation of various tools to optimize product development. This empirical analysis provides a solid foundation for the development of future methodological strategies in the software industry in Chile, with implications that promote continuous improvement and an increase in project success rates.

Keywords: Software development, success and failure factors, industrial practices, ethnography

Introduction

Information systems have gained great importance worldwide during recent years (Beigl et al., 2017). This phenomenon has created new economic activity around the software industry that is considered to be globally promising (Álvarez & Lillo, 2008). This can be seen in Chile through data that show increases in the Information Economy sector during the past years (Fundación País Digital, 2017), representing 3.5 % of the total Chilean economy, with Information Technology (IT) constituting 27.6 % of the sector. The areas within IT are principally represented by information providers, software developers, data processing companies, and data centers. It should be noted that the IT sector has increased in size in relation to the overall Information Economy by 8.2 %, with a projected increase in the coming years (Fundación País Digital, 2017). Despite these performance indicators of the Chilean software industry though, there is insufficient information concerning the working styles or factors influencing the times in the development of software products. The existing few statistical analyses (GEMS, 2015) (GECHS, 2008) (Pereira et al., 2004) have not delved into the problematic, daily action and reality of the Chilean industry, due to limitations in how their data were collected. In these studies, information capture was done using generic questions or surveys that could be answered by anyone in the company, which can be affected by response validation, creating information capture deficiencies (Maul, 2017). It is possible to ascertain some background information, such as the scarcity of studies

Cite this Article as: Andrés ALFARO and Dante CARRIZO, Vol. 2024 (24) "Practices of the Software Industry in Chile: An Exploratory Study " Communications of International Proceedings, Vol. 2024 (24), Article ID 4439724, https://doi.org/10.5171/2024.4439724

about risk, success, and failure factors and the practices and processes that software developers have, highlighting the importance that this type of research can have for the industry (Ebert et al., 2016). These limitations in the literature are the motivation for this study, which examines how software development companies work, providing a general vision of methods, activities, processes, tools, and factors that condition software development, thus helping gain a general vision of their working way.

This study hypothesizes that software development companies in Chile follow a specific set of practices to develop their products and that certain key factors significantly impact the success of the software development process. The main objective is to gain an overview of the methods, activities, processes, tools, and factors that shape software development, thereby contributing to a more comprehensive understanding of how these companies operate. To achieve these objectives, a qualitative methodology based on an ethnographic approach was adopted. The study involved identifying and selecting software development companies in Chile and was structured into five stages: definition, fieldwork, data collection, proposals, and evaluation. During the fieldwork, ethnographic techniques were applied, enabling direct interaction with workers and observation of their work environments, providing a deeper perspective on their practices. The document is divided in the following manner. Section 2 presents a review of prior work related with studies of the industry. Next, section 3 presents the methodology used in the ethnographic study. Section 4 presents the results. Section 5 discusses the gaps, practices, and most outstanding factors and influences in Chilean software development. Section 6 describes the limitations of the study as well as future research directions. Finally, section 7 presents general conclusions from the research.

Prior Published Studies On Chilean Software Industry Software Development Practices

As described in the previous section, the broad statistics relating to the Information Economy sector are known (Fundación País Digital, 2017), (GEMS, 2015), but efforts related with how the Chilean software industry works, develops, and creates products are scarce (GEMS, 2015) (GECHS, 2008) (Pereira et al., 2004), (Rojas & Carrizo, 2017), (Vidal-Silva et al., 2017) and obsolete as (Press, 1991). Such scarcity is not the case in other countries (Barros-Justo et al., 2019) (Wang & Galster, 2018) (The Standish Group, 2013) (Rahikkala et al., 2018). One of the most important and relevant of such studies is CHAOS (The Standish Group, 2013), which gathers information about the IT environment in the United States. Within Latin America, there is work by Jiménez (Orantes Jiménez, 2012), Metaute and Serna (Metaute P. & Serna A., 2016), Casado Vela (Casado, 2010), and Elles and Triviño (Elles & Triviño, 2016) that use statistical results to demonstrate the practices, methods, and characteristics of the software industry in different countries. In Chile, the GECHS (GECHS, 2008) study sought detailed information about the conditions and characteristics of software firms, including elements like “Research and Development,” “Human Capital,” and “Processes and Tools,” among others. Another study developed risk analysis and mitigation models to help project managers identify potential risks early in the process, so that they could take corrective actions (Pereira et al., 2004). However, these studies were published in 2008 and 2004, respectively, and likely no longer provide useful, up-to-date information. There are, though, two more up-to-date studies. The first, by Rojas and Carrizo (Rojas & Carrizo, 2017) describes the adoption level of techniques and tools in requirements engineering within the Latin American software industry. The second study, by GEMS (GEMS, 2015), surveyed different Chilean software development companies in order to characterize the Chilean industry in terms of developers, projects, processes, and work climate, among other aspects. Finally, Vidal-Silva et all. (Vidal-Silva et al., 2017) analyzes the relevance of the performance, safety and reliability variables to develop quality software products and determines the software developed in Chile is a product that meets these variables. Apart from these few studies that look directly at the software industries themselves, there are tangentially related studies that describe characteristics of particular work practices within the software industry. Two examples of such connected research are the studies by Rahikkala et al. (Rahikkala et al., 2018) and Sehra et al. (Sehra et al., 2017), which present case studies and systematic reviews, respectively, of the characteristics and most commonly used methods when considering software products. In addition to having scant studies to draw from, the existing literature demonstrates a few shortcomings. These include such as not indicating that corrective actions are influential at the time of producing positive results in software product creation (Pereira et al., 2004), not delving into everyday development practices (GEMS, 2015), or being focused only on a specific aspect of development (Rojas & Carrizo, 2017) (Vidal-Silva et al., 2017).

