A Review of SDLCs for Big Data Analytics Systems in the Context of Very Small Entities Using the ISO/IEC 29110 Standard-Basic Profile

David Montoya-Murillo
Department of Electronic Systems
Autonomous University of Aguascalientes, Mexico
david.montoya@edu.uaa.mx

Manuel Mora
Department of Information Systems
Autonomous University of Aguascalientes, Mexico
jose.mora@edu.uaa.mx

Sergio Galvan-Cruz
Department of Electronic Systems
Autonomous University of Aguascalientes, Mexico
sergio.galvan@edu.uaa.mx

Angel Munoz-Zavala
Department of Statistics
Autonomous University of Aguascalientes, Mexico
eduardo.munoz@edu.uaa.mx

Francisco Alvarez-Rodriguez
Department of Computer Science
Autonomous University of Aguascalientes, Mexico
francisco.alvarez@edu.uaa.mx

Abstract: *Context: A Systems Development Life Cycle (SDLC) is a model of phases-activities, roles, and products systematically used to develop software with functional expected quality. Although SDLC is widely applied to various software types, it remains unusual in Big Data Analytics Systems (BDAS). Objective: To address this issue, several SDLCs for BDAS have been proposed, along with comparative studies, to guide interested organizations in adapting them. This research seeks a lightweight, balanced, and feasible for small development teams or organizations, taking advantage of favorable characteristics of the international ISO standard. Method and Materials: This study describes the knowledge gap by reporting a comparative analysis of four relevant SDLCs. A selective research method was applied (CRISP-DM, TDSP, BDPL, and DDSL), focusing on alignment with the recent ISO/IEC 29110-basic profilestandard. The goal was to identify which SDLC contributes and fits better from a lightweight approach. Results: From the rigorous approach Cross Industry Standard Process for Data Mining (CRISP-DM) showed the highest alignment with the standard, for the agile approach it was Domino Data Science Lifecycle (DDSL) being the closest of the four. Team Data Science Process (TDSP) stood out as the most agile of those analyzed but fell short of the required results. BDPL, which manages another standard, was too rigorous and more distant. Conclusions: Research on new SDLC for Big Data Project Lifecycle (BDPL) has been practically nonexistent in software engineering from 2000 to 2023. Only BDPL was found in the academic literature, while the other three came from gray literature. Despite the relevance of this topic for BDAS organizations, no adequate SDLC was identified.*

Keywords: *BDAS, SDLC, ISO/IEC-29110-standard for VSEs, CRISP-DM, BSPL, TDSP and DDSL.*

Received June 6, 2024; accepted November 7, 2024
https://doi.org/10.34028/iajit/22/1/15

1. Introduction

The systematic, disciplined, and quantified development of software has been guided by System Development Life Cycles (SDLCs) provided by the software engineering discipline [6]. An SDLC refers to "the software processes used to specify and transform software requirements into a deliverable software product" [6]. An SDLC is usually represented as a software development process model [26] of phases-activities, roles, and products proposed to build systematic software products on the expected time, budget, and functional quality, i.e., the named Iron Triangle [54]. SDLCs are realized by practitioners and academics through software development methodologies (e.g., Rational Unified Process (RUP) [28]), international software process standards (e.g., ISO/IEC 12207 [24]), and international software process frameworks (e.g., CMMI-DEV [9]).

The software development methodologies, and international software process standards and frameworks have provided valuable benefits to the software product, the software process, and stakeholders such as customers-users and the development team [13, 51, 70] such as: reduction of project costs, software products with higher quality, more precise project schedule estimations, greater user satisfaction, and in overall less wasting of relevant human, economic and technological organizational resources [13, 51, 70]. Consequently, utilizing these software development methodologies and international software process standards and frameworks is a common practice in large- and medium-sized organizations for practically any type of software.

However, although SDLCs have been widely used for diverse types of software, for the emergent kind of Big Data Analytics Systems (BDAS), their utilization has been reported as very scarce [18, 29, 31, 38, 39, 62]. For instance, a conceptual review study [38] reported a research bias on BDAS algorithms, platforms, languages, and applications, but minimal in the Big Data Software Engineering (BDSE) area where SDLCs can be proposed. Later, in three studies using systematic literature review or systematic mapping methods [29, 31, 62], it is reported that there is initial research on BDSE, but it is partial and focused on particular phase-activities of a generic SDLC rather than on a complete SDLC for BDAS. In [29], it was found that most available literature concentrated on proposing BDAS frameworks and architectures rather than full SDLCs. In turn, in [62], it was reported a set of critical success factors for BDAS projects, one of them the software development process model, and thus its implicit realized SDLC through a methodology or standard. In [31], it was reported that Architecture Design is the most published research on phases of SDLCs for BDAS but again minimal on full SDLCs. In [39], a Systematic Literature Review found 19 SDLCs of type heavy-weight, lightweight, and agile but reported that only a few ones can be considered almost complete SDLCs, including project management, team management, and Data Management roles, activities, and products. Two mains were Microsoft Team Data Science Process (TDSP) [41] and Domino Data Science Lifecycle (DDSL) [12], which have been classified as agile [60] and lightweight, respectively [42]. Finally, in [18], single case study research conducted in a large international company reviewed the agile Microsoft TDSP [41] and the classic Cross industry standard process for data mining (CRISP-DM) [7] SDLCs and was reported that both SDLCs need to be completed with missing activities and expected products from a BDSE perspective.

In summary, these studies [18, 29, 31, 38, 39, 62] using conceptual review, systematic literature review, systematic mapping, or single case study research methods report that there are already several SDLCs proposed for BDAS but:

1. None well-accepted, and systematic SDLC specific for BDAS has gained a relevant international acceptance.
2. Large and medium-sized organizations have used CRISP-DM but it was proposed before the technical and organizational requirements demanded by BDAS and thus it must be ad-hoc adapted.
3. Initial agile SDLCs have been proposed but scarcely tested in real-life projects.

The research on and the practical availability of well-tested SDLCs for BDAS is relevant because BDAS is new high-valued software for organizations because they have provided decision-making benefits mainly in the domains of marketing, healthcare, finance, and manufacturing [1, 72], but they are also complex software products because they require complex computational processing, storing, and networking resources to apply advanced algorithms on large or very large datasets [30, 49, 56], and relevant organizational, economic and human technical resources usually only owned by large enterprises [10]. Consequently, whereas the adaptation of a few comprehensive SDLCs for BDAS as such CRISP-DM has been relatively useful, still frequent failed BDAS projects are reported [11, 55].

Hence, practitioners and academics in the domain of BDAS demand well-tested SDLCs designed for BDAS. Given this problematic situation for BDAS, some former data mining machine learning SDLCs before the big data context have been adapted [14, 15, 53], and new ones have been proposed [12, 41, 36]. The main classic SDLCs that are being adapted are CRISP-DM [7], and SEMMA and three relevant relatively new ones are Microsoft TDSP [41], DDSL [12], and Big Data Project Lifecycle (BDPL) [36], which extends the ISO/IEC 15288 standard [25]. These SDLCs approach has been effective in creating large-scale robust systems, but they are usually applied in large- or mature medium-sized organizations [53].

For the case of small organizations or Very Small Entities (VSEs), defined as development teams of 5 to 25 people at maximum [32, 33], these main SDLCs for BDAS can be considered difficult to completely the lack the economic, organizational, and technical resources required for this type of projects [32, 33]. Consequently, small businesses and VSEs lose all potential benefits provided by heavyweight SDLCs that large and mature medium-sized organizations receive [32, 33].

To cope with this derived problematic situation faced by small organizations and VSEs, the software engineering discipline has proposed agile [66] and lightweight [23], i.e., balanced SDLCs between heavyweight and agile ones such as Scrum [36] and the ISO/IEC 29110 standard [23] for small organizations and VSEs. Whereas agile SDLCs have gained relevant momentum for this type of organization, other small organizations and VSEs require a balanced software process for gaining international certifications [45, 48, 71], and thus, lightweight SDLCs as the provided by the ISO/IEC 29110 [48] have also gained relevant acceptance in the international community of small organizations and VSEs [45, 48, 71]. In particular, a balanced heavyweight-agile approach is strongly recommended for small software projects [5] and thus for being feasible to be used for small organizations and VSEs. Figure 1, adapted from [5], illustrates a 5-factor landscape to situate the landscape of heavyweight, lightweight, and agile SDLCs. This 5-factor refers to organizational culture, requirements dynamism, development personnel expertise, application criticality, and project size. Thus, SDLCs closer to the center favor agility (inner circle), and their counterparts (outer circle) are suitable for heavyweight software projects where time, technological complexity, or the number of people are considerably higher than in agile projects. A lightweight SDLC lies between these two approaches for relatively small and medium projects.

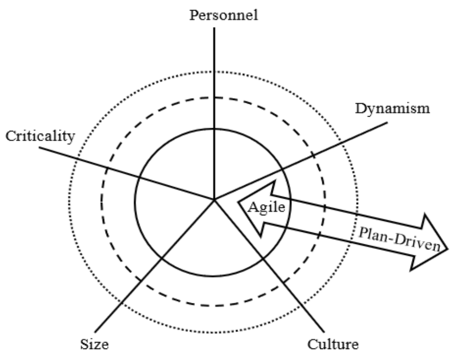

Figure 1. 5-factor model of heavyweight, balanced, and agile software development project cases [5].

Hence, although the previous research consulted [18, 29, 31, 38, 39, 53, 62] on SDLCs for BDAS has provided valuable insights, it was also identified that studies on lightweight SDLCs for BDAS in the context of small organizations and VSEs are missed [42, 60]. Consequently, this research addresses this knowledge gap and reports a conceptual review of four main SDLCs for BDAS identified in the literature such as CRISP-DM [7], Microsoft TDSP [41], BDPL [36], and DDSL [12]) regarding the new country-certifiable ISO/IEC 29110 standard-basic profile [23], which was designed for VSEs.

For this aim, a Pro-Forma of the ISO/IEC 29110 standard is used to examine the conceptual structure, i.e., phases-activities, roles, and products of the three SDLCs. A Pro-Forma [2] is a pre-defined template reporting a set of organized concepts used as conceptual lenses to verify the extent of convergence to them from conceptual entities of interest in study (in this research, the four SDLCs for BDAS). Pro-Formas have been used for similar conceptual reviews [3, 57].

This article continues as follows: Section 2 reports a summary of the Research Approach. Section 3 reports the Theoretical Background regarding BDAS and the ISO/IEC 29110 standard-basic profile. Section 4 reports the summaries of the review of the three SDLCs for BDAS and an overall evaluation of them. Finally, in section 5, the Conclusions and recommendations for further research are presented.

2. Research Approach

This research applies a conceptual review research method [3]. This study does not apply a Systematic Literature Review research method [57] because its research purpose is not to provide statistical-descriptive accountability of findings on SDLCs for BDAS but to provide a thorough analysis for a better understanding of the structure of the main SDLCs for BDAS, according to the consulted literature [18, 29, 31, 38, 39, 62]. Then, sub-section 2.1. reports the research objective and the research questions. Sub-section 2.2. reports the conceptual review research method.

2.1. Research Objective and Questions

In this research, it is used a structured research objective template adapted from [64]. The adapted template is as follows: Analyze<Objects of study>with the purpose of <Purpose> concerning their <Quality Focus> from the perspective of <Perspective> in the context of <Context>.

Consequently, by using the previous template, the research objective is formulated as follows: "Analyze <the main SDLCs for BDAS> with the purpose of <describing, comparing, and evaluating them> regarding their <alignment of roles, phases-activities, and products against a lightweight SDLC ISO/IEC 29110 standard-basic profile> from the perspective of <the BDAS academic community> in the context of <SDLCs for BDAS reported in the main literature on BDAS>." Three specific Research Questions (RQ) were also stated as follows:

- **RQ 1:** What is the high-level structure (roles, phases-activities, and products) of the main selected SDLCs for BDAS?
- **RQ 2:** What is the degree of alignment in roles, phases-activities, and products of the SDLCs for BDAS identified in RQ.2 concerning the lightweight SDLC of the ISO/IEC 29110 standard-basic profile?
- **RQ 3:** Can the four analyzed SDLCs for BDAS be considered lightweight? First, confirm that you have the correct template for your paper size. This template has been tailored for output on the A4 paper size. If you are using US letter-sized paper, please close this file and download the Microsoft Word, letter file.

