Universiteit Leiden
The Netherlands

Bachelor Computer Science & Economics

Secure software development at BDO today and tomorrow: The current methodology and how it can be enhanced following the relevant standards and regulations.

Wouter ten Brinke

First supervisor, second supervisor and BDO supervisor:
Olga Gadyatskaya
Arina Kudriavtseva
Jeffrey Orie (BDO)

BACHELOR THESIS Leiden Institute of Advanced Computer Science (LIACS)
www.liacs.leidenuniv.nl 23/02/2024

Abstract

In today's digital age, cybersecurity is a major concern for businesses, and this thesis, conducted in collaboration with the University of Leiden and BDO, proposes modifications to enhance BDO's secure software development methodology (SSDM). Through a comprehensive review of literature, detailed analysis of BDO's current practices, and stakeholder interviews, the research identifies gaps and suggests improvements. Key findings include the need for end-of-life policies, a bug bounty program, red team engagements, and the adoption of security user stories. These recommendations aim to ensure BDO's SSDM aligns with industry standards and effectively mitigates security risks.

Contents

1 Introduction **1**
- 1.1 Research Objectives . . . 1
- 1.2 Thesis Overview . . . 2

2 Definitions **3**

3 Method **4**
- 3.1 Outline of Mapping BDO's SSDM . . . 4
  - 3.1.1 Organizational-wide Processes . . . 4
  - 3.1.2 Processes that Cover All Stages of SDLC . . . 5
  - 3.1.3 SDLC Stages for a Specific Project . . . 5
  - 3.1.4 Verification of Results . . . 6
- 3.2 Outline of SSDM Comparison . . . 7
- 3.3 Outline of the gap analysis . . . 8

4 Literature Review **10**
- 4.1 Introduction to Secure Software Development . . . 10
  - 4.1.1 Waterfall method . . . 10
  - 4.1.2 Agile method . . . 10
- 4.2 Secure software development . . . 10
- 4.3 SSDM Comparison table . . . 12

5 Results **14**
- 5.1 Comparison of BDO's SSDM with Industries' Best Agile Practices . . . 14
  - 5.1.1 Mapping BDO's SSDM to the Comparison Table . . . 14
  - 5.1.2 Gap Analysis . . . 17
  - 5.1.3 Identified gaps . . . 19
- 5.2 Comparing BDO's SSDM to all other industry recognized SSDMs . . . 20
- 5.3 Gap Analysis . . . 30
  - 5.3.1 Identified Gaps and Areas for Improvement . . . 30
  - 5.3.2 Summary . . . 31

6 Proposed Improvements **32**
6.1 End-of-Life Policies 32
6.2 Bug Bounty Program 32
6.3 Red Team Engagements 32
6.4 Security User Stories 32

7 Discussion **33**
7.1 Reflection on the Process and Findings 33
7.2 Limitations of the Research 33

8 Conclusions **34**

1 Introduction

In today's digital age, cybersecurity has become a major concern for businesses and organizations of all sizes. [5] As more sensitive information is stored and transmitted online, the threat of cyber attacks and data breaches has risen significantly [14]. To mitigate these risks, it is important for companies to implement SSDMs that comply with industry standards and regulations[18].
This research was done jointly with University of Leiden and BDO. BDO (Binder, Dijker Otte & Co.) is an internationally operating accountancy and advisory firm. BDO develops sensitive software for audit in-house.
The aim of this thesis is to propose modifications to enhance BDO's methodology to ensure adherence with industry standards and regulations. The thesis reviews various literature on SSDMs, including studies on security risks, best practices, and SSDMs.
Furthermore, the thesis analyzes BDO's current methodology in detail by interviewing relevant stakeholders of the methodology in practice. Based on the gaps found and the analysis of BDO's current methodology, the thesis proposes modifications to the methodology to ensure compliance while maintaining its effectiveness.
The main research question this thesis aims to answer is:

- How can BDO enhance its current secure software development methodology to comply with relevant industry standards?

To answer the main research question, multiple sub questions are answered:

- What are the best practices for enhancing SSDMs?
- What are the current SSDMs used by BDO?
- What are the gaps between BDO's current methodology and the required industry standards?

1.1 Research Objectives

The objectives of this research are as follows:

- **Gap Analysis:**
  - Identify specific areas where BDO's SSDMs (SSDM) diverges from industry best practices.
  - Highlight any missing practices in BDO's SSDM that are recommended by other methodologies.
  - Assess the significance of these gaps in terms of potential security risks and compliance issues.
- **Formulating Recommendations:**
  - Develop actionable recommendations to address each identified gap.

- **Validation and Feedback:**
  - Present the identified gaps and recommendations to key stakeholders for validation and feedback.

By systematically interpreting the results and forming tailored advice, this research aims to provide a clear pathway for BDO to enhance its secure software development methodology, ensuring robust security and compliance with industry standards.

1.2 Thesis Overview

The thesis is structured into several chapters, each addressing specific aspects of the research. Below is a brief overview of the structure:

- **Introduction** (1): This chapter introduces the problem, research question, and provides a situational overview. It establishes the context and significance of the research, outlining the objectives and scope of the study.
- **Definitions** (2): This section provides precise definitions of key terms and concepts related to SSDMs, industry standards, and regulations. It ensures a clear understanding of the terminology used throughout the thesis.
- **Method** (3): This chapter describes the research design and methodology employed in the study. It details the multi-step approach, including desk research, interviews, and observations, used to capture and analyze BDO's current secure software development methodology.
- **Literature Review** (4): This chapter delves into existing literature on SSDMs, particularly focusing on secure software development. It reviews various studies, best practices, and methodologies to provide a comprehensive background for the research.
- **Results** (5): The findings of the research are presented in this chapter. It includes a detailed comparison of BDO's methodology with industry best practices, identifies gaps, and assesses the impact of these gaps on BDO's security posture.
- **Proposed Improvements** (6): Based on the identified gaps, this section proposes specific modifications to enhance BDO's secure software development methodology. The recommendations are tailored to ensure compliance with relevant industry standards and regulations.
- **Discussion** (7): This chapter reflects on the research process, findings, and limitations. It discusses the implications of the results, the effectiveness of the methodology used, and areas for future research.
- **Conclusions and Further Research** (8): The final chapter summarizes the key findings and contributions of the thesis. It offers conclusions based on the research and suggests directions for further investigation and improvement in SSDMs.