In sum, the constant renewal of IT requires frequent studies of the sector in order to detect and understand the vital signs related with the software development industry. Furthermore, such studies should use more in-depth research methods that can help elucidate the current practices of software development and quality management.

Methodology

Detailed descriptions of the different stages and activities used to develop this research are found in Carrizo and Alfaro (Carrizo & Alfaro, 2017). Briefly, the methodology uses a qualitative approach, with an ethnographic technique at its core. It consists of five stages, with different activities in each stage, as shown in Table 1. During the Definition stage, software development companies were identified using information found on their web pages. Other types of contacts were made, such as via universities and professional societies, but they offered no concrete answers. From this collection, roughly 90 companies were identified, with different business areas related to information technology. An in-depth analysis whittled the selection to 36 companies, of which more than half responded positively to our request for a visit. However, visits were made to only 8 companies, due to a variety of limiting factors, including financial resources, travel distance, and scheduling conflicts. An open-ended interview protocol was prepared with questions that could help evaluate the study hypotheses. Since the information for analysis would be in the form of work narratives, all responses were recorded for subsequent analysis. In the second stage (Fieldwork), the participant companies were visited, and researchers interacted with workers by applying the interview protocol, participating in team meetings, and familiarizing themselves with the physical works spaces and some of the tools used in project development. In the third stage (Collection), the information from the audio files were transcribed to paper. Concepts and statements from each interview were then fragmented into individual phrases that were transferred to collaborative murals where they would be classified into categories of Risk Factors, Failure Factors, Success Factors, Work Style, Problems, Assessments, and Company Data. In addition, the narratives were used to diagram the software development process for each company, as described in by the participants.

Table 1. Proposed research methodology

Stages	Description	Steps
Definition	Seeks to explain and describe how the population or group will be studied, how contacts were made, the information to be sought, and the form or technique that will be used to collect the data.	• Select and contact informants • Define the type of data for collecting • Define data collection procedures
Fieldwork	Start the empirical study of the selected group or population using ethnographic study techniques.	• Conduct field work • Apply ethnographic techniques
Collection	Analyze the data via transcription, classification, and categorization of the gathered information.	• Reduce the data • Reconstruct the data
Proposals	Propose different possibilities and options that seek to improve different parts of the selected group or population. Note that improvements are based on the different critical aspects that affect the selected groups or area.	• Construct improvement alternatives
Evaluation	Obtain feedback and evaluate the efficacy of the proposals.	• Validate the improvement alternatives.

This diagram-building was guided by various “elements of analysis” (EoA), which are a set of concepts that allow for studying and examining the software industry in a far more transversal manner. It was possible to define a series of “tags” that divide and synthesize each element, in order to connect the responses with the hypotheses defined at the start of the project (Table 2).

Table 2. Elements of Analysis and Tags corresponding to each research area

Research area	Element of Analysis (EoA)	Tags
Demographics	Company	Organization
	Clients	Client
Software Engineering	Tools	Tools, Documentation
	Methods	Methods, Requirements, Testing, Development, Receiving,

		Estimation, Changes
	Processes	Project
	Quality assurance	Quality, Metrics
Factors	Success factor	F_success
	Failure factor	F_failure
	Risk factor	F_risk

The construction of different improvement proposals for critical elements found in the study is started in stage four (Proposals). Some improvements, actions, or activities have already been implemented by some companies. In the final stage (Evaluation), a report was created and sent to all participant companies for their involvement in the project. In addition, three of the companies were selected to receive retrospectives about the project, in particular regarding improvement proposals.

Results

As mentioned in the methodology section, this study involved visits to 8 companies nationwide. Of these, six had between 2 and 15 employees, while the remaining two had between 16 and 25 employees. It is worth noting that the number of employees varies depending on the number of projects under development in each company. During these visits, owners, project managers, analysts, and programmers were interviewed using a questionnaire structured into three types of questions. The specific questions covered aspects such as the number of employees, type of specialization, number of active projects, and their average duration. The general questions explored work methods, team composition, time estimation, development phases, and management and development tools, as well as quality management. Finally, the company's experience was investigated to identify success factors, causes of failure, risks, client relationships, and potential improvements for the software industry. The collected information is presented as a function of the different elements of analysis (EoA) measured, within the three larger research areas: Demographics, Software Engineering, and Factors.

