REPORT ON:

System Development Lifecycle

IAAS | INTERNAL AUDIT & ADVISORY SERVICES

BRITISH COLUMBIA | Ministry of Finance

Table of Contents

Section **Page No.**

Executive Summary and Overall Conclusion

This audit looked into ministry system development methodologies; specifically, how methodologies help ministries deliver products that meet objectives and stakeholder requirements.

We selected six recent system implementations (initiatives) from the ministries in the sectors of:

- Natural Resources;
- Economy; and
- Justice and Public Safety.

Audit Observations

There were two key observations from our audit. First, ministries used Agile approaches and existing technology to develop new systems. They did so to manage the complexity of program requirements, meet timelines, and reduce the risk of system implementation failure. This aligns with the Government's Chief Information Officer's recent directions.

Secondly, ministries lacked consistent, up-to-date frameworks. This created some process deficiencies in the areas of:

- **Stakeholder Engagement:** The initiatives did not engage early and sufficiently with the Government's central agencies and their ministry experts. We found this most often happened with security, privacy, and financial controls. It contributed to overlooking or de-prioritizing some requirements from government standards and policies.
- **Performance Management:** Most initiatives did not have a process to define and track the realization of benefits that ministry executives could expect from the investments. Ineffective performance management increases the risks of cost overruns and poor decisions.
- **Project records:** The initiatives could not show the adequacy of some key processes as project records such as a business case, feasibility study or executive approvals, were missing or could not be found. Poor record management impairs the accountability and transparency of the Government's decisions.

- **Compliance:** The initiatives did not always follow government security standards for system developments. In addition, some initiatives did not complete a Privacy Impact Assessment, a Security Threat and Risk Assessment, and a Financial Risk and Controls Review before implementation. In other cases, no documentation supported the ministry's decision to forego the assessments.

Audit Recommendations

Following this audit, we have 12 recommendations. Ten recommendations address instances of non-compliance with government standards and process deficiencies. Implementing these recommendations:

- Increases users' and stakeholders' satisfaction;
- Improves information for ministry executives' oversight;
- Strengthens the control environment around development work; and
- Enhances the proactive management of security, privacy, and financial risks of systems in operations.

One recommendation helps ministries develop or mature their system development frameworks. This recommendation improves ministry practices by:

- Creating a comprehensive approach for system development and promoting standardization;
- Increasing internal staff and contractors' awareness of government and ministry-level expectations in areas of security, privacy, financial controls, and record management; and
- Providing ministry executives with the assurance that development teams manage the risks of delays, cost overruns and low user adoption.

One recommendation is for the Office of the Chief Information Officer (OCIO).
We saw similar issues throughout the selected initiatives and ministries.
This means these issues are common. This is why we recommend that the OCIO collaborates with ministries to continue improving government resources. This will help all ministries develop systems that follow government standards and increase success rates.

This report is part of a multi-phased audit of the Government's system development practices. A recent study[1] found that information technology projects generate, on average, about 40% less value than predicted. Public sector projects are more likely to miss their objectives, due to:

- Complex stakeholder landscape;
- Volatile policy objectives; and
- Slower governance processes.

Looking ahead, we will continue to monitor risks across Government and identify the scope and timing of future system development audits.

* * *

Thank you to all government staff, who participated in and contributed to this audit. We appreciate your help and cooperation.

Alex Kortum, CPA, CA
Executive Director
Internal Audit & Advisory Services
Ministry of Finance

[1] *Unlocking the potential of public-sector IT projects*, McKinsey & Company, July 5, 2022 (link)

Introduction

To deliver efficient public services, the Government of British Columbia (Government or Province) must deliver information technology systems (systems) that satisfy a broad range of needs, including users' expectations, program, and compliance requirements.

The key to successfully delivering systems begins with an adequate system development lifecycle methodology.

System development methodologies define the necessary steps, tasks, responsibilities, and internal controls required to execute system implementations economically, effectively, and efficiently. The methodologies can be sequential or iterative workflows.