2.2. Conceptual Review Research Method

Table 1 summarizes the four research steps, purposes, and outcomes of the conceptual review research method [3] used in this study:

1. Research formulation.
2. Research design.
3. Research analysis and synthesis.
4. Research reporting.

- *Step* 1. Research formulation was reported in section 2.1.
- *Step* 2. The research design was executed in 3 sub-steps:

  a) Selection of potential sources of the objects of study.
  b) Selection of the objects of study in the potential sources.

c) Establishment of the concept tool for the analysis (standard proforma).

In step 2.a), the research team identified three recent comprehensive literature review studies published in high-quality journals (high impact on Journal Citation Report (JCR)) regarding system development cycles for big data projects: (Martinez *et al*. [39] with 122 citations, Giray [16] with 164 citations and Saltz and Iva [61] with 30 citations).

Table 1. Conceptual research method.

Step	Purpose	Outcomes
1. Research formulation.	To state the expected research objective that delimits the scope of the research, and the research questions that focus on the knowledge gaps of interest.	• Research objective statement. • Research questions.
2. Research design.	To agree the sources to collect the materials regarding the objects of study, and to define the conceptual tool that will be used to analyze the objects of study.	• Sources of materials. • Documents of the objects of study. • Conceptual tool for conducting the analysis.
3. Research analysis and synthesis.	To conduct the analysis and synthesis of findings, using the conceptual tool, on the objects of study.	• Structured schemas of findings. • Summary of findings. • Conclusions on findings.
4. Research reporting.	To produce valid and visible results for academic venues and outlets.	• Research report.

Martinez *et al*. [39] analyzed 19 development cycles; Giray [16] identified 17 studies on software engineering life cycles for big data; Saltz and Iva [61] located 27 studies where three methodologies were the main ones.

In step 2.b), with these three studies of potential sources of SDLC for BDAS, the research team agreed to select the top 5 reported in any of these 3 comprehensive articles. Table 2 reports the 5 selected methodologies: KDD, CRISP, BDPL, TDSP, and DDSL, with descriptive data. The research team agreed to carefully analyze each of the 5 methodologies and found that the Knowledge Discovery in Databases (KDD) did not meet the characteristics of a development cycle (roles, phases, and artifacts) and was therefore not considered. KDD [14, 15] was not included because a more detailed review indicated that it does not qualify as an SDLC but as a quick guide for developing BDAS. CRISP-DM is the main first one that is always reported in the main consulted literature and is highly referenced in its utilization [18, 31, 39, 62]. Microsoft TDSP is a new one more referenced in the modern literature [18, 31, 39]. DDSL [12] Although it has been recently reported and analyzed [39], this study updates its comparative analysis. The SDLC of BPDL [36] is the unique SDLC for BDAS found that it is based on an ISO/IEC standard i.e., the ISO/IEC 15228 systems and software engineering system life cycle processes [25]. Additionally, it must be remarked that this research focuses on lightweight SDLCs but not on agile ones. Thus, well-identified agile SDLCs for BDAS were not considered objects of study in this research. In step 2.c), the research team also agreed to elaborate a Pro-Forma of the SDLC implicit in the ISO/IEC 29110 standard-basic profile of roles, phases-activities, and products. Step 3, research analysis and synthesis were conducted by the two first researchers and later reviewed by the remaining three researchers. Finally, all five researchers agreed on the final version of the findings, and in step 4, research reporting, this article was written.

Table 2. Set of the top 5 to SDLC for BDAS.

SDLC for BDAS	Type of SDLC for BDAS	Publication domain	Publication name	Type of publication	Publication IF	Publication year	Citations	Is the SDLC reported in other SLR studies?
KDD	-	Analytics data science	Common. of the ACM	JCR journal	22.7	1996	3641	Martinez *et al*. [39], Saltz and Iva [61]
CRISP-DM	Heavyweight	Analytics data science	SPSS Inc. Website	Grey literature	-	2000	2,017	Martinez *et al*. [39], Saltz and Iva [61]
TDSP	Lightweight	Analytics data science	Microsoft Azure Website	Grey Literature	-	2016	22	Martinez *et al*. [39], Saltz and Iva [61]
DDSL	Lightweight	Analytics data science	Domino Data Lab Website	Grey literature	-	2017	6	Martinez *et al*. [39]
BDPL	Heavyweight	Software engineering	IEEE IT PROF	JCR journal	2.590	2018	15	Saltz and Iva [61], Giray [16]

3. Theoretical Background

To obtain the main heavyweight and lightweight SDLC for BDAS, the research design was executed by a selective manual search of the SDLCs for BDAS collected in the main literature [18, 29, 31, 38, 39, 62].

3.1. Big Data Analytics Systems (BDAS)

BDAS is a specific category of software applications in analytical data science. Analytics data science is a new area of study combining statistics, artificial intelligence, and computer science to examine, predict, or prescribe decisions. However, only large business organizations are the usual customers and end-users of analytics data science projects, and they focus on costly big data platforms [50, 68]. It is usually difficult for small and medium-sized enterprises to benefit from the use of analytics data science projects. Nevertheless, analytics data science approaches can also be used for big data for small business projects [17, 21, 27, 67].

Nonetheless, it is challenging to complete development projects for both small and BDAS successfully [11, 55]. Numerous global reports suggest that a significant portion of BDAS initiatives did not come in under budget, on time, or with the functionality that was anticipated [11, 55]. To address the issue of unsuccessful projects, agile development methodologies for BDAS have been proposed [35, 69]. However, agile methodologies have also come under fire for producing more stable software applications, leading to the recommendation of a more balanced development approach for VSE, which includes small business projects and projects involving five to twenty-five people in medium-sized organizations [5, 23, 48].

BDAS has been characterized by 5 V's qualities [4, 37, 44, 52, 58]: Volume, Velocity, Variety, Veracity, and Value. Volume alludes to a tremendous number of commerce occasions to be enlisted that request capacity capabilities as a rule datasets within the run of Terabytes (1012 bytes), Petabytes (1015 bytes), Exabytes (1018 bytes), or bigger datasets. Velocity alludes to the rate of information era, i.e., higher frequency of information enrolling within the run of million or more occasions by trade day. Variety accepts totally with wealthy differing qualities of information sources (inside vs. outside, manual client vs. robotized machine produced, real-time vs. clump ingestion motors), information structuredness (SQL vs. non-SQL), information designs (content vs. twofold, analogic vs. computerized, scrambled vs non-encrypted), and information sorts (char, string, numbers, genuine, picture, sound, video). Veracity accounts for the general quality of the datasets characterized by objectivity, honesty, and validity [37]. The value measures the unmistakable (taken a toll diminishment, benefit increases, trade proficiency measurements, among others) and intangible (technique, commerce notoriety, advertise esteem, among others) benefits created by utilizing huge amounts of Information. Value can be classified in esteem disclosure through exploratory activities for finding potentially profitable trade experiences, esteem creation through the inner utilization of BDAS for augmenting commerce esteem of the firm, and esteem realization through the conveyance of end-user's items and administrations upgraded with BDAS.

3.2. The ISO/IEC 29110 Standard-Basic Profile

The discipline of software engineering has developed software process standards and models [46, 70] to help business organizations manage and build high-quality software such as the ISO/IEC/IEEE 12207 [24], the ISO/IEC 33004 [22], the CMMI-DEV [9], and the Team Software Process (TSP) [19] that have produced multiple benefits to large enterprise software projects.

In contrast, small business has very limited budgets, less technical and managerial expertise, a lack of interest in using heavy-process software standards and models, a highly dynamic and informal organizational culture, and pressures for fast delivery from customers [8, 34, 47, 65]. Then, small businesses either ignore or reject the utilization of these software process standards and models designed for large enterprise software projects by the inherent organizational, technical, and economic barriers.

To address the need for counting with software process standards and/or models for small business software projects, the International Organization for Standardization and the International Electrotechnical Commission elaborated the ISO/IEC 29110 series of standards and guidelines [23, 48], which can be considered of lightweight type, i.e., a balanced approach between heavyweight and agile development approaches. The ISO/IEC 29110 series provide software and systems engineering processes to VSE, i.e., project teams from 5 to 25 people to improve their product quality as well as their process performance [23, 48]. These series of standards and guides have been proposed to have several profile groups [34]. The generic profile group of the ISO/IEC 29110 standard applies to VSEs that do not develop critical systems or critical software, i.e., software usually developed by large businesses. This generic profile group contains four categories: Entry, basic, intermediate, and advanced. Table 3 reports the difference between these four profiles regarding the number of processes, tasks, products, and roles.

Table 3. Processes, tasks, work products, and functions of each software profile in the ISO/IEC 29110 standard.

	Entry	Basic	Intermediate	Advanced
Number of processes	2	2	3 (+1 conditional)	3 (+3 conditional)
Number of tasks	40	67	107 (+8 conditional)	120 (+24 conditional)
Number of work products	14	22	39 (+3 conditional)	41 (+5 conditional)
Number of roles	3	7	8 (+1 conditional)	8 (+1 conditional)

The core of the ISO/IEC 29110 standard is a set of predesigned engineering and management guides that focus on project management and software or system development. The ISO/IEC 29110 standard is designed for use with any life cycle, such as waterfall, iterative, incremental, evolutionary, or agile [23, 48]. Figure 2, adapted from [23], shows the two phases (called processes) and activities of the software engineering ISO/IEC 29110 standard-basic profile.

The 7 roles of the ISO/IEC 29110 standard-basic profile [23] refer to: Customer (CUS) as the person or group of persons who know the customer domain process and requirements, and the authority to make decisions on the requirements, and changes, and the delivered product; Project Manager (PM) as the administrative person responsible for the project who has attributes of leadership, supervision of personnel, and financial and software development knowledge and experience; and Work Team (WT) role that can be Analyst (AN), Designer (DES), Programmer (PR), or

Team Leader (TL), as to the software development technical people responsible for building the expected software product. It is expected that the analyst elicits, documents, and validates software requirements; the designer defines and validates the proposed software architecture; the PR Programmer builds, tests, and integrates the software code pieces; and the team leader coordinates the technical Software Implementation process.

Table 4. Pro-Forma of the ISO/IEC 29110 standard-basic profile conceptual research method.

SDLC element	SDLC element description
Roles (7)	• User roles: **R.1:** Customer. • Management roles: **R.2:** Project manager. • Technical roles: **R.3:** Work team. **R.4:** Technical leader. **R.5:** Programmer. **R.6:** Designer. **R.7:** Analyst.
Process Activities (2-10)	• **Process 1.** Project Management: The purpose of the project management process is to establish and carry out in a systematic way the tasks of the software implementation project, which allows complying with the project's objectives in the expected quality, time and costs. • **Activity 1:** 1. Project Planning: It documents the planning details needed to manage the project. 2. Project Plan Execution: It implements the documented plan on the project. 3. Project Assessment and Control: It evaluates the performance of the plan against documented commitments. 4. Project Closure: It provides the project's documentation and products in accordance with contract requirements.
	• **Process 2.** Software Implementation: The purpose of the software implementation process is the systematic performance of the analysis, design, construction, integration, and tests activities for new or modified software products according to the specified requirements. • **Inception Phases** • **Activities 2:** 1. Software Implementation Initiation: It ensures that the project plan established in project planning activity is committed to by the work team. • **Elaboration Phases** • **Activities 2:** 2. Software Requirements Analysis: It analyzes the agreed customer's requirements and establishes the validated project requirements. 3. Software Architectural and Detailed Design: It transforms the software requirements to the system software architecture and software detailed design • **Construction-Deployment Phase** • **Activities 2:** 4. Software Construction: It develops the software code and data from the software design. 5. Software Integration and Tests: It ensures that the integrated software components satisfy the software requirements. 6. Product Delivery: It provides the integrated software product to the customer.
Products (22)	• **Process 1.** • Project Management. • Input products: **P.17:** Statement of Work. **P.14:** Software Configuration, **P.2:** Change Request. • Internal products: **P.2:** Change Request. **P.3:** Correction Register. **P.5:** Meeting Record. **P.21:** Verification Results. **P.7:** Progress Status Record. **P.10:** Project Repository Backup. • Output products: **P.8**: Project Plan. **P.1:** Acceptance Record. **P.9:** Project Repository. **P.5:** Meeting Record. **P.14:** Software Configuration.
	• **Process 2.** • Software Implementation. • Inception Phases:**P.8:** Project Plan (input). **P.9:** Project Repository (input). • Elaboration Phases:**P.22:** Validation Results (internal). **P.21:** Verification Results(internal). • Construction-Deployment Phase:**P.2:** Change Request (output). **P.14:** Software Configuration. **P.11:** Requirements Specification. **P.15:** Software Design. **P.20:** Traceability Record. **P.13:** Software Components. **P.12:** Software. **P.18:** Test Cases and Test Procedures. **P.19:** Test Report. **P.6:** Product Operation Guide. **P.16:** Software User Documentation. **P.4:** Maintenance Documentation (output).