Demographics

This research area treats data related principally with software development companies, highlighting the working environment and its relationship with the client. Most of the collected data in this area are related to the Company EoA, with 122 tags for “Organization,” which relates to each company’s organizational characteristics and includes attributes like number of workers, roles, level of rotation, training, and project type and quantity. In contrast, the information collected for the Client EoA is scarce, with only 26 tags. However, the Client tags were found in every interview, indicating that the client is a fundamental factor when developing a software product, despite the relatively low number of tags.

On the business front (Fig. 1), special attention is paid to personalized development, with Web Development and Maintenance of created systems being the most common. Note that business areas included Mobile Development and Collaborations (i.e., work with other companies) as well.

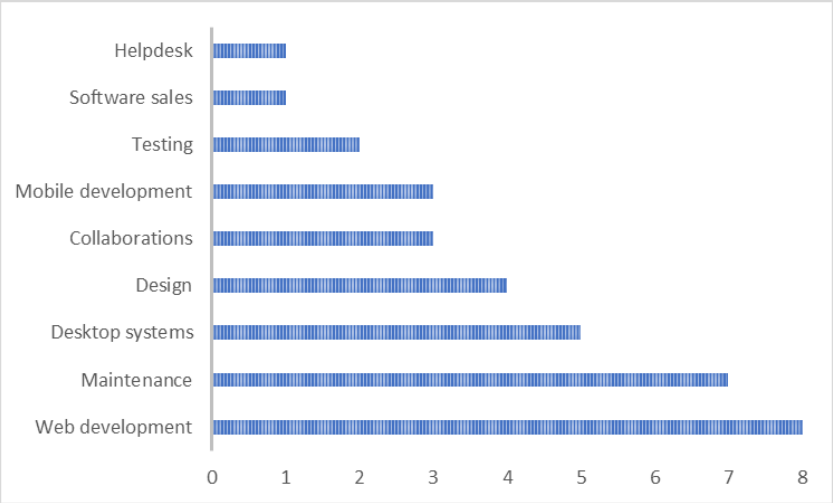

Figure 1. Areas of business.

Two organizational structures are highlighted during interviews. The first structure comprises two workers, with one taking the initiative to the client and the other developing the project. The second has more workers with identified roles, including General Manager, Project Manager, Technical Manager, Sales Manager, Project Manager, Developers, Designers, and Testers, among others. It is important to note that these roles are closely related to the organizational structures of each company; the more workers in a company, the greater the distribution of their roles. In most cases, the interviews indicated a good working environment, a horizontal organizational structure, highly qualified staff, collaborative and comfortable spaces, good communication between colleagues, and low levels of personnel rotation. At least three means to access project work were described, the most frequent being, "going out to look for projects" by presenting and offering services to clients. The second means was having a client contact or arrive at the company and requesting their services. The third most commonly described means was public bidding. Regarding the number of projects in development during the study, five companies were working on 5 to 10 projects simultaneously, two on 15 to 20, and only one was working on fewer than 4 projects. The interviews elicited information about the Client EoA, which describes the company-client interaction in different areas, such as meetings, testing, documentation, and estimates, with the participant companies indicating the relevant participation that clients have in project development, identifying such participation as a crucial factor in building a software product. According to these companies, clients have more involvement and influence in project development during meetings, project advancement, approvals, and product functionality requirements or approvals.

Software engineering

This research area describes the line of production that Chilean companies have in producing a software product. As seen in Fig. 2, the largest amount of data collected is in the Processes EoA, with 144 tags classified as "Project." This is because this concept includes information related to development strategies, client relationships, team structure and roles, project timelines, and project closures. In contrast, the least amount of collected data was from the Methods EoA, with only 4 tags classified as "Methodology," since most companies have no standardized or predefined way of working. As mentioned at the beginning of this section, the EoA Process is the most notable, as it describes the characteristics of the project and how companies approach the management. In relation to the execution time of the projects, this may vary. Small projects take between one and three months, medium projects take three to six months and large projects last more than six months. Note that the Quality Assurance EoA is treated in two ways. First, "Quality" relates to certifications, quality control, and quality management tools used by companies. Second, it relates to the concept of "Metrics" necessary for quality improvement.

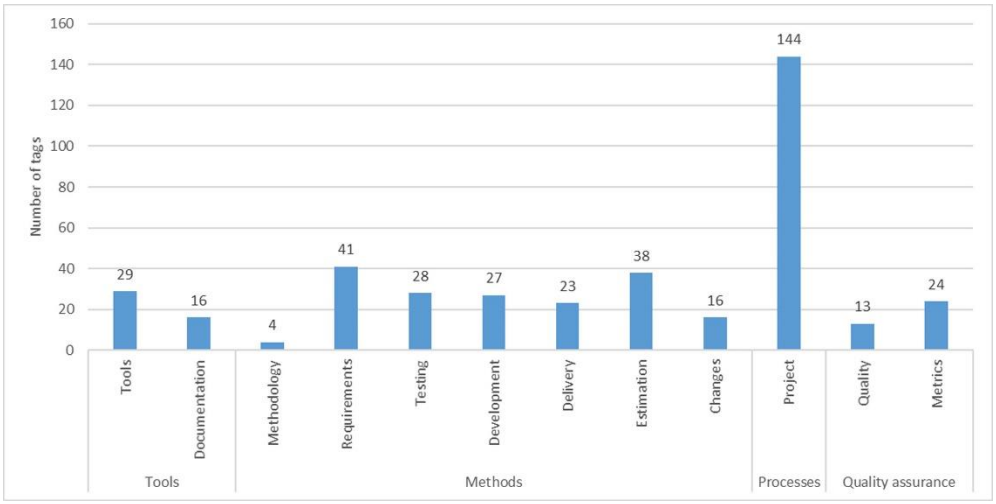

Figure 2. Quantity of tags collected for the Tools, Methods, Processes, and Quality Assurance elements of analysis within the Software Engineering research area.