The key system development stages are:

Planning	Ministries understand program area needs and assess the feasibility of various alternatives, as well as organize a team to plan and govern the development.
Requirement Identification	Ministries research and gather specific details required of the planned solution, defining system, user, control, and quality requirements.
Selection & Configuration	For purchased systems, ministries prepare a request for proposal to invite bids from prospective vendors. Ministries configure the selected systems to their requirements.
Design & Development	Ministries define design and specifications that developers will use to develop the systems.
Implementation	Users test the systems before they become operational. Ministries assess the success of the new or modified systems against pre-determined criteria.
Operations & Sunset	Ministries maintain their systems and keep them up to date until retirement.

Establishing system development practices is important for ministries to set expectations for their development teams. Following defined processes:

- Enables stakeholders to operate in a predictable and repeatable manner;
- Reduces the risk of not meeting program requirements, delays, cost overruns, and low user acceptance.
- Identifies controls to reduce financial loss and fraud risks for financial systems.

Purpose, Scope, and Approach

This is part of a multi-phased audit of the Government's system development practices. Phase I focused on the development and implementation of financial systems. The scope and timing of Phase II will be determined later.

This audit assessed whether the system development methodologies in place help ministries deliver systems that meet their objectives and stakeholders' requirements. The objectives of this audit were to:

- Determine if ministries used adequate system development methodologies to meet the requirements for new or modified financial systems. That includes program objectives, user expectations, and policy alignment.
- Evaluate if ministries delivered their financial systems following system development methodologies and good practices.

We focused on ministry systems that transfer financial data into the Province's Corporate Financial System and other systems that ministries have identified as financial per Chapter 13 of the Core Policy and Procedures Manual (CPPM).

Financial systems process a significant amount of data and information. They are vital in delivering government services and producing the Province's Public Accounts. Financial systems must embed specialized controls to reduce the risk of financial losses and fraud.

This audit covered the following stages:

- Identification of system requirements;
- Selection and configuration;
- Design and development; and
- Implementation stages of system implementations.

The audit did not cover the operations and sunset stages and other tasks not directly related to system development, such as project management, funding approvals, and vendor management. We covered some of these stages and tasks in previous audits.[2]

[2] See IAAS' reports on IM/IT Capital Investment Framework, Legacy Technology, and on IM/IT Procurement Phase I on our website (link)

Internal Audit and Advisory Services (IAAS) completed a risk assessment. This assessment selected ministries to include in this audit from the sectors of:

- Natural Resources;
- Economy; and
- Justice and Public Safety.

In consultation with these sectors, we sampled six recent system implementations (initiatives) to include in the audit for phase I (Appendix B).

Our approach involved:

- Reviewing ministry policies, standards, and procedures;
- Conducting interviews with key management and staff;
- Reviewing the practices and documentation of the initiatives;
- Assessing the selected sectors' current practices against industry good practices;[3] and
- Engaging with stakeholders across the Government. This includes the Office of the Chief Information Officer (OCIO) and the Office of the Comptroller General (OCG).

The audit was conducted by IAAS, Ministry of Finance, and fieldwork was completed in November 2022. We met with each ministry to discuss the specific findings of our work. This report consolidates those findings.

Ministries are required to develop and submit an action plan in response to the recommendations provided, including the timeframe for implementation. IAAS conducts an annual follow-up process to assess ministries' progress to address their action plans in response to the recommendations given.

While this audit focused on selected ministries, the recommendations are relevant across Government. We encourage other government organizations to review their processes and consider these recommendations.

[3] The primary good practices used for this audit were the Information Systems Audit and Control Association's Control Objectives for Information Technologies 2019 framework.

1.0 System Development Approaches

Implementation or enhancement of systems is often complex and costly. It can also significantly impact ministry operations and users' experiences, positively or negatively.

A recent study found that system implementations generate about 40% less value than predicted on average. Public sector system implementations are also more likely to miss their objectives than the private sector. This is due in part to:

- Complex stakeholder landscape;
- Volatile policy objectives;
- Slower governance processes and less effective risk management; and
- Less pressure to deliver **minimum viable products**.[4]

A **Minimum Viable Product** is the smallest functional unit that clients can use to achieve a goal.