2 Definitions

Secure Software Development Methodology (SSDM): A structured approach to software development that focuses on ensuring the safety and security of the software and the data it processes throughout its lifecycle. This includes practices for identifying, mitigating, and managing security risks related to software. [17].
Industry Standards: Guidelines and norms established by IT and cybersecurity industry bodies to ensure consistency, uniformity, and safety in software development. Examples include ISO/IEC 27001, NIST SP 800-53, and PCI DSS [20].
Regulations: Legal requirements and regulations that apply to the development of software for certain industries, such as the financial sector. Examples include GDPR (General Data Protection Regulation), HIPAA (Health Insurance Portability and Accountability Act), and SOX (Sarbanes-Oxley Act) [20].
Security Risks: Potential threats to the safety and security of software and the data it processes, such as malware, hacking, and data breaches. Security risks can arise from various sources, including external attackers, internal threats, and software vulnerabilities [17].
Best Practices: Proven and effective methods and techniques recommended for developing secure software. These practices are based on industry standards, expert consensus, and empirical evidence [6].
Threat Modeling: The process of identifying potential security threats and vulnerabilities in software systems and assessing their potential impact. Threat modeling helps in understanding the attack surface and prioritizing security efforts [7].
Secure Coding Practices: Programming techniques and guidelines designed to minimize security vulnerabilities and prevent common software exploits. Examples include input validation, proper error handling, and the use of secure libraries [12].
Security Testing: The evaluation of software systems to identify security weaknesses. This includes various types of testing such as penetration testing, vulnerability scanning, code review, and fuzz testing[7].
Secure Deployment: Procedures and controls for securely deploying and configuring software applications to mitigate risks associated with deployment-related vulnerabilities. This includes securing the deployment environment and ensuring that security settings are correctly applied [20].
Incident Response: A structured approach for handling security incidents, including detection, containment, eradication, and recovery. Effective incident response helps minimize the impact of security breaches and supports timely recovery [7].
Privacy by Design: An approach that incorporates privacy and data protection principles into the design and architecture of software systems from the outset. Privacy by Design aims to ensure that privacy is an integral part of the software development process [14].

3 Method

The study design involved a multi-step approach which included desk research and qualitative research methods. The desk research involved collecting existing data from various sources such as documents. The qualitative research methods involved conducting interviews with relevant stakeholders. The first step was to identify and analyze relevant industry standards that may impact BDO's secure software development methodology. A comprehensive review of relevant standards was planned to ensure all potential impacts were considered. This review consisted of mapping all recognized best practices in the table proposed in 4.3.
Next, the study aimed to capture and examine BDO's current secure software development methodology and its various components thoroughly. This involved reviewing the methodology documentation and interviewing relevant stakeholders. Four interviews were conducted: the first to gather internal documents, the second and third with a solution- and enterprise architect to ask detailed questions about the SSDM, and the fourth to verify the results found. These interviews were recorded for accuracy and thorough documentation. All interviewees were asked for permission to be recorded.
To facilitate a precise comparison between BDO's SSDM and industry-recognized agile SSDMs like 'SDL-Agile' by Microsoft and 'Secure Development Lifecycle 2023' by GE. It was crucial to categorize BDO's practices using the same definitions and classifications employed by the industry. This categorization ensured that the comparison was not only valid but also relevant, providing insights that were actionable and meaningful.
All information gathered was categorized and displayed based on the table proposed in 'Secure Software Development Methodologies: A Multivocal Literature Review' [10]. This table contains all industry-recognized best practices for secure software development.
Based on the requirements of the identified regulations and the analysis of BDO's current methodology, the study proposed modifications to the currently deployed secure software development methodology. The modifications were designed to ensure compliance with the relevant regulations while maintaining the efficacy of the methodology. The proposed modifications were then presented to relevant stakeholders for feedback and validation.

3.1 Outline of Mapping BDO's SSDM

To effectively map BDO's SSDM using the table proposed in 4.3, a series of interviews and reviews were conducted. The first interview was an introductory session to gather documents and insights that might be useful during the later stages of the research. This interview was with an IT Security Specialist. The second and third interviews were with Solution Architects and Enterprise Architects to gather detailed information about the best practices BDO adheres to during software development. The fourth interview was with an IT Security Specialist to verify and potentially add to the results of the first three interviews. Below are some example questions that were asked, corresponding with their category.

3.1.1 Organizational-wide Processes

- **Policies and Strategies**

  - What formal security policies are currently in place at BDO?
  - How are these security strategies communicated and enforced across the organization?

- **Response and Recovery**
  - What is the current incident response protocol at BDO?
  - How does the organization test and evaluate its incident response procedures?

- **Risk Management Framework**
  - How does BDO identify and assess security risks?
  - What tools and methodologies are used for risk management?

- **Supply Chain Security**
  - What measures does BDO take to secure its software supply chain?
  - How does BDO assess and manage security risks from third-party vendors?

- **Security Culture**
  - What initiatives are in place to foster a security-centric culture at BDO?
  - How frequently does BDO conduct security training and awareness programs?

- **Other**
  - Are there any other security practices that BDO implements which do not fit into the above categories?

3.1.2 Processes that Cover All Stages of SDLC

- How are security practices integrated into each stage of the software development lifecycle at BDO?
- What tools or frameworks does BDO use to ensure security is maintained throughout the SDLC?

3.1.3 SDLC Stages for a Specific Project

- **Project Inception**
  - How are security goals established at the beginning of a project?
  - What governance structures are in place to oversee security during the project lifecycle?

- **Analysis and Requirements**
  - How does BDO determine security requirements for software projects?

  - What methodologies are used for threat modeling at this stage?

- **Architectural and Detailed Design**
  - How is security integrated into system architecture and design at BDO?
  - Are secure design principles documented and followed?

- **Implementation**
  - What secure coding practices are mandatory at BDO?
  - How does BDO ensure that developers adhere to secure coding guidelines?

- **Verification and Testing**
  - What types of security testing are conducted before software releases?
  - How does BDO handle the findings from security tests?

- **Release and Maintenance**
  - How does BDO manage security patching and updates post-deployment?
  - What is the process for handling security issues that arise after release?

- **Disposal**
  - What procedures does BDO follow for the secure disposal of software and data?
  - How does BDO ensure that all sensitive data is securely erased?

3.1.4 Verification of Results

- **Verification of Policies and Strategies**
  - Based on the initial interviews, these are the identified security policies at BDO. Can you confirm their accuracy and completeness?
  - Are there any additional policies or strategies that were not mentioned previously?

- **Incident Response and Recovery**
  - Can you verify the details of BDO's incident response protocol as described earlier?
  - Are there any recent changes or updates to the incident response procedures?

- **Risk Management**
  - Based on the initial information, these are the tools and methodologies used for risk management at BDO. Can you confirm their accuracy?
  - Are there any additional tools or new practices that have been adopted recently?

- **Supply Chain Security**

  - Can you confirm the measures in place for securing BDO's software supply chain?
  - Are there any recent initiatives or improvements in managing third-party security risks?

- **Security Culture**
  - How effective do you find the current security training and awareness programs?
  - Are there any additional initiatives to enhance the security culture within BDO?

- **Additional Documentation and Insights**
  - Can you provide any additional documents or resources that might have been overlooked in the initial interviews?
  - Are there any recent case studies, incidents, or examples that illustrate the effectiveness of BDO's SSDM?

- **Feedback on Initial Findings**
  - Here are the preliminary findings and gaps identified from the initial interviews. Can you provide your feedback on these points?
  - Are there any discrepancies or areas that need further investigation or clarification?

- **Future Improvements**
  - Based on your expertise, what are some potential areas for improvement in BDO's SSDM?
  - Are there any upcoming industry trends or regulations that BDO should be preparing for?