On the one hand, none the studied companies – save one – explicitly manage quality; namely, there was no area, position, or role in the company with responsibility for quality. This is reflected in the amount of information delivered by companies (Fig. 2), where it was possible to collect more data on Metrics than Quality Assurance. Note that the concept of Quality Assurance is explicitly identified by some companies and directly related to product testing in Development. As previously mentioned, only one company mentioned that it specifically manages quality, since it has an Area and Quality Manager who helps project managers meet objectives, review project times, and prevent possible risks. With respect to the use of quality management tools, one company among the participants provided a template that is used to record different types of project data to be used in future analyses. In some cases, the project management system itself can acquire these metrics, which makes it easier to control the quality at different stages of product development. On the other hand, there is a large amount of information collected associated with Metrics tags (Fig. 2). There is a difference between managing quality and metrics, because – as mentioned above – only one company manages quality. However, a number of companies do collect data and information about their processes. Among the metrics collected are: real hours worked, hours dedicated to development, available development hours, average requirements per project, project delay time, billing delay time, impact of the system on the company, and the number of system users. Other ways of measuring are 5-minute meetings (stand up) that outline the status of the project, where it is counted, what has been done, what is currently being done, etc. It is important to mention that only two of the studied companies had a level of international certification, specifically one had CMMI 2 and the other had ISO 9001. The other companies had no certifications. Through the interviews, it was possible to collate the types of activities and tools used during software project development (Table 3).

Table 3. Tools used to develop software projects

Activities	Tools
Development languages	.Net, Jave, Php, in some cases Ruby, Python and frameworks like Larabel, Django Rails; creation of their own frameworks or code libraries
Project management	Pivotal Traker, Tagg, Gantt Charts (used frequently), custom project management systems (in some cases)
Team management	Slack
Implementation	Heroku
Documentation	Proposal document, Specification of requirements (2nd option)
Version control system (VCS)	Git, Subversion (principally)

Testing	Fabric
Database	Sql Server, MySql, PostgreSQL

Note that most companies use online and free tools, which reduce costs and optimize their services. In relation to the Documentation activity, the proposal document outlines high-level system functions, the cost of each in terms of development hours, some milestones, and the total value of the proposed project. The contents of this document depend on the client and the size of the project.

The software project development process

Some of the most important stages in software product development will be described in detail.

The software project development process was described for each company during the interviews, which could be used to develop a generalized schema of between 5 and 9 development stages (Fig. 3). Initial Meetings mark the starting points of software projects, where clients describe their problem or the project that they want developed. Then, a High-level Estimate of cost and time is calculated. The roles that participate most in these meetings are Project managers, Commercial or Sales managers, and/or Analysts, whose expertise helps estimate the approximate time and monetary costs of the project. The first requirement gathering is done during the Initial Meeting with the client. This is conducted by a technician that is capable of quantifying and organizing the client's ideas and supporting the sale of the product. One company stated that it uses particular requirement capture techniques, such as templates (screenshots) with a basic workflow, user experience, border conditions, and business model canvas in order to clarify the client's ideas.

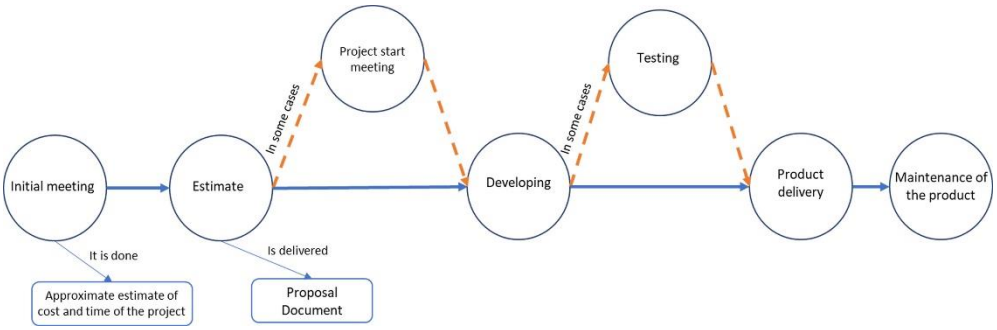

Figure 3. Stages of the Development of a software product

The next stage is Estimation, in which the project is analyzed more strictly in order to develop a more detailed proposal for the problem. Some companies hold meetings to present their work plans and identify project participants following proposal acceptance. During the Estimation stage, all companies stated that they use expert criteria as the technique to set prices and total time of the project. Among the factors considered for determining time are the complexity of the functionalities, a history of prior estimates, and the abilities of the programmers. Some companies also mention that slack time is included in projects, in order to solve eventualities or issues in development that may arise. In relation to changes, companies show flexibility, but this depends on the types of changes, with those greater than simple modifications requiring revaluation and freezing until the end of the project, in order to develop a new version of the system. Work during the Development stage is conducted by teams that consist of one to three professionals at most. In most of the participating companies, programmers work in only one project at a time, although this will depend exclusively on the number of workers in the company. In terms of work style during this stage, most interviewees talked about "parallelizing" and "modularizing" or otherwise dividing the project into different parts and creating parallel workflows that help develop the software product. Code management was done primarily using version control systems (VCS), although the companies that didn't use a VCS managed the code via emails or pen drives, with backup being done locally. Most development teams comprise a Project manager and Programmer. In terms of