As stewards of public funds, taxpayers expect ministries to implement new systems and enhance existing ones efficiently. Ministries should follow established system development methodologies (refer to section 6.0). These methodologies have several benefits for ministries, including:

- Helping ministries produce high-quality systems by defining the processes, responsibilities, and internal controls;
- Mitigating a broad set of risks, such as failing to meet program expectations, experiencing delays, cost overruns, and low user acceptance; and
- Ensuring and demonstrating that the Government uses its resources effectively and efficiently.

1.1 Waterfall and Agile Approaches

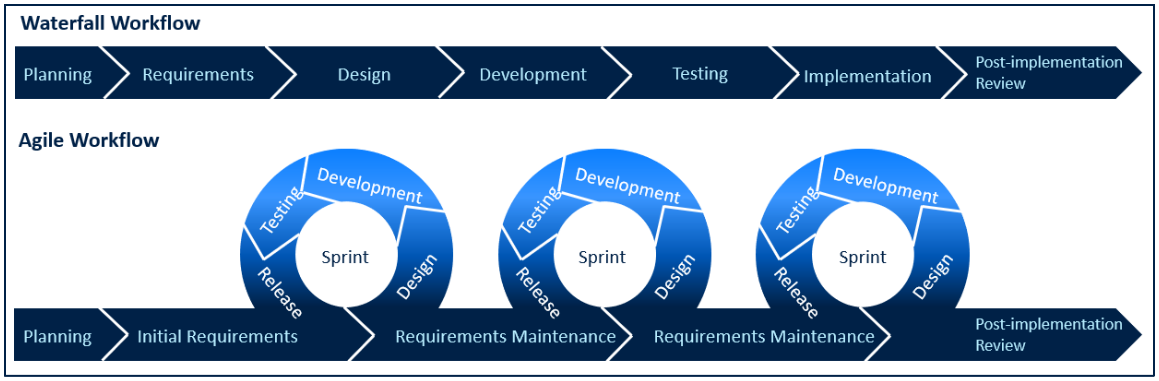
In the industry, two general approaches for system development stand out (Figure 1): the Waterfall and Agile approaches.

While the approaches differ, they share several principles:

- Adequate governance structures and processes;
- Effective involvement of users and stakeholders;
- Well-defined requirements and acceptance procedures; and
- Clear communication between development teams, stakeholders, and users.

[4] *Unlocking the potential of public-sector IT projects*, McKinsey & Company, July 5, 2022 (link)

Figure 1 – Waterfall and Agile Workflows



Source: IAAS

Waterfall

In this approach, a development team completes each phase before starting the next one. To proceed, the team needs the approval of the deliverables of the current phase and the next one's plan. Development teams often release the system to users at the end of the initiative. This can take several months or years.

The Waterfall approach is more suitable for small system implementations with:

- Low chance of deviation due to system requirements that are stable and well-identified upfront;
- Defined start and end dates; and
- Program constraints limiting the number of releases.

Two of the sampled initiatives adopted Waterfall.

Agile

Agile is an iterative approach. It emphasizes early and continuous planning, feedback, and delivery of system functionalities. It involves multiple releases. The Agile approach breaks initiatives into small, dynamic phases, commonly known as sprints. They last between two and four weeks each. They include their own set of phases to build and test new pieces and enhance existing ones.

The Agile approach is more suitable for system implementations with a vision and features that:

- Cannot be well understood and defined upfront; or
- Are likely going to evolve and continuously improve over time.

Agile is a good fit for systems with ongoing client involvement, such as public-facing systems. Another benefit is the frequent release of functionalities. If a system implementation stops prematurely, the investment already made is not lost.

Four sampled initiatives used approaches leveraging Agile principles. Ministries selected these approaches to:

- Manage the complexity of program requirements,
- Meet tight timelines; and
- Reduce the risk of implementation failure.