3.2 Outline of SSDM Comparison

From the first interview two key documents were received that correlated to BDO's SSDM. They are 'DTS Beleid Security' [3] and 'BDO Referentie Software Architectuur' [4]. 'DTS Beleid Security' outlines BDO's IT security policies. A first version of the document was approved in 2014 and it has been continiously updated, reviewed and authorised. The file is on version 8.0.0 and the last update has been in October 2023. 'BDO Referentie Software Architectuur' is a reference for the software architecture used by BDO. The file contains guidelines and standards BDO produced code has to adhere to. The document was created in 2017 and has since had multiple versions published. The current version of the file is 4.0.0 which was reviewed and authorised in February 2023. Both files adhere to the standards which define a mature document as defined in [19]. After gathering the best practices incorporated by BDO, by analysing both files and by incorporating the feedback and answers received in the interviews, they were mapped in the table proposed in section 4.3 'SSDM Comparison table'. BDO's SSDM was compared firstly to the other agile methodologies 'SDL-Agile' by Microsoft and 'Secure Development Lifecycle 2023' by GE. After this comparison, the BDO SSDM was compared to all other available methodologies to gather potentially meaningful

insights which weren't captured in the two agile methodologies. After the SSDM of BDO was drawn up, the results were presented in the fourth interview to be verified by BDO's cybersecurity expert.
By systematically interpreting the results and continiously incorporating feedback, this research aims to provide a clear pathway for BDO to enhance its secure software development methodology, ensuring robust security and compliance with industry standards.

3.3 Outline of the gap analysis

After BDO's SSDM was mapped and compared to 1. All other agile SSDMs and 2. All other SSDMs a gap analysis had to be conducted to find potential misalignment. The gap analysis for BDO's comparison to the agile SSDMs followed the following structure. For every best practice proposed by the agile SSDMs it was searched within BDO's documentation and interview answers if BDO had it covered and under which category it was potentially stated. If no best practice by BDO was found that covered the best practice it was flagged as a potential gap. In the second comparison where BDO's SSDM was compared to the 28 other SSDMs the practices by the 28 SSDMs where firstly tidied up and duplicates and synonyms where joint together. The gaps were then identified in the same manner as the first gap analysis.

Figure 1: Flowchart of the Methodology for identifying gaps between BDO's SSDM and industries SSDM

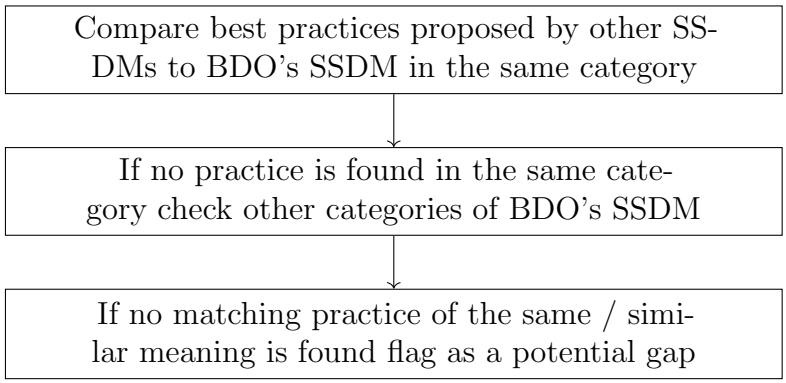
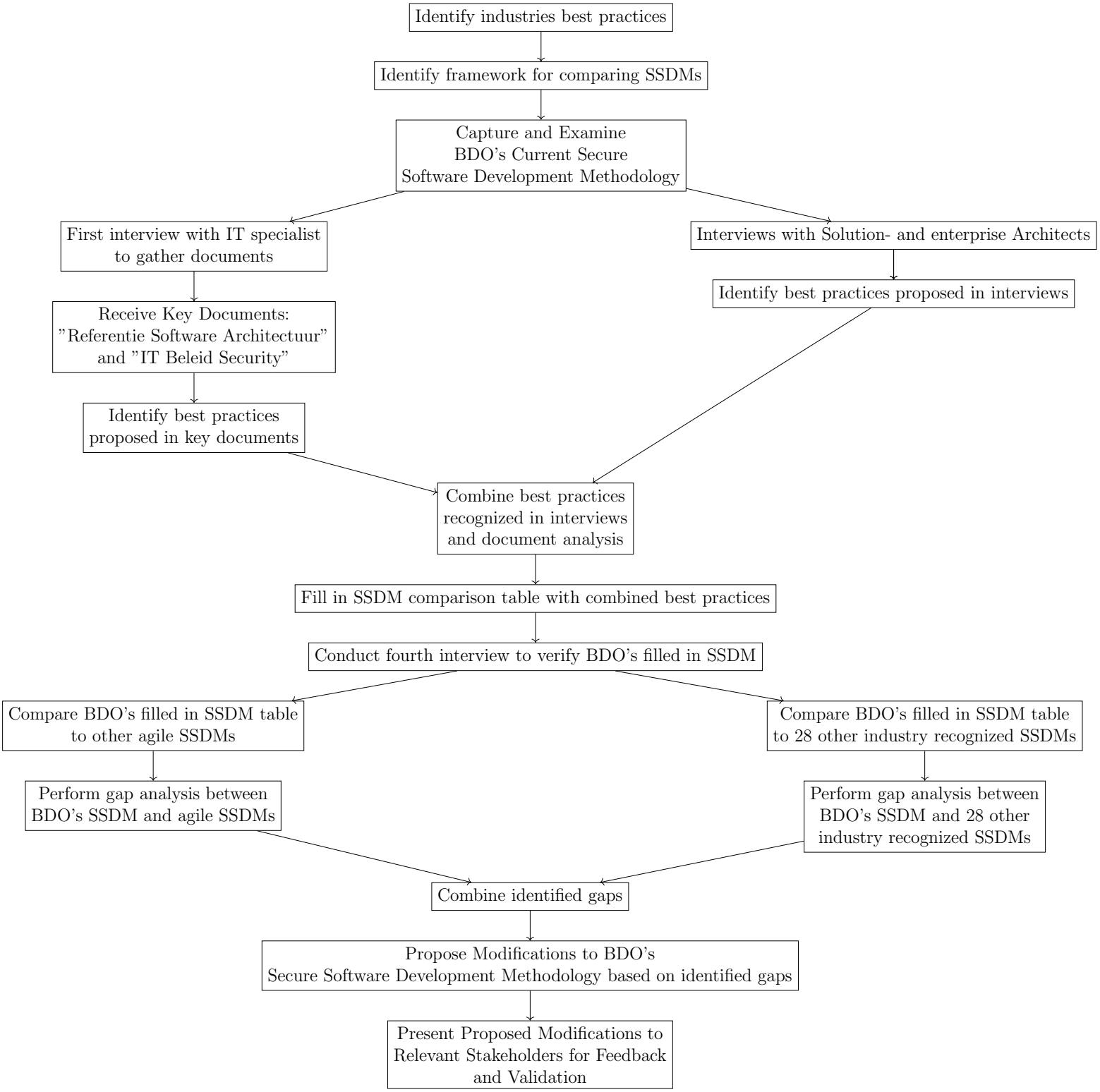


Figure 2: Flowchart of the Methodology for Analyzing and Improving BDO's Secure Software Development Methodology



4 Literature Review

4.1 Introduction to Secure Software Development

In today's digital landscape, businesses and organizations rely heavily on software to store, process and transmit data. SSDMs play a crucial role in safeguarding against cyber threats and protecting valuable information from unauthorized access, modification, or disclosure [1]. SSDMs refer to systematic approaches and practices employed by software development teams to integrate security measures throughout the SDLC. These methodologies include various principles, processes, and techniques aimed at identifying, mitigating, and managing security risks inherent in software applications.

4.1.1 Waterfall method

The Waterfall development method is based on a linear, sequential approach [1]. Each phase of the SDLC must be completed before the next one begins. Typical phases of the waterfall method include: requirements gathering, system design, implementation, testing, deployment and maintenance. An advantage of the waterfall method is its simplicity and straightforwardness [15].