Table 4 reports the Pro-Forma of the ISO/IEC 29110 standard-basic profile with 7 roles, 4 activities de project management, and 3-6 phases, activities software implementation (grouped in three categories named inception, elaboration, and construction-deployment phases), and 22 products.

4. Analysis and Discussion

The review of the four SDLCs for BDAS and their analysis regarding their conformance with the lightweight SDLC of the ISO/IEC 29110 standard-basic profile is reported in this section. First, the four SDLCs are described in the sub-sections 4.1. CRISP-DM, 4.2. TDSP, 4.3. BDPL, and 4.4. DDSL. Each description reports the main SDLC structure of phases-activities, roles, and products and illustrates an SDLC diagram (direct or derived) from the available public documentation. These reviews are conducted in this research step independently of the expected lightweight SDLC of the ISO/IEC 29110-basic profile. Each review was realized by the first two researchers of this research team and discussed and finally agreed with the other three researchers. Next, each review of the four SDLCs for BDAS was mapped to the Pro-Forma of the SDLC of the ISO/IEC 29110 standard-basic profile (reported in sub-section 3.2. and Appendix A.1.), and the research team assessed a qualitative ordinal scale (from very low, low, moderate, high, to very high) of conformance of each SDLC structural item analyzed. This is reported in the sub-section 4.5. (Table 5).

Additionally, in sub-section 4.5. it is reported an overall evaluation of each SDLC for BDAS analyzed regarding a recent rigor-agility seven-attributes framework of SDLCs [43]. This framework considers seven attributes to visualize the location toward heavyweight SDLC (left-side extreme), lightweight SDLC (central side), or agile SDLC (right-side extreme) of the four analyzed SDLCs for BDAS. This rigor-agility seven-attributes framework utilizes a differential semantic scale ranging from -3 (completely aligned with the rigor-oriented approach) to +3 (completely aligned with the agile-oriented approach) to assess the following attributes: bureaucratic-responsive, rigid-flexible, slow-speedy, hard-simple, sophisticated-lean, heavyweight-lightweight, and mandatory-optional documentation. The description of each pair of rigorous-agile attributes is reported later (Table 6).

4.1. Review of the CRISP-DM SDLC

CRISP-DM [7, 20, 40], an acronym for Cross-Industry Standard Process for Data Mining, is an SDLC for data science projects, and it has served as the basis for other SDLCs [40]. It was developed by an industrial consortium in the European Union and is currently commercialized by the IBM SPSS statistics software tool [20]. The CRISP-DM methodology provides a structured approach to planning a data science project. It is a well-structured and most used practice methodology [40] consisting of 6 sequential phases initially, but that is usually performed in a backward-forward way.

These 6 phases are:

1. Business understanding.
2. Data understanding.
3. Data preparation.
4. Modelling.
5. Evaluation.
6. Deployment.

- **Phase 1.** Business understanding phase starts with an initial data collection and continues with activities to become familiar with the data, identify data quality issues, discover early insights from the data, or detect interesting subsets to form hypotheses about hidden information. In this first phase, there are 4 activities:
- **Activity 1:**
  1. Determine business objectives.
  2. Assess situation.
  3. Determine data science goals.
  4. Produce project plan.
- **Phase 2.** Data understanding phase begins with an initial data collection and continues with activities to become familiar with the data, identify data quality issues, discover early insights from the data, or detect interesting subsets to form hypotheses about hidden information. This phase consists of 4 activities:
- **Activity 2:**
  1. Collect initial data.
  2. Describe data.
  3. Explore data.
  4. Verify data quality.
- **Phase 3.** Data preparation phase encompasses all activities to build the final dataset (data to be fed into the modelling tool(s)) from the initial raw data. Data preparation tasks are likely to be performed several times and not in a pre-set or-der. The tasks include the selection of tables, records, and attributes, as well as the transformation and cleaning of the data for the modelling tools. This phase includes the following five activities:
- **Activity 3:**
  1. Select data.
  2. Clean data.
  3. Construct data.
  4. Integrate data.
  5. Format data.
- **Phase 4.** Modelling, various modelling techniques can be used for the same data science problem, and their parameters are calibrated until optimal values are reached. Due to that, some modelling techniques have specific requirements for the data format;

therefore, it is often necessary to go back to the phase 3, Data Preparation. This phase consists of 4 activities:

- **Activity 4:**

1. Select modelling techniques.
2. Generate test design.
3. Build model.
4. Assess model.

- **Phase 5.** Evaluation phase determines the degree to which the models satisfy the business objectives. At the end of this phase, a business decision must be made on the use of the data science project's results. There are three activities in this phase:

- **Activity 5:**

1. Evaluate results.
2. Review process.
3. Determine next steps.

- **Phase 6.** Deployment phase is only performed when the decision is to implement the models from phase 5. depending on the specific characteristics of the data science project, this deployment phase can range from the production of a simple report to the establishment of a recurring enterprise-wide data science system. There are four activities in this phase:

- **Activity 6:**

1. Plan deployment.
2. Plan monitoring and maintenance.
3. Produce final report.
4. Review project.

Regarding roles and work products, CRISP-DM does not explicitly report roles, but it can be assumed that these are the minimum necessary to apply the CRISP-DM, SDLC: Customer, project manager, and data science development team. regarding work products, crisp-dm SLDC report explicitly 38 work products. CRISP-DM has been used for real-life projects [20, 40]. Figure 3 shows the CRISP-DM SDLC for BDAS.

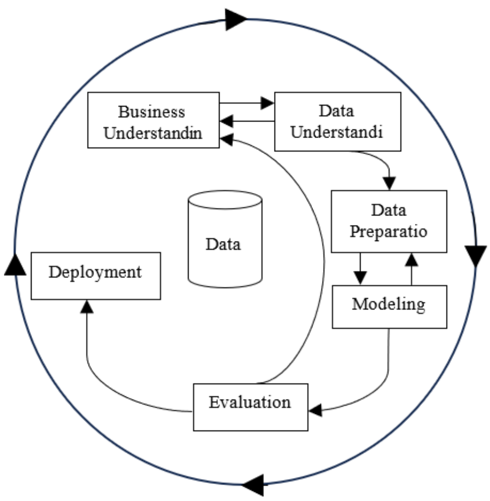

Figure 3. The CRISP-DM SDLC for BDAS [7].

4.2. Review of the Microsoft TDSP SDLC

The Microsoft TDSP [41] aims to provide a standardized SDLC for BDAS. This SDLC is reported as an iterative data science methodology to develop BDAS systematically, and it is self-claimed as agile, and thus a lightweight type, compared with heavyweight former SDLCs such as SEMMA [63] and CRISP-DM [7, 20, 40]. TDSP relies on four main components: a definition of the data science lifecycle, a standardized project structure, recommended infrastructure and resources and recommended tools and utilities needed for project execution. TDSP SDLC is structured with 5 phases:

1. Business understanding.
2. Data acquisition and understanding.
3. Modelling.
4. Deployment
5. Customer acceptance.

Each specific TDSP project can be formed by the following four main roles: Customer, project manager (for the overall managerial coordination of the BDAS development project), project lead (for the technical coordination of the BDAS development project), and project individual contributors (solution architect, data engineer, data scientist, and application developers).

- **Phase 1.** Business understanding focuses on identifying the main variables that serve as objectives of the model, their associated project success metrics, and all relevant data sources required for the BDAS. There are two main activities in this phase:

- **Activity 1:**

1. Define objectives where the BDAS development team interacts with the customer to understand and frame the business problem and to formulate the core questions that the BDAS can help to answer.
2. Identify data sources where the required datasets for the BDAS can help to answer the customer's inquiries are defined.

- **Product:**

1. Work products
   a) Charter document (the project plan).
   b) Data sources (directory of all datasets and associated pre-processing scripts).
   c) Data dictionaries (computational descriptors for all datasets

- **Activity 1:**

Additionally, TDSP documentation reports a TDSP workflow for project execution with three activities:

3. Plan sprint.
4. Review code built from several branches.
5. Merge-delete branches. These activities can be considered project management ones.

- **Phase 2.** Data acquisition and understanding refers to the production, from several pre-processed datasets, a final cleaned and high-quality dataset, as well as to the definition of the data ingestion pipeline (in batch-based, streaming, real-time, or hybrid mode). Three main activities are performed in this phase:
- **Activity 2:**
  1. Ingest the data where datasets are moved from origin locations to the target analysis BDAS environment.
  2. Explore the data where the raw datasets are iteratively explored and processed to produce the target high-quality dataset to be used for the BDAS.
  3. Set up a data pipeline where the data ingestion pipeline architecture (in batch-based, streaming, real-time, or hybrid mode) is specified. This phase 2 produces 3 work products:
- **Product:**
  1. Data quality report (an auditing detailed report on the target high-quality processed dataset);
  2. Solution Architecture (full data pipeline architecture diagram and specifications of the technological platform).
  3. Check-point decision where it is documented the decision to advance to next phase or close the project because the achieved target dataset is insufficient to produce a useful BDAS. In the event the BDAS development project continues, phase 3. Modelling refers to designing and build-test-evaluate the Statistical/Machine Learning model to be implemented considering the available target high-quality dataset. There are three activities in this phase:
- **Activity 3:**
  1. Feature engineering, where the statistical/machine learning model is designed through an iterative process based on the expertise of the data scientists.

In this activity 3.1, TDSP provides a methodological guide to select the most appropriate model (called machine learning algorithm cheat sheet).

2. Model training, where the statistical/machine learning model is calibrated and trained to be ready for processing the new data ingested.
3. Model evaluation, where it is determined whether the calibrated-trained statistical/machine learning model produces results of an adequate level of validity to be released to production utilization. implicit work products generated in phase 3 are:

- **Product:**
  1. Calibrated-trained statistical/machine learning model (already computationally implemented)
  2. Model evaluation report. In phase 4 deployment transfers the designed, built, and tested statistical/machine learning model to its final production environment to be used by the customer. There is a single activity:
- **Activity 4:**
  1. Operationalize the model where the computational model is implemented into the data pipeline in the production environment for its utilization (by online websites, spreadsheets, dashboards, line-of-business applications, or back-end applications). Three work products are elaborated in phase 4.
- **Product:**
  1. Status dashboard that displays the system health and key metrics.
  2. Final Modelling report including model deployment details.
  3. Solution architecture document.
- **Phase 5.** Customer acceptance, refers to the confirmation that the BDAS development project reached the expected objectives and the BDAS control by the customer, with the potential usual support of the IT department, and the project closure. There are two activities in phase 5.
- **Activity 5:**
  1. System validation, where the customer verifies that the BDAS deployed meets the planned objectives.
  2. Project hand-off, where the control of the BDAS is transferred to the organizational area that will run the BDAS in the IT production environment. One work product is elaborated in this phase:
- **Product:**
  1. Exit report of the project for the customer (a technical and business documentation on the full development and user utilization of the BDAS). TDSP is not recommended for very small BDAS projects with a single data scientist, data engineer, application developer team. No academic reference was found on its utilization, but TDSP [41] is promoted by the company Microsoft, and thus, we can infer it has been empirically used in BDAS projects in real-life settings. Figure 4 shows the Microsoft TDSP SDLC for BDAS.