project distribution, Project managers lead more than one project at a time, and a Programmer can participate in more than one project. In some cases, a Programmer – depending on their experience – can develop a project alone.

In most companies, the Project manager is in direct contact with the client, organizing meetings, showing progress, distributing functionalities, etc. The most used tools to manage a software project are Gantt charts and free or custom-built project management systems, which are combined in some cases (Table 3). These tools carry a lot of data, including: activities, dates, delivery milestones, and development times. Other tools and activities mentioned are creating local or remote project folders to store all related information and holding regular "clearing meetings" for people involved in the project to show progress on indicators (e.g., work hours, deliverables, real advances, etc.) using a Gantt chart. Clearing meeting participants also comment on a table or list of commitments, referring to the time expired or are about to expire, either on the part of the client or the company. In a few cases, project management is not supported by tools, with tasks or activities being in a mental model of the programmers that have different tasks to complete. Another concept mentioned in software project management is the slack or time margin in development. For the most part, these times are set aside for critical situations or to compensate for some changes in the project. Although they are only an estimation, during Development their time consideration is set aside for any eventuality. Note that not all companies use slack time, because the inclusion of these times could be detrimental to project bids.

After the Development stage (software product coding), a small percentage of companies conduct a Testing phase, in which they submit the software product to their own tests (functional testing) and those of the client. It should be noted that these companies have separate Development and Testing phases. In contrast, companies that pass directly from Development to Delivery mostly incorporate testing into the Development stage.

Once these development phases are complete, the project moves to the Delivery stage, which this is done in a variety of ways. The Delivery stage for most products is done by delivering a package or a set of functionalities (milestones), which range between 20% and 80% of total system operation. Once the functionalities are approved, the Development stage is reinitiated; in some cases, some requirements or functionalities need to be revisited. After delivering the product, some companies begin a Warranty stage, where product maintenance is offered to the client.

Factors

The Factors area shows each EoA – Success, Failure, or Risk – to be considered when developing a software project. As shown in Fig. 5, the smallest number of tags collected related to the Risk Factor EoA (tag: F_risk), which is because participant companies did not record canceled projects. In contrast, more data was found for the Success Factor EoA (tag: F_success), because most companies have several years of experience and service, making it easier to recognize the variables influencing company growth and success. Finally, the Failure Factor EoA (tag: F_failure) shows nearly the same number of tags as the Success Factor. This is because failure factors were counterparts to success factors. In other words, for a software project to succeed, the company must comply with the success factors while also accounting for and preventing failure factors.

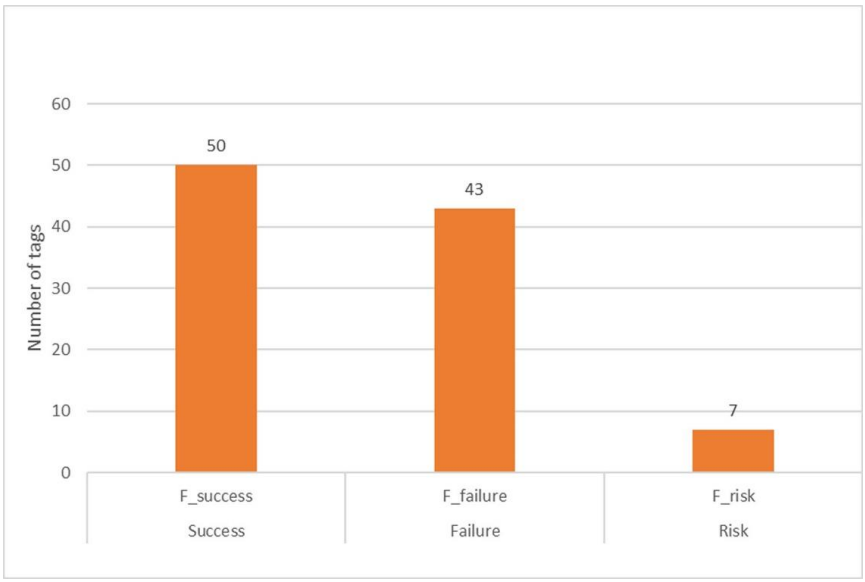

Figure 5. Amounts of data tags collected for the Success Factor, Failure Factor, and Risk Factor elements of analysis.