4.1.2 Agile method

In contrast, the Agile methodology prioritizes flexibility, iterative development, and continuous customer collaboration. It fosters a customer-centric approach, early bug detection, and faster time-to-market, making it well-suited for dynamic environments and evolving project requirements. Agile's adaptability to changing circumstances and its focus on delivering value incrementally are significant advantages. However, Agile also presents challenges such as resource intensiveness, complexity in larger projects, and the potential for scope creep.

4.2 Secure software development

After analyzing the different SSDMs, some methodologies have found practices that can't be assigned to the different stages of the SDLC [10]. To still be able to map all the security practices to their corresponding sections the practices are divided into three categories: organization-wide practices, practices that cover all stages of the SDLC and practices specific to a project [10].The software development lifecycle can be divided into multiple stages.
All categories and stages of software development methodologies have their own relevant research areas.
Kanniah and Mahrin [6] emphasize the importance of security training and awareness programs within organizations to ensure all team members are knowledgeable about security risks and mitagtion strategies. These findings helped shape the interview questions that were designed to assess the current level of security awareness among BDO employees and to identify areas for improvement.
Rindell et al. [17] identify key security activities during the requirement, design, and implementation phases of agile development, highlighting the use of automated testing tools. These

activities help ensure that security is integrated into the agile development process. Insights from this study were used to develop questions that investigate how BDO integrates security practices into its agile development process, particularly focusing on the use of automated testing tools.

Khan et al. [7] explore various security risks and practices in secure software development, aiming to enhance product security and efficiency. Their study underscores the importance of systematic risk management throughout the SDLC. Risk management practices were an important point in evaluating BDO's methodology, and interview questions were designed to assess how BDO identifies and mitigates risks.

Throughout the years, many software development methodologies have emerged. Among the first to be published was Microsoft's initiative [10], which was developed in response to the emerging threat of cybersecurity risks. 'Secure Software Development Methodologies: A Multivocal Literature Review' by Arina Kudriavtseva and Olga Gadyatskaya currently recognizes 28 different software development methodologies. These methodologies are mapped in the following table:

The source of methodology	Name	Year of publication
Industry	Microsoft Software Development Life Cycle (SDL) [4], [33]	2006
	McGraw's Secure Software Development Lifecycle Process [57], [61]	2006
	Comprehensive, Lightweight Application Security Process (CLASP) [62]	2006
	Microsoft SDL version 5.2 for Agile Development [63]	2012
	Software Assurance Forum for Excellence in Code (SAFECode) [64]	2018
	Building Secure and Reliable Systems [65]	2020
	BSA framework [37]	2020
	The Secure Software Development Lifecycle at SAP [11]	2020
	ReBIT Application Security Framework [60]	2020
	OWASP Software Assurance Maturity Model [23]	2020
	Cisco Secure Development Lifecycle [13]	2021
	Citrix Security Development Lifecycle [12]	2021
	Building Security in Maturity Model [66]	2021
	GE Secure Development Lifecycle [34]	2022
Government	Grip on Secure Software Development [41]	2015
	NIST 800-160 [67], [68]	2016
	CSA Singapore [36]	2017
	SSDLC cybersecurity Malaysia [42]	2020
	Security in SDLC – Secure Software Development Lifecycle – SSDLC [31]	2021
	NIST 800-218 [35]	2022
Academia	Secure Coding: Building Security into the Software Development Life Cycle [69]	2004
	Secure Software Development Life Cycle Process [32]	2005
	The Secure Software Development Model (SSDM) [43]	2006
	The Integrated Security Development Framework (ISDF) [38]	2010
	Secure Software Development Model: A Guide for Secure Software Life Cycle [44]	2010
	Secure Software Development: a Prescriptive Framework [39]	2011
	Framework for Development of Secure Software [40]	2013
	Methodology for Enhancing Software Security During Development Processes [45]	2018

Figure 3: an overview of 28 recognized software development methodologies [10]

The study researches and documents each best practice the corresponding SSDM proposes. The following chronological timeline shows the evolution of SSDMs

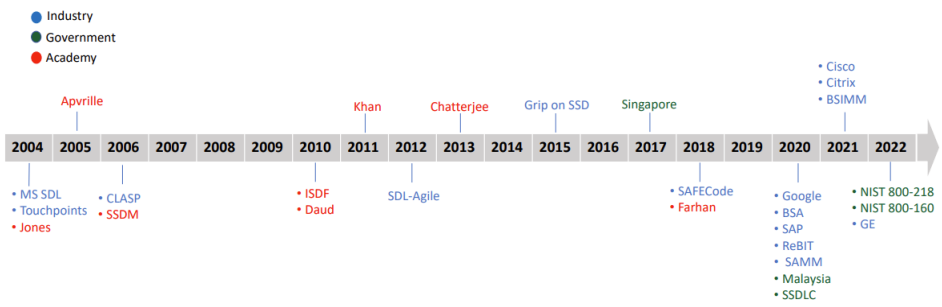


Figure 4: A chronological order of the evolution of software development methodologies [10]

4.3 SSDM Comparison table

To be able to analyze and compare the best practices proposed by these methodologies the comparison table from "Secure Software Development Methodologies: A Multivocal Literature Review" by Arina Kudriavtseva and Olga Gadyatskaya is used. The table is depicted below. [10]

Methodology	Organizational-wide processes						Processes that cover all stages of SDLC	SDLC stages for a specific project					
	Policies and strategies	Response and recovery	Risk management framework	Supply chain security	Security culture	Other		Project inception	Analysis and requirements	Architectural and detailed design	Implementation	Verification and testing	Release and maintenance	Disposal

Figure 5: The 14 different columns of a software development methodology

The 14 different columns of the table can be categorized into three sections:
Organizational-wide processes: Practices that are applicable across the organization and not limited to specific projects or stages of the SDLC.

- Policies and strategies: Encompasses the security policies and strategic approaches embedded within the methodology.
- Response and recovery: Focuses on practices related to incident response and recovery processes within the SSDM.
- Risk management framework: Includes practices that help in identifying, assessing, and managing risks.
- Supply chain security: Security measures related to managing and securing the software supply chain.
- Security culture: Pertains to practices aimed at fostering a security-centric culture within the organization.
- Other: Any additional practices not covered in the other categories but relevant to the SSDM's implementation.

Processes that cover all stages of SDLC: Security practices that are implemented consistently throughout all phases of the software development lifecycle.

SDLC stages for a specific project: This section is further divided into multiple columns, each representing a phase of the SDLC. It details the specific security practices applied during each phase:

- Project inception: Establishes security goals, governance, and initial risk assessments, setting the foundation for security measures throughout the project lifecycle [13].
- Analysis and requirements: Identifies security-specific requirements like data protection, authentication, and authorization, and begins threat modeling to pinpoint vulnerabilities [13].
- Architectural and detailed design: Designs secure system architecture, detailing secure communication protocols, data storage solutions, and specific security controls.
- Implementation: Actual coding and building of the system, incorporating secure coding practices to prevent vulnerabilities like SQL injections and cross-site scripting [8][13].
- Verification and testing: Tests security measures implemented across the system to identify any weaknesses or gaps in security before going live [9].
- Release and maintenance: Deploys the system with ongoing security updates and patch management to address emerging threats and vulnerabilities.
- Disposal: Ensures secure decommissioning of system components to prevent data leakage, following proper data deletion and media sanitization protocols.