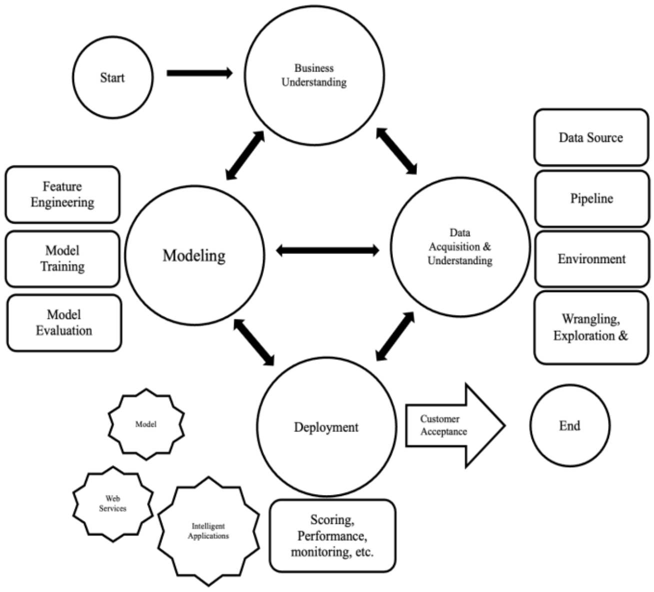

Figure 4. The Microsoft TDSP SDLC for BDAS [41].

4.3. Review of the BDPL SDLC

The BDPL [36] is a heavyweight SDLC for BDAS generated by extending the system and software engineering ISO/IEC 15288 standard [25] from 4 categories of processes-activities including 25 process-activities, to 5 categories, including 43 process-activities. These new 18 new process-activities for the ISO/IEC 15288 standard [25] are organized as follows: 1 new category of processes-activities named data processes with 13 new process-activities, and 5 new process-activities in the categories of agreement processes (with 1 new process-activity named Data Value, Result, and In-novation process), organizational project-enabling process (with 1 new process-activity named domain specialist resource management process), and technical processes (with 3 new process-activities named data automation and monitoring process, data visualization process, and data decision support process). The category of project process-activities was not extended [25]. Consequently, the Project category keeps seven process-activities, the agreement category increases from 2 to 3 process-activities, the organizational project-enabling category passes from 5 to 6 process-activities, and the technical category from 11 to 14 process-activities. Lin and Huang [36] reported that there was no available IEEE standard on an SDLC for BDAS at the date of publication, and the rate of unsuccessful BDAS projects was higher than traditional software systems. Consequently, Lin and Huang [36] elaborated the BDPL considering elements from data variety, data innovation, data analytics, and software engineering and using the ISO/IEC 15288:2008 standard as a basis.

Regarding roles and work products, BDPL does not explicitly report the roles. However, similarly to CRISP-DM, it can be assumed that these are the minimum necessary to apply the BDPL SDLC: Customer, project manager, and data science development team. Regarding work products, although BDPL does not explicitly report them, the original unmodified ISO/IEC 15288 standard proposes about 70 work products. The BDPL SDLC was used for real projects in the banking sector [36]. This third SDLC brings to the research the importance of using SDLC based on ISO/IEC standards to effectively support the development of BDAS projects. Figure 5 shows the BDPL SDLC for BDAS (only the phase and activities added to the ISO/IEC 15288 standard).

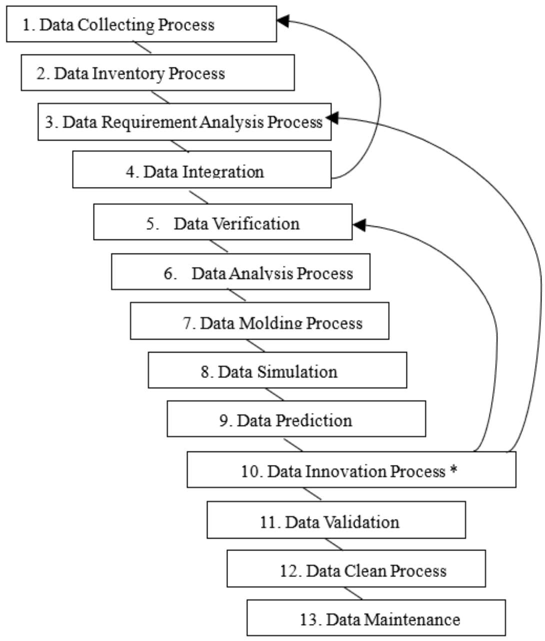

Figure 5. The BPDL SDLC for BDAS [36].

4.4. Review of the DDSL SDLC

The DDSL [12] is a full SDLC for BDAS motivated to provide a modern and light-weight SDLC regarding the former and heavy-weight CRISP-DM SDLC, based on three principles:

1. Modern SDLC for BDAS are highly iterative.
2. Modern SDLCs for BDAS relies on high collaboration between customer, project manager and development team.
3. Modern SDLCs for BDAS need to anticipate auditing and tracking requirements due to the strong organizational implications of the results produced by BDAS.

Then, DDSL [12] is structured in 6 phases:

1. Ideation.
2. Data acquisition and exploration.
3. research and development.
4. Validation.
5. Delivery.
6. Monitoring.

- **Role:**

1. Business stakeholders (customers, users).
2. Data scientists.

3. IT team (data product manager, data story teller, and data infrastructure engineers).

- **Phase 1.** Ideation focuses on establishing the business objectives for the planned BDAS, but a specific business problem must be previously selected, the economic and technical feasibility of the BDAS project is assessed, the BDAS requirements are documented, and the decision to advance to the next stage or abandon the project is made. There are four activities as follows in this phase:
- **Activity 1:**
  1. Project scoping, where business objectives of the BDAS are set up, and economical-technical feasibility is assessed, classifying the project as “sweet spot,” “transformational”, “quick wins” or “don’t just don’t” types.
  2. Proceed decision, where it is agreed to continue or abandon the BDAS development project.
  3. Select artifacts, in the case of BDAS project continuation, where is agreed an overall BDAS architectural solution.
- **Product:**
  1. Requirements documentation.
  2. Overall BDAS architectural Solution (i.e., the final expected deliverable).
- **Phase 2.** Data acquisition and exploration refers to the identification of the required available and non-available datasets, its financial-technical authorization for getting them, and its iterative exploration, pre-processing, and understanding for the next phase.
- **Activity 2:**
  1. Identify datasets where the access to internal datasets must be authorized by the IT department and/or external datasets must be authorized to be bought.
  2. Ingest data where internal and/or external datasets are transferred from original locations to the target BDAS location.
  3. Explore data where iteratively and interactively the datasets are explored to determine the final ones to be used.
  4. Prepare data where pre-processing and processing operations on the final datasets to be used are applied.
- **Product:**

  a) Data dictionary.
- **Phase 3.** Research and development refer to the selection, building, and calibration of the statistical/machine learning model, which also includes the selection of the data science and analytics platforms and tools to be used. There are five activities in phase 3.
- **Activity 3:**
  1. Generate a hypothesis and model where the set of specific inquiries are formulated, and the statistical/machine learning model is selected.
  2. Validate the right platforms and tools where the computational development resources are already available or are requested by the IT department.
  3. Experiment and assess results where the statistical/machine learning model is performed calibrated, and the results are assessed whether they are sufficiently insightful to advance to the next phase or more datasets and experimentation-calibration is required.
- **Product:**

  a) Calibrated statistical/machine learning model.
- **Phase 4.** Validation refers to the business and technical validation of the calibrated statistical/machine learning model to authorize its delivery to production or return to conduct required previous stages or to stop the BDAS development project because this does not reach the business reliability expectations. There are three activities in this phase:
- **Activity 4:**
  1. Business validation, where business stakeholders determine whether the results of the BDAS model are useful and reliable from the business perspective.
  2. Technical validation, where the IT team determines whether the BDAS model is ready for its deployment to production.

There is one work product elaborated in this phase 4, BDAS business and technical validation documentation.

- **Phase 5.** Delivery focuses on becoming the statistical/machine learning model in a “product” usable by customers and users. There are four activities in phase 5.
- **Activity 5:**
  1. Plan delivery, where a detailed plan for deploying the BDAS is conceived (among final modes such as an ad-hoc report, scheduled report, application launcher, web application, batch API, or real-time API).
  2. Deploy where it is applied to the selected deployment mode.
  3. Alpha/Beta Test, where technical internal Alpha and pilot user Beta tests are applied.
  4. User acceptance test, where an official group of users verifies the acceptance of the BDAS. In this, it is not expected that Users reject the BDAS, i.e., if the BDAS development project reached this activity because it is a satisfactory product.

- **Product:**
  1. Monitoring and training plan.
  2. Tests documentations (Alpha, Beta, and user acceptance types).
- **Phase 6.** Monitoring, refers to the periodic supervision and evaluation of the usage, technical performance, and created value of the BDAS. This phase has two activities.
- **Activity 6:**
  1. Supervise and evaluate usage and performance where usual IT service managerial metrics can be applied to keep the BDAS usable.
  2. Evaluate value where business stakeholders determine the overall and specific contributions of the BDAS to the business value.

Additionally, business stakeholders can propose improvements to the BDAS for the next version of the installed and used product. This phase generates two work products:

- **Product:**
  1. Periodical usage and performance evaluation report.
  2. Overall value evaluation report. Similarly to Microsoft TDSP [41], no academic reference was found on its utilization, but domino DSL [12] is promoted by the company domino Inc., and thus, we can infer it has been empirically used in BDAS projects in real-life settings. Figure 6 shows the domino DSL SDLC for BDAS.

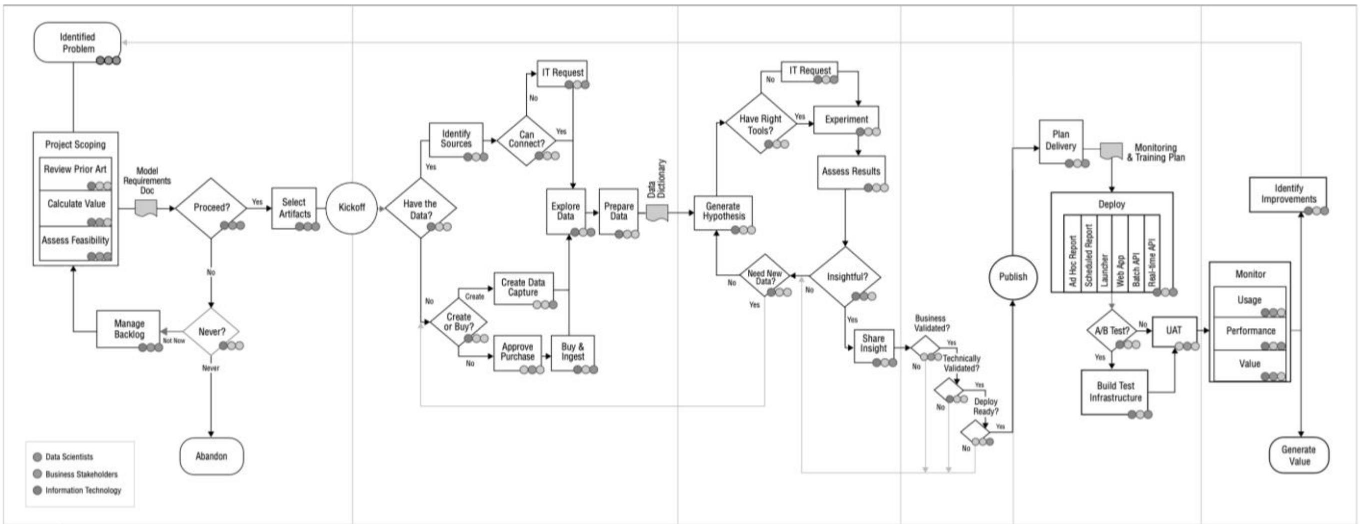

Figure 6. The DDSL SDLC for BDAS [12].