Among the Success Factors, interviews indicated the most dominant was the client. The participant companies describe a series of attributes related to the client, such as knowing and listening, showing progress, being transparent, and communicating, all of which help create a good client-company relationship. Such a relationship helps develop successful projects and facilitates negotiations when unforeseen eventualities occur. Another significant Success Factor is the identification of some client roles when developing a software project. It is fundamental to perceive the client as an active and empowered counterparty, able to take technical or financial decisions. It is also important to determine the final user of the product, both for requirement capture and system validation. Other Success Factors mentioned by companies are high-quality codes, orderly and structured programming, development team communication and commitment, and compliance or execution of a series of tasks, such as creating a solid data model, prototypes with some functionalities, test automation, good architecture, and project analysis that helps identify the backbone and needs of the project.

Among the Risk Factors, the most frequently mentioned is also the client, especially their relationship with the development team, with factors like little transparency, little communication, and lying about errors or project advances as elements damaging this relationship being notable. Another identified factor was the client not understanding the role of their "empowered counterpart" (Sponsor or Project Manager) in taking technical or financial decisions, either early on or as the project progresses. The loss, dilution, or change of this role can lead to deteriorating the relationship with the client and eventual project failures. Other notable risk factors are weak requirement capture processes, low levels of empathy with the client ("not putting oneself in the client's shoes"), the non-participation of key roles by either the client or the company, and poor project planning. As mentioned before, none of the participating companies had cancelled projects or unused projects. However, in some cases, elements such as rigidity in the project or by the client, and poor analysis were described as Failure Factors.

Discussion

It was possible to collect data relevant to this study on how Chilean software developer work or execute their projects. This data makes it possible to compare this study with results of prior Chilean and international studies.

The area of Demographics, which describes the attributes of the surveyed companies, showed that six of the eight surveyed companies had fewer than 15 employees. This result is in line with GEMS (GEMS, 2015), who showed that more than half of surveyed companies have fewer than 16 workers. Based on this, it is possible to deduce that most software development companies in the Chilean market are small businesses, according to Dini and Stumpo (Dini &

Stumpo, 2002) and Saavedra and Hernández (Saavedra G. & Hernández C., 2008). Chilean companies report having a good work environment, with very little staff turnover and fluid communication. As one CEO stated, "Caring for the work environment is relevant. Happy people make happy codes." These mirror results from GEMS (GEMS, 2015) and Vidal-Silva et al. (Vidal-Silva et al., 2017), which showed low rates of turnover and workers with more than four years' experience, respectively. Taken together, these results demonstrate that the Chilean software market is a good place for professional development. The results are similar internationally. Acevedo and Mejía (Elles & Triviño, 2016) indicate that worker experience in the Colombian software development sector is greater than five years. The high level of experience and low level of turnover could be due to the growth of the Latin American software economy (Fundación País Digital, 2017), (Casado, 2010) (Elles & Triviño, 2016). When analyzing the information from the Software Engineering area, the most-used tools in developing a software product were proprietary code libraries, version control systems, project management systems (mostly freeware), and the Proposal document (as a physical document). Meetings were given a special mention, since they are a highly used activity, especially to meet with clients. All of these results align with the Chilean and international findings of GEMS (GEMS, 2015) and Rojas and Carrizo (Rojas & Carrizo, 2017), especially the use of "code versioning," code repositories, and both formal and informal client meetings. This result shows the modern software industry, using tools to streamline work processes and remain in constant contact with clients. When considered as development methodologies, rather than established and strictly executed work methodologies, they are a set of good practices, with some variants, depending on the company. In most cases, these good practices are respected and will depend on the project type. When comparing these results, GEMS (GEMS, 2015) found that 80% of surveyed companies stated they had defined their own development processes, and Rojas and Carrizo (Rojas & Carrizo, 2017) found that 77% of surveyed companies use Scrum as their development methodology. These data indicate that most companies tend to standardize their development methods, either through good practices or defined methodologies. However, GEMS [4] also examined adaptability, with only 40% of surveyed companies indicating process adaptation. However, given the ubiquitous use of Scrum (Rojas & Carrizo, 2017),which allows for executing more adaptable development stages (Letelier et al., 2016), and the strong influence that clients can exert on the software development process, this low reported percentage of companies that adapt or accommodate their project processes is questionable.