5 Results

5.1 Comparison of BDO's SSDM with Industries' Best Agile Practices

The mapping of BDO's SSDM was concluded after the following steps took place.

- An initial interview was conducted to gather relevant documents in regards to BDO's software development process.
- The documents were reviewed and best practices were recognized
- Two interviews were held with Solution- and Enterprise architects to identify best practices
- Based on the documents provided and the answers to the interview questions BDO's best practices were mapped in the table.
- The results of the intial mapping were sent to a cybersecurity expert within BDO for verification
- Based on the feedback received a final mapping was made.
- A gap analysis as defined in 3.3 was then performed on between BDO's SSDM and the agile SSDMs.

5.1.1 Mapping BDO's SSDM to the Comparison Table

Table 1: Comparison of SSDMs SDL-Agile (2012) + GE (2023) Practices and BDO (2023)

Category	**SDL-Agile (2012) + GE (2023) Practices**	**BDO (2023) Practices**
Policies and strategies		• Backup Policies • Software usage policies • Software developments policies • Network security policies • Access control policies

Continued on next page

Table 1 continued from previous page

Category	SDL-Agile (2012) + GE (2023) Practices	BDO (2023) Practices
Response and recovery	• Backup requirements • Backup tests • Restore procedures • Incident response policies • Determine security response standards	• Backup management • Response management
Risk management framework		• Active vendor information monitoring • Vulnerability management • Multiple different frameworks for both static and dynamic analysis
Supply chain security		• Vendor and third-party Management • Third party software policies • Development strategies
Security culture	• Security Education & Developer training	• Regular Security audits and reviews • Employee training and awareness • Developer training and awareness
Other		• Specific policies for the use of cloud services
Processes that cover all stages of SDLC	• Continuous risk assessment	• Privacy by Design Framework • Continuous use of cloud service security • Static and dynamic risk assessment and testing
Project inception	• Identify security and privacy expert • Identify primary security and privacy contacts • Ensure all team members have had security education within the past year	• Establish Development, Testing, Acceptation environment • Ensure team members have right qualifications

Continued on next page

Table 1 continued from previous page

Category	SDL-Agile (2012) + GE (2023) Practices	BDO (2023) Practices
Analysis and requirements	• Review design • Provide security design input • Establish security user stories	• Identifying Security Requirements
Architectural and detailed design	• Update the threat model • Communicate privacy-impact design changes to the team privacy advisor • Design review • Baseline threat model • Threat model the product, its attack surface and its new features • Threat model • Update security user stories	• Secure System Architecture guidelines
Implementation	• Security coding standards • Host security deployment review • Fix all issues identified by code analysis tools for unmanaged code • SAST • DAST • Open source vulnerability assessment	• Secure coding practices • Secure configuration management • static software testing
Verification and testing	• Verification tasks	• Security testing • dynamic and periodic software testing
Release and maintenance	• Final security review • Response planning • Update response services	• Continuous monitoring • Ongoing security updates • Patch management
Disposal		

5.1.2 Gap Analysis

A gap analysis for the Agile methodologies and BDO's methodology will now be conducted.

Policies and strategies In the policies and strategies section, the agile SSDMs do not propose any specific best practices. BDO's policies and strategies section consists of multiple different policies based on the processes of software usage and -development, backup policies, access control policies and network security policies. In the Policies and strategies section, no gaps are found.

Response and recovery BDO's backup policies and backup management are extensive and include the definition of multiple different backup requirements. These include determination of data to be backed up, frequency of backups, backup restore procedures, testing of backups, storage of backup location specification, documenation and retention of backup results. Their security response mangement consists of an extensive monitoring and reporting infrastructure. (Sections 2.4, 2.5 3.1, 3.2, 3.2.1, 3.5.4 and 3.8 of [3]).
In the response and recovery section no clear gaps are identified.

Risk management framework The agile development methodologies do not propose any best practices in the risk management framework section. BDO however does propose best practices with regards to Risk management framework(s). BDO's risk management framework consists of the identification and assessment of risks, monitoring and reporting of risks, handling known vulnerabilities, backup and recovery procedures, security policies and acces control policies, incident reponse and recovery, periodic security audits and improvement of security requirement and regulatory compliance. (Sections 2.4, 2.5 3.1, 3.2, 3.2.1, 3.5.4 and 3.8 [3]).

Supply chain security In the section Supply chain security, BDO's SSDM proposes vendor and third party management policies and strategies to ensure the third party providers adhere to BDO's security standards (section 3.3 of [3]). Development policies are also in place to reduce risk in the usage of third party software in software development within BDO. (section 3.6 of [3]). Network security policies as stated in the "Policies and strategies" section also propose guidelines which control and limit access to the software supply chain to only necessary services (section 3.4.1 of [3]). The agile methodologies do not propose best practices in the supply chain security section. Hence, no gaps are found.

Security culture BDO's security culture consists of both employee and developer security training to mitigate risk ranging from tailgating awareness to annual development risk training. External employee security audits also take place. The agile methodologies propose Security Education & Developer training. This is in-line with BDO's SSDM. No gaps are identified.

Other The policies on the usage of cloud services are defined in the "Other" section. These include information on storage restriction, usage restrictions and information exchange

(section 3.12 of [3]). No best practices are proposed by the agile methodologies. No gaps are found.

Processes that cover all stages of SDLC BDO proposes a privacy by design framework which is to be incorporated throughout the development of software (Section 2 of [4]). The continuous use of cloud services to assess and monitor security risks in program code and to monitor the following of coding standards is also defined (Sections 3.2.1, 5.9 and 5.10 of [4]). Different methods and guidelines on static and dynamic risk assesment and code assessment are also defined (Section 5 of [4]). The agile methodolgies propose continuous risk assesment. BDO's continuous risk assesment is suffice. No gaps are found.

Project inception The project inception phase of BDO's SSDM consists of establishing the Development, Testing and acception environment. It is an environment used to develop, test, train, demonstrate or to accept a product. During project inception, it is also checked whether all project participants have the right qualifications (certificates etc.). The best practices proposed by the agile methodologies are to identify a security and privacy expert, identify primary security and privacy contacts and ensure all team members have had security education in the past years. A gap can be found because BDO's SSDM doesn't propose these best practices. However, because of the size of BDO's development teams, for all developments these are the same people. As referenced in the security culture paragraph, both employees and developers receive security training. Based on these practices no clear gap can be found.

Analysis and requirements BDO's SSDM best practice in the analysis and requirements phase consists of identifying security requirements. Alot of these requirements are based on the OWASP recognized security risks and previous development cycles (section 6 of [4]). The best practices proposed by the agile SSDMs are review design, provide security design input and establish security user stories. BDO does review design and provides security design input however, this is categorized under the development strategies, software development policies and risk management frameworks. The security design input is also provided in BDO's software development policies (Section 5 and 6 of [4]).However no clear usage of user stories to establish security requirements is found in BDO's documentation so this is a potential gap.