4.5. Integrated Analysis and Discussion on the three SDLCs for BDAS against the Lightweight SDLC of the ISO/IEC 29110 Standard-Basic Profile

We conducted a detailed analysis of the structure and content of the roles, phases-activities, and products proposed in the four SDLCs to populate four descriptive Table (Appendix A.1.) on such SDLC structures. Next, we conducted an iterative analysis to map the structure of roles, phases-activities, and products identified in each SDLC to the expected ones in the theoretical Pro-Forma SDLC of the used ISO/IEC 29110 standard-basic profile. With this detailed analysis-mapping for the three SDLCs, we elaborated during several iterative cycles of analysis, discussion and agreements, a visual evaluative summary of the qualitative extent of conformance of each structural component of the reviewed-analyzed SDLCs for BDAS to the expected one in the theoretical Pro-Forma SDLC of the ISO/IEC standard-basic profile. The following qualitative scale of alignment and adherence of the analyzed SDLC and the theoretical SDLC of the ISO/IEC 29110 standard Pro-Forma was used:

- Low level (1 point). This corresponds to a cell shaded in light grey when the analyzed SDLC contains relevant omissions regarding the expected content on roles, categories of phases-activities, or categories of work products (packages) of the theoretical SDLC for BDAS.
- Moderate level (3 points). This corresponds to a grey cell when the analyzed SDLC contains slight omissions regarding the expected content on roles, categories of phases-activities, or categories of work products (packages) of the theoretical SDLC for BDAS.
- High level (5 points). This corresponds to a dark grey cell when the analyzed SDLC contains relevant similarities regarding the expected content on roles, categories of phases-activities, or categories of work products (packages) of the theoretical SDLC for BDAS. Then, the visual summarized evaluation of conformance of the SDLC structure of the four analyzed SDLCs for BDAS (CRISP-DM, TDSP, BPDL, and DDSL) is reported in Table 5.

Table 5. Summary of the conformance evaluation of the SLDCs for BDAS versus the theoretical SDLC for BDAS.

SDLC element	Theoretical lightweight SDLC Pro-Forma for BDAS of the ISO/IEC 29110-basic profile	CRISP-DM t	BDPL	TDSP	DDSL
Roles (7)	User roles: 1. Management roles: 1. Technical roles: 5.	Moderate	Moderate	Moderate	High
	Overall evaluation of roles:	**Roles: Moderate**	**Roles: Moderate**	**Roles: Moderate**	**Roles: High**
Process-activities (2, 10)	Process 1. Project management: (with 4 activities)	High	High	Very Low	High
	Process 2. Software implementation: Inception phases: (with 1 activities)	High	High	Low	High
	Process 2. Software implementation: Elaboration phases: (with 2 activities)	Moderate	High	Very Low	High
	Process 2. Software implementation: Construction-deployment phase: (with 3 activities)	High	High	Moderate	Moderate
	Overall evaluation of phases-activities:	**Phase-activities: High**	**Phase-activities: High**	**Phase-activities: Low**	**Phase-activities: High**
Products (22)	Process 1. Project management: (with 11 products and 1 SSP)	High	Very Low	Very Low	Moderate
	Process 2. Software implementation: Inception phases: (with 2 products and 1 SSP)	High	Very Low	Low	Moderate
	Process 2. Software Implementation: Elaboration Phases: (with 6 products and 1 SSP)	High	Very Low	Very Low	Moderate
	Process 2. Software implementation: Construction-deployment phase: (with 7 products and 1 SSP)	High	Very Low	Moderate	Moderate
	Overall evaluation of products:	**Products: High**	**Products: Very Low**	**Products: Low**	**Products: Moderate**
	Overall evaluation of the SDLC:	**Overall SDLC: High**	**Overall SDLC: Moderate**	**Overall SDLC: Low**	**Overall SDLC: High**

Table 6. Summary of the evaluation on the rigor-agility framework for the four SLDCs for BDAS.

<table>
<tr><th rowspan="3">Rigor attributes</th><th colspan="7">Level assigned to the SDLC for BDAS</th><th rowspan="3">Agility attributes</th></tr>
<tr><th colspan="2">Zones of rigorous SDLCs</th><th colspan="3">Zones of lightweight SDLCs</th><th colspan="2">Zones of agile SDLCs</th></tr>
<tr><th>-3 Very High</th><th>-2 High</th><th>-1 Low</th><th>0 Neutral</th><th>+1 Low</th><th>+2 High</th><th>+3 Very High</th></tr>
<tr><td>Rigid: To keep and apply BDAS practices without any variation.</td><td rowspan="7">BDPL</td><td rowspan="7">CRISP-DM</td><td rowspan="7">-</td><td rowspan="7">ISO/IEC 29110</td><td rowspan="7">DDSL</td><td rowspan="7">TDSP</td><td rowspan="7">-</td><td>Flexible: To reconfigure BDAS practices when necessary.</td></tr>
<tr><td>Bureaucratic: to ignore unexpected events during the BDAS development process accepting potential negative consequences.</td><td>Responsive: To sense environment and react appropriately to unexpected events during the BDAS development process.</td></tr>
<tr><td>Slow: To deliver a useable BDAS in relatively large periods.</td><td>Speedy: To deliver quickly a useable BDAS.</td></tr>
<tr><td>Sophisticated: To pursue the best designed and built BDAS.</td><td>Lean: To pursue a minimum viable BDAS (that could be incremented in next releases).</td></tr>
<tr><td>Hard: High cognitive load and high training effort to be learned and used.</td><td>Simple: Low cognitive load and low training effort to be learned and used.</td></tr>
<tr><td>Heavyweight: High volume of practices.</td><td>Lightweight: Shortened practices from the original heavyweight practices but still considered useful for agile domains.</td></tr>
<tr><td>Mandatory documentation: It demands the fulfilment of mandatory technical and user documentation.</td><td>Optional documentation: It permits the fulfilment of technical and user documentation, when necessary.</td></tr>
</table>

This qualitative evaluation was conducted by the first two researchers and revised by the fourth and fifth ones during several iterations based on the full content reported in the original sources. In this iterative analysis were identified, corrected, and agreed on content omissions, differences in interpretation, and typographical errors in the nomenclature of phases-activities, roles, and products. Similarly, the research team evaluated each SDLC for BDAS against the theoretical rigor-agility seven-attributes framework, and these results are reported in Table 6. Table 6 also includes the ISO/IEC 29110 standard-basic profile placed in the neutral rigor-agility column (0.0 score). Therefore, we can summarize the following strengths and weaknesses based on the results reported in Tables 5 and 6 for the three SDLCs for BDAS as follows:

- **Strengths:**
  - The four SDLCs share the claim about the need to have a specific SDLC for BDAS instead of using a generic SDLC.
  - The four SDLCs have been used in real BDAS projects.
  - CRISP-DM reached a high overall conformance level, but its SDLC rigor-agility level (-2.0) implies that it covers satisfactory the expected structure of the theoretical Pro-Forma SDLC of the ISO/IEC 29110 standard-basic profile but in excess, so elimination of elements would be required to fit practically the ISO/IEC 29110

standard.

- TDSP reached a low overall conformance level, and its SDLC rigor-agility level (+2.0) implies that this qualifies more as an agile SDLC than a lightweight one. Then, it would require a relevant methodological effort to add elements required to fit practically the ISO/IEC 29110 standard.
- BPDL also reached a moderate overall conformance level, but its SDLC rigor-agility level (-3.0) implies that this is the most heavyweight SDLC for BDAS of the four analyzed. Thus, it is far away to be adapted to fit practically the ISO/IEC 29110 standard-basic profile.
- DDSL also reached a high overall conformance level (similar to CRISP-DM), but its rigor-agility level (+1.0) is located in the zone of the lightweight SDLCs (-1.0, 0.0 and +1.0), and thus DDSL qualifies as the best SDLC for BDAS to be adjusted to fit the ISO/IEC 29110 standard-basic profile regarding the other three SDLCs analyzed.
- CRISP-DM was found to be the most still used SDLC for BDAS projects, and thus BDAS developers interested in the ISO/IEC 29110 standard could use it and make adequations, but it will demand more methodological effort than required for adjusting the DDSL SDLC.

- **Weaknesses:**
  - The most widely used SDLC for BDAS (CRISP-DM) is already more than 20 years old and qualifies as a heavyweight SDLC with a rigor-agility level of -2.0. Thus, agile BDAS developers cannot use it in your original structure.
  - The four SDLCs are proprietary, public documentation is limited, and usually, it presents inconsistencies in nomenclature, missed information, and a lack of practical work product templates. Thus, BDAS developers must propose ad-hoc adjustments.

5. Conclusions

In this research, we applied a research method using a selective manual search of the SDLCs for BDAS collected in the main literature [18, 29, 31, 38, 39, 62] on the basic BDAS and software engineering literature regarding the availability of SDLCs for BDAS. This selection and analysis ended up with a total of four SDLCs for BDAS (CRISP-DM, TDSP, BDPL, and DDSL), and the four ones were reviewed and evaluated against a theoretical SDLC extracted from the ISO/IEC 29110 standard-basic profile. This review and evaluation were performed by the research team (composed of a PhD student, three full-time professors in the software engineering discipline, and one full-time professor in the analytics data science discipline).

Based on the results obtained (Tables 5 and 6), the following theoretical and practical conclusions can be made:

- **Theoretical Conclusion**
  1. Research on new SDLCs for BDAS has been practically null in the software engineering discipline in the 2000-2023 period (only one SDLC for BDAS was found BDPL) in the literature consulted. The other three SDLCs for BDAS were in the grey literature.
  2. The four analyzed SDLCs (CRISP-DM, TDSP, BDPL, and DDSL) are proprietary, and their public free-access documentation is limited. CRISP-DM and DDSL were evaluated with HIGH conformance levels, but the analysis is conducted on the structural content of the SDLC on roles, phases-activities, and artifacts, but their full-documented descriptions are incomplete.
  3. Although the main literature consulted [18, 29, 31, 38, 39, 62] on BDAS is adequately reported, and the topic is still relevant to business organizations today, we did not find an SDLC that can be considered a de-facto standard as RUP was for software systems for two decades.

- **Practical conclusion**
  1. For BDAS developers interested in using a heavyweight SDLC, CRISP-DM is considered the most widely used and to be potentially converted into the de facto standard. BDPL is not recommended despite its reliance on a well-known, used, and tested ISO/IEC 15288 standard due to its public documentation being very limited (this development cycle is proprietary).
  2. BDAS developers interested in using a lightweight SDLC that fits the ISO/IEC 29110 standard-basic profile the recommendation is to use the DDSL SDLC, but it will be required to add and adapt several technical adjustments.
  3. For BDAS developers interested in agile approaches, the recommended SDLC is TDSP.

- Based on the results obtained (Tables 5 and 6), the following recommendations for future research can also be made:
- Research on rigorous SDLCs for BDAS is not encouraged, given the interest and need for lightweight and agile approaches at present.
- Conceptual research on lightweight SDLCs for BDAS is encouraged to move towards an SDLC for BDAS that directly fits the ISO/IEC 29110 standard-basic profile without the need to make minor adjustments, that can be accepted and endorsed by the academic community.
- Both conceptual and empirical research on specific types of BDAS projects adequate for lightweight

SDLCs vs. agile SDLCs is required in the software engineering discipline.
- To advance research on ISO/IEC standards as a basis for future SDLCs for BDAS.