With respect to Estimation, all studied companies based this task on experience, with expertise being the most use criterion. A one Factory Manager explains, "With experience you take the shortest past to the end. No experience means lots of twists and turns, but you do get there in the end." These results concur with the findings of Rahikkala et al. (Rahikkala et al., 2018), who state that all the cases they studied use experience as one of the main estimation factors. This also resonates with Sehra et al. (Sehra et al., 2017), who found that expert judgment or experience are the most-used estimation approaches. Regarding changes, there is no mention of modification documents; they are only reflected in agreements as part of the final project collection. Likewise, most companies show flexibility when addressing these changes. In most cases, this is due to the slack time incorporated in projects, which allows such modifications to be made without issue. Sometimes, there are mentions of agreements between the company and client to make bigger changes to the project, to be revalued and billed separately. The completion times for software projects can vary between one and six months, with one- and two-month projects being the most common. Development teams vary between one and three members, depending on the project size; in some cases a project can be developed by a single programmer. As one Project Manager and Programmer explains, "The software industry is bespoke, so it's impossible to have 100% occupation. You work based on projects, and it possible that someone is idle for a month." Results similar to GEMS (GEMS, 2015), showing that more than half of respondents had projects shorter than four months and in most cases developers work on only one project at a time. This describes agile and dynamic small companies that develop quickly, but with few resources. Although quality management is known to be fundamental to product development, not all companies include it, due to the cost involved of implementing and maintaining a quality management unit in the company. In this study, only two companies declare having quality control certifications. This low level of certification mirrors a trend over time, with Gechs (GECHS, 2008) and Gómez (Gómez, 2013) also showing low CMMI and ISO certification levels among surveyed Chilean companies. The low levels of certification common throughout Latin America (Gómez, 2013) could be explained by the continued context of artisanal development that doesn't incorporate certified methods in order to standardize and assure the development of consistent high-quality software, despite the push for professionalizing software development, incorporating tools, and having agile and contemporary development methods. In relation to Metrics, certain companies generate some data, such as real hours worked, development hours, available development hours, and available management hours. In the GEMS (GEMS, 2015) report, the majority of respondents kept track of development defects, post-delivery defects, and unplanned changes. Based on these combined findings, it appears that Chilean companies do not collect metrics at all project stages. This is very similar to the findings of Metaute and Serna (Metaute P. & Serna A., 2016),

who found that 71% of medium-sized companies in Medellin, Colombia, apply metrics, but use them mostly in the development and requirement stages. These limitations in incorporating metrics in all development stages could be due to the small size of these companies, and the associated limitations of staff and resources for fully carrying out this task.

Among the Factors related with Success, Failure, and Risk, the client stands out, with communication, trust, and engagement as significant aspects of the client relationship. This influence is due to the high level of involvement and the type of relationship they have with projects, due primarily to a more open and inclusive work style development companies currently have, which allows them to be integrated through all stages of project development. This was expressed by another CEO as, "The developer helps the client, as the client must help to build the solution." Likewise participating companies state that they do not have failed projects, which is consistent with the results of GEMS (GEMS, 2015), that found only 4% of projects were rejected, compared to 84% acceptance and 12% conditional acceptance. These results imply the presence of a solid and reliable industry, but with the same limitations in resources, personnel, and process standardization.

Limitations And Future Work

This study has some limitations. First is the low level of participation by software development companies, claiming a lack of personnel and time, which limits the sample size of the study. The number of participating companies is not representative of the totality of the Chilean software development industry. As such, these results cannot be generalized, but should be considered as tendencies. However, financial and time limitations on the part of the study group also restricted the number of possible companies included in the study. Finally, the lack of studies focused on understanding the condition of the Chilean software development company practices limited the types of discussion and comparison of results as well as made it impossible to generate new proposals based on existing studies. In order to mitigate or counteract threats to validity, a number of actions were conducted to allow for obtaining more detailed results. When collecting data at each company, interviews and face-to-face conversations were conducted and recorded with different employees. This allowed for directly extracting information from individuals, instead of using generalized questionnaires or other methods that would distort the information. The collected data was split into different concepts (tags), and this allowed for gaining specific understandings of each activity, task, or action conducted in each company when developing a software product. Finally, there was a process of transferring research results to participating companies, obtaining good feedback, thanking them for their efforts, and committing to future research.

Conclusions

Understanding how the software industry operates in Chile is essential for implementing and fostering continuous improvements in the sector, especially in a context where various studies highlight significant growth in software development and programming services. Therefore, analyzing this industry in Chile takes on particular relevance. Since this exploratory study is based on a limited sample of companies (only 8), it is important to emphasize that the results reflect trends observed in this specific sample and do not necessarily represent the entire Chilean software industry. Through this study, it was observed that companies develop their projects by employing sets of best practices that can vary or adapt depending on the type of project. These processes are complemented by tools that facilitate management, control, and support in the creation of software products. Although there are various methodologies for software development, all begin with requirements gathering, followed by budgeting, product coding, and testing, leading up to the final delivery, which is often done in phases depending on the project type. One notable aspect is that, although the participating companies recognize quality management as a fundamental pillar in software development, few are actually performing this task, attributing this to a lack of resources. In fact, all companies indicated that many of the limitations they face are due to limited resources, whether financial, human, or otherwise. It is worth noting that software development processes are not solely technical, formal, and structured; in many cases, they also involve social skills that combine with technical elements. As one interviewee noted, examples of these skills include the ability to accurately capture requirements or to handle situations where clients request changes without considering the consequences of these adjustments, which generates pressures in work and business relationships. Among the factors influencing the success, risk, and failure of software project development, the client plays a prominent role. This indicates that development companies are adopting an open and inclusive approach with their clients, integrating them throughout the software development process. Future work will propose various

improvements based on the findings of this study, which can be implemented in similar companies. These proposals aim to contribute to the progress and strengthening of the software development industry in Chile.