Implementation BDO's SSDM proposes secure coding practices and secure configuration management in the implementation category (sections 3.4, 3.5 and 3.7 of [3]). These are in place to ensure code is written uniformly and unambiguously. The configuration management also ensures systems and applications are configured securely and remain so throughout their lifecycle upon implementation. The agile methodologies propose 1. secure coding standards, host security deployment review, 2. fix all issues identified by code analysis tools for unmanaged code, 3. SAST (static application security testing), 4. DAST (dynamic application security testing), 5. open source vulnerability testing. All issues identified by BDO's code analysis tools (like whitesoure and sonarcloud) are fixed in their respective SDLC stages. Static and dynamic analysis is also conducted by these tools as referenced in the section 'processes that cover all stages of SDLC'). Open source vulnerability testing is also conducted as part of

their risk management framework and software usage policies. In this section no gaps are found between BDO's best practices and the agile methodologies proposed best practices.

Verification and testing In the verification stage and testing stage of the SDLC, BDO proposes security testing and static, dynamic and periodic software testing. These practices include a wide range of different sorts of tests. (section 5 of [4]). BDO also uses a 4-eyes principle to verify all work done before it is implemented into the existing code. The static, dynamic and periodic testing is executed through the use of cloud services like Whitesource and Sonarcloud as referenced in the implementation paragraph. The agile methodologies propose verification tasks as a best practice. No clear gap can be found with the verification tasks BDO uses.

Release and maintenance BDO proposes continuous monitoring, ongoing security updates and patch management as part of their software release and maintenance programs. These practices ensure software is maintained inline with BDO's security guidelines. The patch management is a form of response planning as it ensures that during the exposing of a vulnerability the code can be altered quickly (section 3.8 of [3]). A final security review isn't proposed but every stage of the development process is reviewed for security purposes as mentioned in the "Processes that cover all stages of SDLC" section. No clear gaps can be identified.

Disposal Both BDO and Agile SSDMs don't propose best practices for the disposal of software.

5.1.3 Identified gaps

BDO does not clearly use user stories to establish security requirements.
Assessment:
User stories are an essential part of Agile methodologies as they provide a structured and detailed understanding of security requirements from an end-user perspective. The absence of user stories in BDO's SSDM could lead to the following impacts:

- **Incomplete Requirement Gathering:**
  - Important security requirements may be overlooked if user stories are not explicitly defined [16].
- **Misalignment with User Needs:**
  - Without user stories, there is a risk that security measures may not fully address the actual needs and scenarios faced by users [2].
- **Reduced Stakeholder Communication:**
  - User stories facilitate better communication and understanding among stakeholders, developers, and security teams. Their absence might lead to miscommunication and gaps in security implementation [11].

5.2 Comparing BDO's SSDM to all other industry recognized SSDMs

The mapping of BDO's SSDM was concluded after the following steps took place.

- An initial interview was conducted to gather relevant documents in regards to BDO's software development process
- The documents were thoroughly analyzed and best practices were recognized
- Two interviews were held with Solution- and Enterprise architects to identify best practices
- Based on the documents provided and the answers to the interview questions BDO's best practices were mapped in the table
- The results of the intial mapping were sent to a cybersecurity expert within BDO for verification
- Based on the feedback received a final mapping was made.
- A gap analysis as defined in 3.3 was then performed on between BDO's SSDM and the other SSDMs.

Table 2: Comparison of all industry recognized SSDMs and BDO (2023)

Category	**Comparison of all industry recognized SSDMs and BDO (2023)**	**BDO (2023) Practices**
Policies and strategies	• Define a list of approved tools • Define cryptographic standards • Track factors that influence security requirements • Establish guidelines, principles, and rules • Identify global security policy • Establish coding standards and conventions • Identification of coding standards • Define strategy and metrics • Policy and compliance • Set up standard security requirements • Maintain standard security requirements • Develop end-of-life policies • Define general security requirements	• Backup Policies • Software usage policies • Software developments policies • Network security policies • Access control policies

Continued on next page

Table 2 continued from previous page

Category	Comparison of all industry recognized SSDMs and BDO (2023)	BDO (2023) Practices
Response and recovery	• Establish a standard incident response process • Define internal and external policies of vulnerability disclosure • Define roles and responsibilities of vulnerability management • Disaster planning • Recovery planning • Crisis management • Remediation programs • Bug bounty • Red team engagements • Product-wide pentests • External assessment • Define response process for handling security bugs • Define backup procedures • Define business continuity procedures	• Backup management • Response management
Risk management framework	• Active vendor information monitoring • Vulnerability management • Multiple different frameworks for both static and dynamic analysis	• Active vendor information monitoring • Vulnerability management • Multiple different frameworks for both static and dynamic analysis
Supply chain security	• Research and assess security posture of technology solutions • Third-party dependency tracking • CI/CD pipeline security • Verify third-party software complies with security requirements	• Vendor and third-party Management • Third party software policies • Development strategies

Continued on next page

Table 2 continued from previous page

Category	Comparison of all industry recognized SSDMs and BDO (2023)	BDO (2023) Practices
Security culture	• Provide training • Knowledge management and training • Institute security awareness program • Build a culture of security and reliability • Security training • Education & guidance • Knowledge management	• Regular security audits and reviews • Employee training and awareness • Developer training and awareness
Other	• Planning the implementation and deployment of secure development • Understanding roles and responsibilities • Create and maintain software development environment • Personnel is accountable for software security • Standards and requirements • Attack models • Security features and design • Business impact analysis • Maturity guidance • Implement roles and responsibilities • Reuse existing software • Implement supporting toolchains • Implement and maintain secure environments • Lifecycle model management • Infrastructure management • Portfolio management • Human resource management • Quality management	• Specific policies for the use of cloud services

Continued on next page

Table 2 continued from previous page

Category	Comparison of all industry recognized SSDMs and BDO (2023)	BDO (2023) Practices
Processes that cover all stages of SDLC	• Continuous risk assessment • Risk control and risk acceptance • Define and use criteria for software security checks • Protect all forms of code from unauthorized access and tampering • Decision management • Configuration management • Information management • Measurement • Project assessment and control • Quality assurance • System analysis	• Privacy by design framework • Continuous use of cloud service security • Static and dynamic risk assessment and testing
Project inception	• Identify security and privacy expert • Identify primary security and privacy contacts • Ensure all team members have had security education within the past year • Define metrics and compliance reporting • Request for proposal • Security planning • Configure the compilation, interpreter, and build processes to improve executable security • Acquisition • Stakeholder needs and requirements definition • Business or mission analysis • Project planning • Form the security team	• Establish Development, Testing, Acceptation environment • Ensure team members have right qualifications

Continued on next page

Table 2 continued from previous page

Category	Comparison of all industry recognized SSDMs and BDO (2023)	BDO (2023) Practices
Analysis and requirements	• Review design • Provide security design input • Establish security user stories • Define security requirements and -controls • Define abuse and misuse cases • Specify operational environment • Identify user roles and resource capabilities • Document security-relevant requirements • Application security control definition • Understanding adversaries • threat- and risk assessment and analysis • Data protection compliance evaluation • Define the security and privacy requirements • Define security controls • Security planning	• Identifying security requirements

Continued on next page

Table 2 continued from previous page

Category	Comparison of all industry recognized SSDMs and BDO (2023)	BDO (2023) Practices
Architectural and detailed design	• Update the threat model, including product, attack surface, and new features • Communicate privacy-impact design changes to the team privacy advisor • Design review and baseline threat model • Update security user stories • Establish design requirements and apply security principles to design • Perform architectural and design reviews, including external reviews • Develop an encryption strategy and standardize identity and access management • Establish log requirements, audit practices, and configuration for secure settings by default • Review software design for compliance with security requirements and risk information • Design for least privilege, understandability, resilience, recovery, and a changing landscape • Mitigate DoS attacks and perform threat modeling and risk analysis	• Secure system architecture guidelines