Finally, we report the following methodological limitations of our study:

- This was focused only on the top 5 reported in at least 1 of the 3 comprehensive articles on big data development cycle, and one development cycle was discarded as it did not meet the characteristics of a development cycle.
- Only lightweight big data cycles were considered, and as a historical reference, the main methodology still used CRISP-DM (heavy type). Agile methodologies are not considered in this study.
- This study uses the ISO/IEC 29110-basic profile as a conceptual framework for analysis, but future studies may consider other standards.
- This study analyzed the four methodologies using exclusively the original materials in a conceptual manner without adding empirical evidence of their use by practitioners.
- The conceptual analysis was performed by a research team composed of: 1 senior doctoral student; 3 senior professors in the area of software engineering (2 specialized in the ISO/IEC 29110 standard); and 1 professor specialized in data science, with an average combined academic and research experience of 13 years. We believe that a research team with similar demographic characteristics would reach similar conclusions.

Hence, we can indicate that there is a need to achieve better lightweight SDLCs for BDAS that can be supported theoretically and used in practice (i.e., with high levels of usability, ease of use, compatibility, and perceived value by BDAS developers) for the small and very small organizations interested in ISO/IEC certifications. In another similar study, with a specifically agile and Scrum-XP focus, the conclusions were comparable, revealing the need for better agile SDLCs for BDAS that can be theoretically grounded and practically applied [59]. Therefore, further conceptual, and empirical research is encouraged in these relevant research streams.

References

[1] Ajah I. and Nweke H., "Big Data and Business Analytics: Trends, Platforms, Success Factors and Applications," *Big Data and Cognitive Computing*, vol. 3, no. 2, pp. 1-30, 2019. https://doi.org/10.3390/bdcc3020032

[2] Andoh-Baidoo F., Baker E., Susarapu S., and Kasper G., "A Review of IS Research Activities and outputs Using Pro-Forma Abstracts," *Information Resources Management Journal*, vol. 20, no. 4, pp. 65-79, 2007. https://www.igi-global.com/article/review-research-activities-outputs-using/1327

[3] Andoh-Baidoo F., Chavarria J., Jones M., Wang Y., and Takieddine S., "Examining the State of Empirical Business Intelligence and Analytics Research: A Poly-Theoretic Approach," *Information and Management*, vol. 59, no. 6, pp. 103677, 2022. https://doi.org/10.1016/j.im.2022.103677

[4] Beulke D., "Big Data Impacts Data Management: The 5 vs of Big Data, https://davebeulke.com/big-data-impacts-data-management-the-five-vs-of-big-data/, Last Visited, 2024.

[5] Boehm B. and Turner R., "Using Risk to Balance Agile and Plan-Driven Methods," *Computer*, vol. 36, no. 6, pp. 57-66, 2003. https://ieeexplore.ieee.org/document/1204376

[6] Bourque P. and Fairly R., *Guide to the Software Engineering Body of Knowledge, SWEBOK Version 3.0*, IEEE Computer Society, 2014. https://ieeecs-media.computer.org/media/education/swebok/swebok-v3.pdf

[7] Chapman P., Clinton J., Kerber R., Khabaza T., Reinartz T., Shearer C., and Wirth R., *CRISP-DM 1.0-Step-by-Step Data Mining Guide*, SPSS Inc, 2000. https://www.kde.cs.uni-kassel.de/wp-content/uploads/lehre/ws2012-13/kdd/files/CRISPWP-0800.pdf

[8] Clarke P. and O'Connor R., "An Empirical Examination of the Extent of Software Process Improvement in Software SMEs," *Journal of Software: Evolution and Process*, vol. 25, no. 9, pp. 981-998, 2013. https://doi.org/10.1002/smr.1580

[9] CMMI Institute, CMMI for Development v2.0, https://cmmiinstitute.com/products/cmmi/cmmiv2-products, Last Visited, 2024.

[10] Davenport T. and Bean R., "The Quest to Achieve Data-Driven Leadership: A Progress Report on the State of Corporate Data Initiatives-Foreword," *Special Report*, New Advantage Partners, 2022. https://c6abb8db-514c-4f5b-b5a1-fc710f1e464e.filesusr.com/ugd/e5361a_2f859f3457f24cff9b2f8a2bf54f82b7.pdf

[11] Davenport T. and Malone K., "Deployment as a Critical Business Data Science Discipline," *Harvard Data Science Review*, vol. 3, no. 1, pp. 1-11, 2021. https://doi.org/10.1162/99608f92.90814c32

[12] Domino Data Lab, The Practical Guide to Managing Data Science at Scale, Domino Data Lab, https://domino.ai/resources/managing-data-science, Last Visited, 2024.

[13] Ebert C., "The Impacts of Software Product Management," *Journal of Systems and Software*, vol. 80, no. 6, pp. 850-861, 2007. https://doi.org/10.1016/j.jss.2006.09.017

[14] Fayyad U., Haussler D., and Stolorz P., "KDD for Science Data Analysis: Issues and Examples," *in Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, Portland, pp. 50-56, 1996. https://cdn.aaai.org/KDD/1996/KDD96-009.pdf

[15] Fayyad U., Piatetsky-Shapiro G., and Smyth P., "The KDD Process for Extracting Useful Knowledge from Volumes of Data," *Communications of the ACM*, vol. 39, no. 11, pp. 27-34, 1996. https://doi.org/10.1145/240455.240464

[16] Giray G., "A Software Engineering Perspective on Engineering Machine Learning Systems: State of the Art and Challenges," *Journal of Systems and Software*, vol. 180, pp. 111031, 2021. https://doi.org/10.1016/j.jss.2021.111031

[17] Gray E., Jennings W., Farrall S., and Hay C., "Small Big Data: Using Multiple Data-Sets to Explore Unfolding Social and Economic Change," *Big Data and Society*, vol. 2, no. 1, pp. 1-6, 2015. https://doi.org/10.1177/2053951715589418

[18] Haakman M., Cruz L., Huijgens H., and Van Deursen A., "AI Lifecycle Models Need to be Revised: An Exploratory Study in Fintech," *Empirical Software Engineering*, vol. 26, pp. 1-29, 2021. https://link.springer.com/article/10.1007/s10664-021-09993-1

[19] Humphrey W., The Team Software Process (TSP), Software Engineering Institute, Technical Report, 2000. https://resources.sei.cmu.edu/asset_files/TechnicalReport/2000_005_001_13754.pdf

[20] IBM, IBM SPSS Modeler CRISP-DM Guide, https://www.ibm.com/docs/zh/spss-modeler/18.0.0?topic=spss-modeler-crisp-dm-guide, Last Visited, 2024.

[21] Iranmanesh M., Lim K., Foroughi B., Hong M., and Ghobakhloo M., "Determinants of Intention to Adopt Big Data and Outsourcing among SMEs: Organisational and Technological Factors as Moderators," *Management Decision*, vol. 61, no. 1, pp. 201-222, 2023. DOI:10.1108/MD-08-2021-1059

[22] ISO/IEC, *Information Technology-Process Assessment-Requirements for Process Reference, Process Assessment and Maturity Models ISO/IEC 33004*, ISO/IEC, 2015. https://www.iso.org/standard/54178.html

[23] ISO/IEC, *Software Engineering-Lifecycle Profiles for Very Small Entities (VSEs)-Part 5-1-2: Management and Engineering Guide-Generic Pro-File Group: Basic Profile ISO/IEC TR 29110-5-1-2*, IEEE, 2011. https://www.iso.org/standard/51153.html

[24] ISO/IEC, *Systems and Software Engineering-Software Life Cycle Processes, ISO/IEC/IEEE 12207*, IEEE, 2017. https://www.iso.org/standard/63712.html

[25] ISO/IEC, *Systems and Software Engineering-System Life Cycle Processes ISO/IEC15288*, IEEE Std, 2008. https://cdn.standards.iteh.ai/samples/43564/3f3539f541e3448c9d24fa752859d0b0/ISO-IEC-15288-2008.pdf

[26] ISO/IEC/IEEE, *Systems and Software Engineering-Vocabulary*, IEEE, 2017. https://www.iso.org/standard/71952.html

[27] Kitchin R. and Lauriault T., "Small Data in the Era of Big Data," *GeoJournal*, vol. 80, no. 4, pp. 463-475, 2015. https://www.jstor.org/stable/44076310

[28] Kruchten P., *Rational Unified Process-best Practices for Software Development Teams*, Rational Company, 2014. http://srprojects.free.fr/desgest/downloads2/Rational_Unified_Process_Best_Practices.pdf

[29] Kumar V. and Alencar P., "Software Engineering for Big Data Projects: Domains, Methodologies and Gaps," *in Proceedings of the IEEE International Conference on Big Data*, Washington (DC), pp. 2886-2895, 2016. DOI:10.1109/BigData.2016.7840938

[30] Kune R., Konugurthi P., Agarwal A., Chillarige R., and Buyya R., "The Anatomy of Big Data Computing," *Software: Practice and Experience*, vol. 46, no. 1, pp. 79-105, 2016. https://onlinelibrary.wiley.com/doi/10.1002/spe.2374

[31] Laigner R., Kalinowski M., Lifschitz S., Monteiro R., and De Oliveira D., "Systematic Mapping of Software Engineering Approaches to Develop Big Data Systems," *in Proceedings of the 44th Euromicro Conference on Software Engineering and Advanced Applications*, Prague, pp. 446-453, 2018. DOI:10.1109/SEAA.2018.00079

[32] Laporte C. and O'Connor R., "Software Process Improvement Standards and Guides for very Small Organization: An Overview of Eight Implementations," *CrossTalk, The Journal of Defense Software Engineering*, vol. 30, no. 3, pp. 23-27, 2017. https://doras.dcu.ie/21798/

[33] Laporte C. and O'Connor R., "Systems and Software Engineering Standards for Very Small Entities: Accomplishments and Overview," *Computer*, vol. 49, no. 8, pp. 84-87, 2016. DOI:10.1109/MC.2016.242

[34] Laporte C., O'Connor R., and Fanmuy G., "International Systems and Software Engineering Standards for Very Small Entities," *CrossTalk, The Journal of Defense Software Engineering*, vol. 26, no. 3, pp. 28-33, 2013. https://hdl.handle.net/10344/3083

[35] Larson D. and Chang V., "Review and Future Direction of Agile, Business Intelligence, Analytics and Data Science," *International*

Journal of Information Management, vol. 36, no. 5, pp. 700-710, 2016. https://doi.org/10.1016/j.ijinfomgt.2016.04.013

[36] Lin Y. and Huang S., "The Design of a Software Engineering Lifecycle Process for Big Data Projects," *IT Professional*, vol. 20, no. 1, pp. 45-52, 2018. DOI: 10.1109/MITP.2018.011291352

[37] Lukoianova T. and Rubin V., "Veracity Roadmap: Is Big Data Objective, Truthful and Credible?," *Advances in Classification Research Online*, vol. 24, no. 1, pp. 4-15, 2014. https://doi.org/10.7152/acro.v24i1.14671

[38] Madhavji N., Miranskyy A., and Kontogiannis K., "Big Picture of Big Data Software Engineering: with Example Research Challenges," *in Proceedings of the IEEE/ACM 1st International Workshop on Big Data Software Engineering*, Florence, pp. 11-14, 2015. DOI:10.1109/BIGDSE.2015.10

[39] Martinez I., Viles E., and Olaizola I., "Data Science Methodologies: Current Challenges and Future Approaches," *Big Data Research*, vol. 24, pp. 100183, 2021. https://doi.org/10.1016/j.bdr.2020.100183

[40] Martinez-Plumed F., Contreras-Ochando L., Ferri C., Hernendez-Orallo J., Kull M., Lachiche N., and Flach P., "CRISP-DM Twenty Years Later: From Data Mining Processes to Data Science Trajectories," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 8, pp. 3048-3061, 2021. DOI:10.1109/TKDE.2019.2962680

[41] Microsoft, What is the Team Data Science Process?, https://docs.microsoft.com/en-us/azure/machine-learning/team-data-science-process/overview, Last Visited, 2024.