References

- Álvarez, V. A. and Lillo, C. B. (2008) 'Chile: Desarrollo Endógeno con Proyecciones Externas', in Desafíos y oportunidades de la industria del software en América Latina. Primera ed., Mayol Ediciones.
- Barros-Justo, J. L., Olivieri, D. N., and Pinciroli, F. (2019) 'An exploratory study of the standard reuse practice in a medium-sized software development firm', Computer Standards & Interfaces, 61, pp. 137–146. doi: 10.1016/J.CSI.2018.06.005.
- Beigl, M., Michahelles, F., Tokuda, H., and Hodges, S. (2017) 'Programming the World', Computer (Long. Beach. Calif.), 50(11), pp. 14–17. doi: 10.1109/MC.2017.4041361.
- Carrizo, D. and Alfaro, A. (2017) 'Methodological Proposal for Ethnographic Research in Software Engineering: A Case Study', in Proceedings of the International Conference on Information Technology & Systems (ICTS 2018), pp. 700–709.
- Casado, C. V. (2010) La industria de software: una experiencia de empresas, gobiernos y universidades en Uruguay y Ecuador. FLASCO, Ecuador.
- Dini, M. and Stumpo, G. (2002) 'Análisis de la política de fomento a las pequeñas y medianas empresas en Chile (vol. 136)'. Santiago, Chile.
- Ebert, C., Kuhrmann, M., and Prikladnicki, R. (2016) 'Global Software Engineering: Evolution and Trends', in 2016 IEEE 11th International Conference on Global Software Engineering (ICGSE), pp. 144–153. doi: 10.1109/ICGSE.2016.19.
- Elles, A. A. and Triviño, J. I. M. (2016) 'Estudio sobre el nivel de desarrollo de software en la ciudad de Cartagena de Indias', Universidad San Buena Aventura.
- Fundación País Digital (2017) Globalizing ICT Imports and Exports of ICT Products and Services. Santiago, Chile.
- GECHS (2008) Sixth Diagnosis of the National Software and Services Industry. Viña del Mar, Chile.
- GEMS (2015) Statistical analysis of the characterization survey of software SMEs in Chile. Department of Computer Science, Santiago, Chile.
- Gómez, M. (2013) El Impacto de las Certificaciones de Calidad en las Exportaciones de Servicios TI de Chile. Universidad de Chile.
- Letelier, P., Leon, A., and Bruno, M. (2016) 'Assessing and improving the level of agilism in team work', in 2016 XLII Latin American Computing Conference (CLEI), pp. 1–12. doi: 10.1109/CLEI.2016.7833326.
- Maul, A. (2017) 'Rethinking Traditional Methods of Survey Validation', Measurement: Interdisciplinary Research and Perspectives, 15(2), pp. 51–69. doi: 10.1080/15366367.2017.1348108.
- Metaute P., P., and Serna A., A. (2016) 'Diagnosticon the appropriation of metrics in software medium enterprises of Medellin city', Revista Antioqueña de las Ciencias Computacionales y la Ingeniería de Software, 6(1), pp. 6–14.
- Orantes Jiménez, S. D. (2012) 'Metodologías híbridas para desarrollo de software: una opción factible para México', Revista Digital Universitaria, 13(1).
- Pereira R., J., Cerpa T., N., and Rivas M., M. (2004) 'Factores de éxito en proyectos de desarrollo de software: análisis de la industria chilena del software', in Workshop de Ingeniería de Software. Arica, p. 8.
- Press, L. (1991) 'Computing in Chile: Leading the way for other South American LDCs?', Association for Computing Machinery, 34(12), p. 27. [Online]. Available at: https://go.galegroup.com/ps/anonymous?id=GALE%7CA11658692&sid=googleScholar&v=2.1&it=r&linkaccess=abs&issn=00010782&p=AONE&sw=w (Accessed: 29 October 2019).
- Rahikkala, J., Hyrynsalmi, S., Leppänen, V., and Porres, I. (2018) 'The Role of Organisational Phenomena in Software Cost Estimation: A Case Study of Supporting and Hindering Factors', Information and Software Engineering Journal. doi: 10.1111/j.1365-3059.2011.02580.x.
- Rojas, J., and Carrizo, D. (2017) 'Gap in Requirements Engineering Between Scientific Literature and Practice of Latin-American Software Industry', Interciencia, 42(10), pp. 676–682.
- Saavedra G., M. L., and Hernández C., Y. (2008) 'Caracterizació n e importancia de las MIPYMES en Latinoamérica: Un estudio comparativo', Actualidad Contable FACES, 11(17), pp. 122–134.
- Sehra, S. K., Brar, Y. S., Kaur, N., and Sehra, S. S. (2017) 'Research patterns and trends in software effort

estimation', Information and Software Technology. doi: 10.1016/j.infsof.2017.06.002.
- The Standish Group (2013) CHAOS MANIFESTO 2013: Think Big, Act Small. Standish Group International.
- Vidal-Silva, C. L., Madariaga, E. A., and Solís, R. A. (2017) 'Pilot Study on the Importance of Performance, Safety and Reliability in the Software Development Process in Chile', Informacion Tecnologica, 28(3), pp. 95–106. doi: 10.4067/S0718-07642017000300011.
- Wang, D., and Galster, M. (2018) 'An Exploratory Study on Software Products and Development Organizations in New Zealand', in Springer, Cham, pp. 141–149.