Continued on next page

Table 2 continued from previous page

Category	Comparison of all industry recognized SSDMs and BDO (2023)	BDO (2023) Practices
Implementation	• Define and use cryptographic standards • SAST and static code analysis • Use approved tools and integrate security analysis into source management • Implement interface contracts and resource policies • Address reported security issues and perform source-level security reviews • Handle data safely and use safe functions only • Utilize code analysis tools and advanced mitigation strategies • Follow secure coding practices, including checking for known vulnerabilities and unsafe functions • Conduct code reviews and log implementation • Implement measures to prevent counterfeiting and tampering • Ensure proper usage and identifiability of software	• Secure coding practices • Secure configuration management • static software testing

Continued on next page

Table 2 continued from previous page

Category	Comparison of all industry recognized SSDMs and BDO (2023)	BDO (2023) Practices
Verification and testing	• DAST and penetration testing • Identify, implement, and perform security tests • Verify security attributes of resources • Automated and manual testing • Unit, integration, and functional testing • Fuzz testing and debugging • Collecting logs, analysis, and validation of attack surface • Testing software security controls and adversarial techniques • Open-source vulnerability scans and security validation • System security assessment and risk mitigation • Requirements-driven, security, and regression testing • Privacy control validation and automated variant analysis • Vulnerability assessment and feature penetration testing • Application and system security testing • Certification and peer review • Evaluate bugs' criticality and threat-based testing • Automated testing tools and code integrity checks • Security test activities, cases, and vulnerability scanning • Security assessment, audit, and review • Development of security metrics and test reviews	• Security testing • dynamic and periodic software testing

Continued on next page

Table 2 continued from previous page

Category	Comparison of all industry recognized SSDMs and BDO (2023)	BDO (2023) Practices
Release and maintenance	• Code signing and build operational security guide • Manage security issue disclosure and vulnerability reporters • Fix vulnerabilities, manage vulnerability disclosure, notification, and patching • Secure development lifecycle feedback and deploy code using best practices • Recovery and aftermath, vulnerability management, and VAPT • Security authorization, assessment, and secure deployment • Configuration guidance, management, and lifecycle maintenance • Incident, change, environment, and operational management • Security and privacy readiness, operational management, and maintaining privacy controls • Continuous monitoring, logging, and updates • Product security incident response and vulnerability response • Penetration testing and software environment management • Configuration and vulnerability management, risk acceptance, and security review • Software acceptance, verification, and validation • Certification, accreditation, installation, and operation • Maintenance, problem management, and platform security • Verify software release integrity, archive, and protect each release • Identify, confirm, assess, prioritize, and remediate vulnerabilities • Analyze vulnerabilities and identify root causes • Transition, operation, maintenance, supply, and patch management	• Continuous monitoring • Ongoing security updates • Patch management

Continued on next page

Table 2 continued from previous page

Category	Comparison of all industry recognized SSDMs and BDO (2023)	BDO (2023) Practices
Disposal		

5.3 Gap Analysis

BDO's SSDM demonstrates a thorough approach to secure software development, effectively integrating many industry best practices. The methodology includes well-defined policies and strategies, a robust risk management framework, and comprehensive incident response and recovery protocols. The SSDM ensures continuous risk assessment and implements secure coding standards and configuration management throughout the software development lifecycle. Additionally, BDO's commitment to fostering a security-centric culture through regular security audits, training, and awareness programs further solidifies its strong security posture.

BDO's practices are particularly commendable in several areas:

1. **Policies and Strategies**: BDO has established comprehensive policies and strategies, including privacy by design/privacy by default, network security policies, and access control policies, ensuring a solid foundation for secure software development.
2. **Response and Recovery**: The incident response protocols, disaster recovery planning, and crisis management strategies are well-documented and regularly reviewed, highlighting BDO's preparedness in managing and mitigating security incidents.
3. **Risk Management Framework**: BDO's risk assessment and management practices are proactive, utilizing continuous risk assessment and vulnerability management tools to identify and mitigate potential security risks effectively.
4. **Supply Chain Security**: BDO has implemented vendor and third-party management practices to secure the software supply chain, ensuring compliance with BDO's security standards.
5. **Security Culture**: The organization fosters a strong security-centric culture through regular security audits, reviews, and comprehensive training and awareness programs for both employees and developers.

However, there are a few areas where BDO's SSDM could benefit from enhancements to align more closely with the latest industry standards and best practices.

5.3.1 Identified Gaps and Areas for Improvement

1. **End-of-Life Policies**: There is a lack of clearly defined end-of-life policies. Establishing these policies would ensure that the secure decommissioning of software and data, including data sanitization, is systematically managed [20].
2. **Bug Bounty Program**: BDO's SSDM does not include a bug bounty program, which has become a valuable practice for uncovering security vulnerabilities. Implementing a bug bounty program could enhance BDO's proactive security measures by leveraging external security researchers [21].
3. **Red Team Engagements**: The methodology lacks mention of red team engagements, which involve simulated attacks to test the organization's defenses. Incorporating red team

exercises could provide deeper insights into potential security weaknesses and help strengthen BDO's security posture [5].
4. **Security User Stories**: In the analysis and requirements phase, BDO does not explicitly use security user stories. Adopting this practice could improve the identification and communication of security requirements from an end-user perspective, ensuring that security is integrated into the development process from the outset [16].

5.3.2 Summary

While BDO's SSDM is robust and well-aligned with many industry best practices, addressing these identified gaps could further enhance its effectiveness and ensure compliance with evolving industry standards. By focusing on these areas for improvement, BDO can continue to strengthen its secure software development practices and maintain a strong security posture.

6 Proposed Improvements

Based on the identified gaps in BDO's current Secure Software Development Methodology (SSDM), several improvements are proposed to enhance its alignment with industry best practices and ensure a robust security posture.

6.1 End-of-Life Policies

Establishing clearly defined end-of-life policies is essential for the secure decommissioning of software and data. These policies should include procedures for data sanitization, secure disposal of hardware, and formal documentation of the decommissioning process [20].

6.2 Bug Bounty Program

Implementing a bug bounty program can leverage external security researchers to uncover vulnerabilities. This proactive approach can significantly enhance BDO's security measures by identifying and mitigating risks that internal teams might miss [21].

6.3 Red Team Engagements

Incorporating red team engagements, which involve simulated attacks to test the organization's defenses, can provide deeper insights into potential security weaknesses. Regular red team exercises can help strengthen BDO's security posture and readiness against real-world threats [5].

6.4 Security User Stories

Adopting security user stories in the analysis and requirements phase can improve the identification and communication of security requirements from an end-user perspective [16]. This practice ensures that security considerations are integrated into the development process from the outset.

7 Discussion

The process of evaluating and enhancing BDO's Secure Software Development Methodology (SSDM) provided valuable insights into both the strengths and areas for improvement. The structured approach of comparing BDO's practices with industry standards revealed a strong alignment, yet also highlighted specific areas that could benefit from enhancement.

7.1 Reflection on the Process and Findings

The methodology used, including detailed comparisons and stakeholder interviews, proved effective in identifying gaps and areas for improvement. BDO's current practices were found to be robust, particularly in policies and strategies, response and recovery, and risk management frameworks. However, the gaps identified indicate areas where BDO can further fortify its security measures.