[42] Montoya-Murillo D., Mora M., Galvan-Cruz S., and Munoz-Zavala A., *Development Methodologies for Big Data Analytics Systems: Plan-Driven, Agile, Hybrid, Lightweight Approaches*, Springer, 2023. https://link.springer.com/chapter/10.1007/978-3-031-40956-1_5

[43] Mora M., Adelakun O., Reyes-Delgado P., and Diaz O., "AVS_FD_MVITS: An Agile IT Service Design Workflow for Small Data Centers," *The Journal of Supercomputing*, vol. 79, pp. 17519-17561, 2023. https://link.springer.com/article/10.1007/s11227-023-05244-w

[44] Mora M., Reyes-Delgado P., Galvan-Cruz S., and Solano-Romo L., *Development Methodologies for Big Data Analytics Systems: Plan-Driven, Agile, Hybrid, Lightweight Approaches*, Springer, 2023. https://link.springer.com/chapter/10.1007/978-3-031-40956-1_1

[45] Munoz M., Pena A., Mejia J., Gasca-Hurtado G., Gomez-Alvarez M., and Laporte C., "Analysis of 13 Implementations of the Software Engineering Management and Engineering Basic Profile Guide of ISO/IEC 29110 in Very Small Entities Using Different Life Cycles," *Journal of Software: Evolution and Process*, vol. 32, no. 11, pp. 1-26, 2020. https://onlinelibrary.wiley.com/doi/abs/10.1002/smr.2300

[46] Niazi M., "A Comparative Study of Software Process Improvement Implementation Success Factors," *Journal of Software: Evolution and Process*, vol. 27, no. 9, pp. 700-722, 2015. https://doi.org/10.1002/smr.1704

[47] O'Connor R. and Coleman G., "Ignoring "Best Practice": Why Irish Software SMEs are Rejecting CMMI and ISO 9000," *Australasian Journal of Information Systems*, vol. 16, no. 1, pp. 7-30, 2009. https://ajis.aaisnet.org/index.php/ajis/article/view/557/441

[48] O'Connor R. and Laporte C., "The Evolution of the ISO/IEC 29110 Set of Standards and Guides," *International Journal of Information Technologies and Systems Approach*, vol. 10, no. 1, pp. 1-21, 2017. https://www.igi-global.com/gateway/article/169765

[49] Oussous A., Benjelloun F., Lahcen A., and Belfkih S., "Big Data Technologies: A Survey," *Journal of King Saud University-Computer and Information Sciences*, vol. 30, no. 4, pp. 431-448, 2018. https://doi.org/10.1016/j.jksuci.2017.06.001

[50] Paakkonen P. and Pakkala D., "Reference Architecture and Classification of Technologies, Products and Services for Big Data Systems," *Big Data Research*, vol. 2, no. 4, pp. 166-186, 2015. https://doi.org/10.1016/j.bdr.2015.01.001

[51] Pai D., Subramanian G., and Pendharkar P., "Benchmarking Software Development Productivity of CMMI Level 5 Projects," *Information Technology and Management*, vol. 16, no. 3, pp. 235-251, 2015. https://link.springer.com/article/10.1007/s10799-015-0234-4

[52] Phillips-Wren G., Daly M., and Burstein F., "Reconciling Business Intelligence, Analytics and Decision Support Systems: More Data, Deeper Insight," *Decision Support System*, vol. 146, pp. 113560, 2021. https://doi.org/10.1016/j.dss.2021.113560

[53] Plotnikova V., Dumas M., and Milani F., "Adaptations of Data Mining Methodologies: A Systematic Literature Review," *PeerJ Computer Science*, vol. 25, no. 6, pp. 1-43, 2020. https://pmc.ncbi.nlm.nih.gov/articles/PMC7924527/

[54] Pollack J., Helm J., and Adler D., "What is the Iron Triangle, and How has it Changed?," *International Journal of Managing Projects in Business*, vol. 11, no. 2, pp. 527-547, 2018. https://doi.org/10.1108/IJMPB-09-2017-0107

[55] Ransbotham S., Khodabandeh S., Kiron D., Candelon F., Chu M., and LaFountain B., "Expanding AI's Impact with Organizational

Learning," *MIT Sloan Management Review and Boston Consulting Group*, pp. 1-15, 2020. https://www.hbs.edu/faculty/Pages/item.aspx?num=63842

[56] Rao T., Mitra P., Bhatt R., and Goswami A., "The Big Data System, Components, Tools, and Technologies: A Survey," *Knowledge and Information Systems*, vol. 60, no. 3, pp. 1165-1245, 2019. https://link.springer.com/article/10.1007/s10115-018-1248-0

[57] Reyes-Delgado P., Mora M., Wang F., and Gomez J., "AHP Evaluation of Rigorous and Agile IT Service Design-Building Phases-Workflows in Data Centers," *The Journal of Supercomputing*, vol. 79, no. 16, pp. 18089-18166, 2023. https://doi.org/10.1007/s11227-023-05219-x

[58] Russom P., "Big Data Analytics," *TDWI best Practices Report, Fourth Quarter*, 2011. http://download.101com.com/pub/tdwi/Files/TDWI_BPReport_Q411_Big_Data_Analytics_Web.pdf

[59] Salazar-Salazar G., Mora M., Duran-Limon H., Alvarez-Rodriguez F., and Munoz-Zavala A., "Review of Agile SDLC for Big Data Analytics Systems in the Context of Small Organizations Using Scrum-XP," *The International Arab Journal of Information Technology*, vol. 21, no. 6, pp. 1089-1110, 2024. https://doi.org/10.34028/iajit/21/6/12

[60] Salazar-Salazar G., Mora M., Duran-Limon H., and Rodríguez F., *Development Methodologies for Big Data Analytics Systems: Plan-Driven, Agile, Hybrid, Lightweight Approaches*, Springer, 2023. https://doi.org/10.1007/978-3-031-40956-1_6

[61] Saltz J. and Iva K., "Current Approaches for Executing Big Data Science Projects-A Systematic Literature Review," *PeerJ Computer Science*, vol. 8, pp. 1-24, 2022. https://doi.org/10.7717/peerj-cs.862

[62] Saltz J. and Shamshurin I., "Big Data Team Process Methodologies: A Literature Review and the Identification of Key Factors for a Project's Success," *in Proceedings of the IEEE International Conference on Big Data*, Washington (DC), pp. 2872-2879, 2016. DOI:10.1109/BigData.2016.7840936

[63] SAS Institute, Introduction to SEMMA, https://documentation.sas.com/doc/en/emref/14.3/p1tsqq44rg56ron17qd3m7ey4mzu.htm, Last Visited, 2024.

[64] Schryen G., "Writing Qualitative is Literature Reviews-Guidelines for Synthesis, Interpretation, and Guidance of Research," *Communications of the Association for Information Systems*, vol. 37, no. 1, pp. 286-325, 2015. https://doi.org/10.17705/1CAIS.03712

[65] Staples M., Niazi M., Jeffery R., Abrahams A., Byatt P., and Murphy R., "An Exploratory Study of Why Organizations do not Adopt CMMI," *Journal of Systems and Software*, vol. 80, no. 6, pp. 883-895, 2007. https://doi.org/10.1016/j.jss.2006.09.008

[66] Sutherland J., *The Scrum Handbook*, The Scrum Training Institute Press, 2016. https://www.scruminc.com/wp-content/uploads/2014/07/The-Scrum-Handbook.pdf

[67] Todman L., Bush A., and Hood A., "'Small Data' for Big Insights in Ecology," *Trends in Ecology and Evolution*, vol. 38, no. 7, pp. 615-622, 2023. https://doi.org/10.1016/j.tree.2023.01.015

[68] Tsai C., Lai C., Chao H., and Vasilakos A., "Big Data Analytics: A Survey," *Journal of Big Data*, vol. 2, no. 1, pp. 1-32, 2015. DOI:10.1186/s40537-015-0030-3

[69] Tsoy M. and Staples D., "What are the Critical Success Factors for Agile Analytics Projects?," *Information Systems Management*, vol. 38, no. 4, pp. 324-341, 2021. https://doi.org/10.1080/10580530.2020.1818899

[70] Unterkalmsteiner M., Gorschek T., Islam A., Cheng C., Permadi R., and Feldt R., "Evaluation and Measurement of Software Process Improvement-A Systematic Literature Review," *IEEE Transactions on Software Engineering*, vol. 38, no. 2, pp. 398-424, 2012. https://ieeexplore.ieee.org/document/5728832

[71] Vives L., Melendez K., and Davila A., "ISO/IEC 29110 and Software Engineering Education: A Systematic Mapping Study," *Programming and Computer Software*, vol. 48, no. 8, pp. 745-755, 2022. https://link.springer.com/article/10.1134/S0361768822080229

[72] Watson H., "Update Tutorial: Big Data Analytics: Concepts, Technology, and Applications," *Communications of the Association for Information Systems*, vol. 44, no. 1, pp. 364-379, 2019. https://aisel.aisnet.org/cgi/viewcontent.cgi?article=4127&context=cais

David Montoya-Murillo is a computer systems engineer who graduated from the Autonomous University of Aguascalientes. He studied for a Master's Degree in Informatics and Computer Technology at the same institution. Ph.D. Student in Applied and Technological Sciences, with a focus on Lightweight Data Science-Analytics methodology with a standards-based approach to increase the success of Big Data software system projects in small businesses. Professor with 8 years of seniority in the Department of Electronic Systems at the Autonomous University of Aguascalientes (UAA). With a chapter published in the book Development Methodologies for Big Data Analytics Systems: Plan-Driven, Agile, Hybrid, Lightweight Approaches.

Manuel Mora received an M.Sc. Degree in Artificial Intelligence from Monterrey Tech, in 1989, and an Eng.D. Degree in Engineering from the National Autonomous University of Mexico (UNAM), in 2003. He has been a full-time Professor with the Autonomous University of Aguascalientes (UAA), Aguascalientes, Mexico, since 1994. He has published over 100 research papers in international top conferences, research books, refereed journals listed in JCRs and Scopus indexes, and co-edited five international research books on the topics of DMSS, IT Services and Data Centers, and Research Methods. His research articles have been published in JCR indexed journals, such as IEEE Transactions on Systems, Man, and Cybernetics: Systems, European Journal of Operational Research, International Journal of Information Management, Engineering Management Journal, International Journal of Information Technology and Decision Making, Information Technology for Development, International Journal of Software Engineering and Knowledge Engineering, Computer Standards and Interface, Software Quality Journal, Expert Systems, Software and Systems Modeling, Journal of Organizational Computing and Electronic Commerce, and Journal of Supercomputing. His current research interests include Agile Development Methodologies for: IT Services, Big Data Software Systems, Ontology-based KMS, and SOA/MSA-based Software Systems. He is also an ACM Senior Member and a Mexican National Researcher at Level 2.

Sergio Galvan-Cruz is a Computer Systems Engineer who graduated from the Autonomous University of Aguascalientes. He studied for a Master's Degree in Exact Sciences and Information Systems at the same institution. Finally, he completed his Doctorate at the University Juárez Autonomous de Tabasco through the Inter-institutional Doctorate in Computer Science Programmes. He is a professor at the Autonomous University of Aguascalientes in the Electronic Systems Department. In addition, he has published research works related to the implementation of international standards in agile software development environments, for example: the translation to Spanish of the ISO/IEC 29110 Entry Profile Standard; and the development of the guide to implement ISO/IEC 29110 in Agile Environments. Finally, he has been involved in some private software development projects in web and mobile environments. Currently pursuing a postdoc (2022-2024) at CIMAT Zacatecas in the area of Software Engineering.

Angel Munoz-Zavala obtained his M.Sc. and Ph.D. Degrees in Computer Science, both from the Mathematics Research Center (CIMAT), Guanajuato, Mexico. Since August 2009, he is a Professor-Researcher at the Autonomous University of Aguascalientes. He is mainly interested in AI (Evolutionary Algorithms) and Statistical Science (Reliability Systems). He has published more than 20 papers in international journals and conferences on these topics. In 2009 he was awarded 1st place in the Ph.D. Thesis category by the Mexican Society of AI.

Francisco Alvarez-Rodriguez is a Professor of Software Engineering, Department of Computer Science, Autonomous University of Aguascalientes (U.A.A.). Ph.D. in Teaching Methodology, IMEP (Mexico). D. in Engineering from UNAM (Mexico). He has been Dean of the Centre of Basic Sciences of the U.A.A., as well as Head of the Department of Electronic Systems. Member of academic cores of several postgraduate programmes at the U.A.A. Doctorate in Computer Science, Inter-institutional Doctorate in Science, Master of Science with the option in Mathematics and Computer Science. Author of books and articles in the line of Learning Objects and Software Development Processes. Member of the National System of Researchers (SNI) and International Research Groups. He is currently president of the Consejo Nacional de Acreditación de Programas de Computación e Informática, A.C. (National Council for Accreditation of Computer Science and Informatics Programmes).

Appendix A.1.

Table A.1. Analysis of CRISP-DM, BDPL, TDSP and DDSL vs the theoretical lightweight SDLC Pro-Forma for BDAS of the ISO/IEC 29110-standard-basic profile.

SDLC element	Theoretical lightweight SDLC Pro Forma for BDAS of the ISO/IEC 29110-basic profile	CRISP-DM	BDPL	TDSP	DDSL
Roles (4)	• **User roles:** **R.1:** Customer • **Management roles:** **R.2:** Project manager. • **Technical roles:** **R.3:** Work team: **R.4** Technical leader **R.5:** Programmer **R.6:** Designer **R.7:** Analyst	• **User roles:** **R.1:** Customer. • **Management roles:** **R.2:** Project manager. • **Technical roles:** **R.3:** Developer team. • **External roles:** None	• **User roles:** **R.1:** Project process user • **Management roles:** **R.2:** Manager • **Technical roles:** **R.3:** Operator user **R.4:** Developer maintainer • **External roles:** **R.1:** Acquirer supplier	• **User roles:** **R.0:** Customer • **Management roles:** **R.1:** Group manager. **R.2:** Team lead **R.3:** Project lead • **Technical roles:** **R.4:** Project individual contributors (data scientists, business analysts, data engineers, solution architect, application developers)	• **User roles:** **R.1:** Business stakeholder • **Management roles:** **R.2:** Data product manager • **Technical roles:** **R.3** Data scientist **R.4:** Data infrastructure engineer **R.5:** Data storyteller
Process (phase)-activities (2,10)	• **Process 1. Project management:** • **Activity 1:** 1. Project planning 2. Project plan execution 3. Project assessment and control 4. Project closure	• **Phase 1. Business understanding:** • **Activity 1:** 1. Determine business objectives 2. Assess situation 3. Determine data mining goals 4. Produce project plan	• **Process 1. Organizational project-enabling process:** • **Activities:** 1. Lifecycle mode management 2. Infrastructure management process. 3. Project portfolio management process 4. Domain specialist resource management process 5. Human resource management process 6. Quality management process	• **Phase 0. TDSP Workflow for Project Execution:** • **Activity 0:** 1. Plan Sprint 2. Review code built from several branches 3. Merge-Delete branches	• **Phase 1. Ideation:** • **Activity 1:** 1. Identified Problem 2. Project Scoping: a) Review prior art b) Calculate value c) Assess feasibility
	• **Process 2. Software implementation:** • **Inception phases:** • **Activity 2**: 1. Software implementation initiation.	• **Phase 2. Data understanding:** • **Activity 2:** 1. Collect initial data 2. Describe data 3. Explore data 4. Verify data quality • **Phase 3. Data preparation:** • **Activity 3:** 1. Select data 2. Clean data 3. Construct data 4. Integrate data 5. Format data	• **Phase 1.** a) Project process • **Activity 1-a):** 1. Project planning process • **Phase 1.** b) Data process • **Activity 1-b):** 1. Data collecting process 2. Data inventory process • **Phase 1.** c) Technical process • **Activity 1-c):** 1. Stakeholder requirement definition process	• **Phase 1. Business understanding:** • **Activity 1:** 1. Define objectives	• **Phase.1 Ideation:** • **Activity 1:** 3. Manage Backlog 4. Select Artifacts
	• **Process 2. Software implementation:** • **Elaboration phases:** • **Activity 2:** 2. Software requirements analysis 3. Software architectural and detailed design	• **Phase 4.** a) Conceptual modelling: • **Activity 4:** 1. Select modelling techniques 2. Generate test design	• **Phase 2.** a) Project Process • **Activity 2-a):** 1. Project assessment and control process 2. Decision management process 3. Risk management process 4. Configuration management process 5. Information management process 6. Measurement process • **Phase 2.** b) Data process • **Activity 2-b):** 1. Data requirement analysis process 2. Data integration process 3. Data verification process 4. Data analysis process 5. Data modelling process • **Phase 2.** c) Technical process: • **Activity 2-c):** 1. Requirement analysis process. 2. Architectural design process	• **Phase 1. Business understanding:** • **Activity 1:** 2. Identify data sources • **Phase 2. Data acquisition and understanding:** • **Activity 2:** 1. Ingest the data 2. Explore the data 3. Set up a data pipeline	• **Phase 2. Data acquisition and exploration:** • **Activity 2:** 1. Getting the data 2. Identify sources the data a) Connect 3. Create data (capture) 4. Buy and ingest data 5. Explore data 6. Prepare data

			3. Data automation and monitoring process 4. Data visualization process 5. Data decision support process		
	• **Process 2. Software implementation:** • **Construction-Deployment Phase:** • **Activity 2:** 4. Software construction 5. Software integration and tests 6. Product delivery	• **Phase 4.** b) Computational modelling: • **Activities:** 3. Build model 4. Assess model • **Phase 5. Evaluation:** • **Activity 5:** 1. Evaluate results 2. Review process 3. Determine next steps • **Phase 6. Deployment:** • **Activity 6:** 1. Plan deployment 2. Plan monitoring and maintenance 3. Produce final report **4.** Review project	• **Phase 3.** a) Agreement processes • **Activity 3-a):** 1. Data value, result, and innovation process 2. Acquisition process 3. Supply process. • **Phase 3.** b) Project Process • **Activity 3-b):** 1. Project assessment and control process 2. Decision management process 3. Risk management process 4. Configuration management process 5. Information management process 6. Measurement process • **Phase 3.** c) Data process • **Activity 3-c):** 1. Data simulation process. 2. Data prediction process. 3. Data innovation process 4. Data validation process 5. Data cleaning process 6. Data maintenance process • **Phase 3.** d) Technical process • **Activity 3-d):** 1. Implementation process 2. Integration process 3. Verification process 4. Transition process 5. Validation process. 6. Operation process 7. Maintenance process 8. Disposal process	• **Phase 3. Modelling:** • **Activity 3:** 1. Feature engineering 2. Model training 3. Model evaluation • **Phase 4 Deployment:** • **Activity 4:** 1. Operationalize a model • **Phase 5. Customer acceptance:** • **Activity 5:** 1. System validation 2. Project hand-off.	• **Phase 3. Research and Development:** • **Activity 3:** 1. Generate Hypothesis 2. Validate right tools a) IT request b) Experiment c) Assess result 3. Validate the need new data 4. Insightful? 5. Share insight • **Phase 4. Validation:** • **Activity 4:** 1. Validate the business 2. Validate technically 3. Validate ready to deploy 4. Publish • **Phase 5. Delivery:** • **Activity 5:** 1. Plan Delivery 2. Deploy 3. Test • **Phase 6. Monitoring:** • **Activity 6:** 1. Monitor a) Usage b) Performance c) Value 2. Identify improvements 3. Generate value
Products (work products) (22)	• **Process 1. Project management:** • **Input products:** **P.1:** Statement of work • **SSP.1:** Software configuration **P.2:** Change request • **Internal products**: **P.2:** Change request **P.3:** Correction register **P.4:** Meeting record, **P.5:** Verification results **P.6:** Progress status record **P.7:** Project repository backup. • **Output products** **P.8:** Project plan **P.9:** Acceptance record **P.10:** Project repository **P.4:** Meeting record • **SSP.1:** Software configuration	• **Phase 1. Business understanding:** • **Work Products 1:** 1. Background 2. Business objectives 3. Business success criteria 4. Inventory of resources 5. Requirements, assumptions, and constraints 6. Risks and contingencies 7. Terminology 8. Costs and benefits 9. Data mining goals. 10. Data mining success criteria 11. Project plan 12. Initial assessment of tools and techniques	• **Process 1. Organizational project-enabling process: No reported**	• **Phase 0. TDSP Workflow for project execution:** • **Work Products 0:** 1. Sprint plan • **Work Products 1:** 1. Charter document	• **Phase 1. Ideation:** • **Work Products 0:** 1. Project scope document

	• **Process 2. Software implementation:** • **Inception phases:** **P.8:** Project plan (input) **P.10:** Project repository (input)	• **Phase 2. Data understanding:** • **Work Products 2:** 1. Initial data collection report 2. Data description report 3. Data exploration report 4. Data quality report • **Phase 3. Data preparation:** • **Work Products 3:** 1. Rationale for inclusion/exclusion 2. Data cleaning report 3. Derived attributes 4. Generated records 5. Merged data 6. Reformatted data 7. Dataset 8. Dataset description • **Phase 4.** a) Conceptual modelling: • **Work Products 4:** 1. Modelling technique 2. Modelling assumptions 3. Test design	• **Phase1.** a) Project process: No reported b) Data process: No reported c) Technical process: No reported	• **Phase.1 Business understanding:** • **Work Products 1:** 2. Data source 3. Data dictionaries	• **Phase 1. Ideation:** • **Work Products 0:** 1. Project scope document 2. Project kick-off
	• **Process.2 Software implementation:** • **Elaboration phases:** **P.11.** Validation results (internal) **P.5**: Verification results (internal) • **SSP.1:** Software configuration (output) **P.12**: Requirements specification **P.13**: Software design **P.14**: Traceability record **P.15**: Test cases and test procedures	• **Phase 4.** b) Computational modelling • **Work Products 4:** 4. Parameter settings 5. Models 6. Model descriptions 7. Model assessment 8. Revised parameter settings	• **Phase 2.** a) Project process: No reported b) Data process: No reported c) Technical process: No reported	• **Phase 2. Data acquisition and understanding:** • **Work Products 2:** 1. Data quality report 2. Solution architecture 3. Checkpoint decision	• **Phase 1. Ideation** • **Work Products 1:** Model requirements Doc. • **Phase 2. Data acquisition and exploration:** • **Work Products 2:** Data dictionary
	• **Process 2. Software implementation:** • **Construction-deployment phase:** **P.2:** Change Request (output) • **SSP.1:** Software configuration (output) **P.16:** Software components **P.17:** Software **P.18**: Test Report **P.19:** Product operation guide **P.20:** Software user documentation **P.21:** Maintenance documentation	• **Phase 5. Evaluation:** • **Work Products 5:** 1. Assessment of data mining results 2. Approved models 3. Review of process 4. List of possible actions 5. Decision. • **Phase 6. Deployment:** • **Work Products 6:** 1. Deployment plan 2. Monitoring and maintenance plan 3. Final report 4. Final presentation 5. Experience documentation	• **Phase 3.** a) Agreement processes: No reported. b) Project Process: No reported. c) Data process: No reported. d) Technical process: No reported	• **Phase 3. Modelling:** • **Work Products 3:** 1. Model • **Phase 4. Deployment:** • **Work Products 4:** 1. A status Dashboard that displays the system health and key metrics 2. A final modelling report with deployment details 3. A final solution architecture document. • **Phase 5. Customer acceptance:** • **Work Products 5:** 1. Exit report of the project for the customer	• **Phase 3. Research and development:** • **Work Products 3:** Data model experiment • **Phase 4. Validation:** • **Work Products 4:** Validated Data Model} • **Phase 5. Delivery:** • **Work Products 5:** Production Data Model • **Phase 6. Monitoring:** • **Work Products 6:** Monitoring and training plan