7.2 Limitations of the Research

While the research was comprehensive, it had certain limitations. The number of interviews conducted was limited, which may not have captured the full spectrum of internal practices and perspectives. Additionally, the internal nature of the evaluation could introduce biases, as external auditors might uncover different findings. Future research could involve a larger sample size and include independent audits to provide a more comprehensive evaluation.

8 Conclusions

In conclusion, the comprehensive analysis of BDO's SSDM reveals a strong alignment with industry best practices. The structured approach undertaken in this thesis, which involved a detailed comparison between BDO's SSDM and well-established methodologies, highlights the robustness and maturity of BDO's security practices.
Key Findings:

- **Policies and Strategies:** BDO has implemented comprehensive policies and strategies that are in line with industry standards. The adoption of Privacy by Design and Privacy by Default frameworks, along with specific policies on network security and access control, ensures a solid foundation for secure software development.
- **Response and Recovery:** The presence of well-defined incident handling and disaster recovery plans demonstrates BDO's preparedness in managing and mitigating security incidents effectively.
- **Risk Management Framework:** BDO's practices in risk assessment and management, including continuous monitoring and logging, are indicative of a proactive approach to identifying and mitigating potential security risks.
- **Supply Chain Security:** Measures for vendor and third-party management align with industry practices, ensuring the security of the software supply chain.
- **Security Culture:** Regular security audits, reviews, and comprehensive training and awareness programs foster a strong security-centric culture within BDO.
- **Processes that Cover All Stages of SDLC:** Continuous risk assessment and privacy by design frameworks are embedded within each stage of the SDLC, demonstrating BDO's commitment to maintaining security throughout the development lifecycle.
- **Project Inception:** Establishing security goals at the beginning of projects ensures that security is considered from the outset.
- **Analysis and Requirements:** BDO effectively identifies security requirements and utilizes threat modeling methodologies during the analysis phase. Additionally, adopting security user stories in this phase can improve the identification and communication of security requirements from an end-user perspective. This practice ensures that security considerations are integrated into the development process from the beginning.
- **Architectural and Detailed Design:** Secure system design principles are documented and adhered to, ensuring robust architecture and design.
- **Implementation:** Mandatory secure coding practices and secure configuration management are in place to ensure that security is integrated during implementation.

- **Verification and Testing:** Comprehensive security testing and automated testing processes are conducted before software releases. Implementing a bug bounty program to uncover vulnerabilities and incorporating red team engagements can provide deeper insights into potential security weaknesses. Regular red team exercises can help strengthen BDO's security posture and readiness against real-world threats.
- **Release and Maintenance:** Continuous monitoring and updating of response services ensure ongoing security post-deployment.
- **Disposal:** While procedures for secure decommissioning of software and data, including data sanitization, are in place, there is a need for clearly defined end-of-life policies. These policies should include procedures for data sanitization, secure disposal of hardware, and formal documentation of the decommissioning process.

While BDO's SSDM is robust and well-aligned with many industry best practices, addressing the identified gaps could further enhance its effectiveness and ensure compliance with evolving industry standards. Future research could involve a larger sample size and include independent audits to provide a more comprehensive evaluation.

References

[1] Adetokunbo AA Adenowo and Basirat A Adenowo. Software engineering methodologies: a review of the waterfall model and object-oriented approach. *International Journal of Scientific & Engineering Research*, 4(7):427–434, 2013.

[2] A. Alkussayer and W. H. Allen. The isdf framework: towards secure software development. *Journal of Information Processing*, 12(3):157–170, 2010.

[3] BDO. B - dts beleid security, October 2023. Published for BDO in-house use, CONFIDENTIAL.

[4] BDO. Brl - bdo referentie software architectuur, October 2023. Published for BDO in-house use, CONFIDENTIAL.

[5] Yuri Diogenes and Erdal Ozkaya. *Cybersecurity-attack and defense strategies: Infrastructure security with red team and blue team tactics*. Packt Publishing Ltd, 2018.

[6] S. L. Kanniah and M. N. Mahrin. A review on factors influencing implementation of secure software development practices. *Journal of Computer and Systems Engineering*, 9(2):45–60, 2016.

[7] R. A. Khan, S. U. Khan, H. U. Khan, and M. Ilyas. Systematic literature review on security risks and its practices in secure software development. *IEEE Access*, 10:38725–38740, 2022.

[8] Riaan Klopper, Stefan Gruner, and Derrick Kourie. Assessment of a framework to compare software development methodologies. In *SAICSIT '07: Proceedings of the 2007 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on IT Research in Developing Countries*, 10 2007.

[9] Ralf Kneuper. *Software Processes in the Software Product Life Cycle*. Springer, 2018.

[10] Arina Kudriavtseva and Olga Gadyatskaya. Secure software development methodologies: a multivocal literature review. *arXiv preprint arXiv:2211.16987*, 2022.

[11] Garm Lucassen, Fabiano Dalpiaz, Jan Martijn EM van der Werf, and Sjaak Brinkkemper. The use and effectiveness of user stories in practice. In *Requirements Engineering: Foundation for Software Quality: 22nd International Working Conference, REFSQ 2016, Gothenburg, Sweden, March 14-17, 2016, Proceedings 22*, pages 205–222. Springer, 2016.

[12] Predrag Matković and Pere Tumbas. A comparative overview of the evolution of software development models. *International Journal of Industrial Engineering and Management*, 1(4):163–172, 2010.

[13] James W. Moore. *Software Life Cycle Processes*. IEEE, 2006.

[14] Shravan Pargaonkar. Advancements in security testing: A comprehensive review of methodologies and emerging trends in software quality engineering. *International Journal of Science and Research (IJSR)*, 12(9):61–66, 2023.

[15] Shravan Pargaonkar. A comprehensive research analysis of software development life cycle (sdlc) agile & waterfall model advantages, disadvantages, and application suitability in software quality engineering. *International Journal of Scientific and Research Publications (IJSRP)*, 13(08), 2023.

[16] Rashmi Popli, Naresh Chauhan, and Hemant Sharma. Prioritising user stories in agile environment. In *2014 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)*, pages 515–519. IEEE, 2014.

[17] K. Rindell, J. Ruohonen, J. Holvitie, and S. Hyrynsalmi. Security in agile software development: A practitioner survey. *Information and Software Technology*, 128:106433, 2021.

[18] Soobia Saeed, NZ Jhanjhi, Mehmood Naqvi, and Mamoona Humayun. Analysis of software development methodologies. *International Journal of Computing and Digital Systems*, 8(5):446–460, 2019.

[19] Simona Sternad Zabukovšek, Sandra Jordan, and Samo Bobek. Managing document management systems' life cycle in relation to an organization's maturity for digital transformation. *Sustainability*, 15(21):15212, 2023.

[20] Colin C Venters, Rafael Capilla, Stefanie Betz, Birgit Penzenstadler, Tom Crick, Steve Crouch, Elisa Yumi Nakagawa, Christoph Becker, and Carlos Carrillo. Software sustainability: Research and practice from a software architecture viewpoint. *Journal of Systems and Software*, 138:174–188, 2018.

[21] Thomas Walshe and Andrew Simpson. An empirical study of bug bounty programs. In *2020 IEEE 2nd international workshop on intelligent bug fixing (IBF)*, pages 35–44. IEEE, 2020.