1.2 Agile in Government

Waterfall was the primary approach in Government until the late 2010s. It fit well with the budgeting process, project management and risk management culture. After challenges on a few major system implementations, the choice approach changed. The OCIO began promoting Agile principles. Key milestones were:

- **BCDevExchange**: In 2017, the OCIO formed this ecosystem of inter-ministry digital specialists to build digital services and work as an incubator. It encourages Agile system development, operations and procurement methods that are more compatible with the industry's expectations.
- **Digital Framework**: In 2019, the Government launched the Digital Framework that establishes Digital Principles. They promote Agile principles, including user involvement, the sharing of codes and **common components** across ministries. It also promotes the improvement of systems over their lifetime.
- **CPPM – Chapter 12**: It outlines principles for information management and information technology management for the Government. With the inclusion of the Digital Principles in 2022, the Government removed the expectation to define requirements for system implementations upfront. Instead, ministries should build systems iteratively based on user feedback.

Common Components are reusable digital services solving common problems across Government. Examples include identity management, payment processing and notification tools.

2.0 Planning the Initiative

Ministries develop new systems or enhance existing ones to meet program and technical objectives. System implementations may significantly impact ministry operations and budgets for years.

Ministries should invest adequate resources in planning for system implementation. A clear strategy and effective governance are two levers that drive the performance of system implementations.

Through the Planning stage, ministries should:

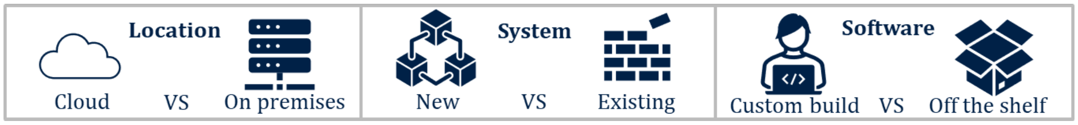
- Ensure the objectives of system implementations are clear and supported by actual needs;
- Select optimal systems in terms of program needs, costs, risks, and sustainability;
- Establish measures and processes to monitor the success of system implementations; and
- Assemble development teams and establish accountability for the resources invested and the success of system implementations.

Without appropriate planning, subsequent stages of a system implementation may face issues, potentially requiring changes or rework. Planning tasks should not only be completed at the start of a system implementation but also be updated as information becomes available.

2.1 Feasibility Study

Defining a technology strategy starts by identifying what technology alternatives are feasible for ministries (Figure 2).

Figure 2 – Technology Alternatives



Source: IAAS

Some alternatives may already exist within the Government; others may need to be procured. The latter adds complexity to system implementations. For instance, a ministry may seek a payment system that another ministry has already established. Both ministries can realize economies by scaling that system to their needs.

CPPM requires that ministries connect with central agencies early in their system implementations. This is to prevent the duplication of system functionalities across Government. For instance:

- **Chapter 12:** Ministries must engage with the OCIO to determine whether system components already exist and partner with other ministries to develop and use existing systems and common components.
- **Chapter 13:** Ministries must collaborate with the OCG before requesting funding to use existing financial system functionalities across Government.

We assessed whether the sampled initiatives assessed several technology alternatives and received appropriate approvals before proceeding.

Technology Selection

We found that most initiatives considered various technology options and selected a technology already available in the Government. The remaining initiatives decided to design and develop their own systems.

Records

We also found that development teams could not show an adequate assessment of the technology alternatives or appropriate approvals for half of the initiatives. Adequate practices would have included an assessment of:

- The fit of alternatives with high-level program and technical needs; and
- Costs, risks, human resources, and organizational impacts.

Teams often shared documentation through emails that they lost with staff turnover. Lost documentation included business cases and program area executives' approvals. This creates a higher risk of not adopting optimal systems. It also impairs the accountability and transparency of government decisions. We recommend that ministries establish a list of key records and approvals that development teams maintain and monitor.

2.2 Benefit Realization

Given the costs and disruptions that system implementations can cause, ministries should have a clear idea of the benefits they expect from their investments before proceeding with the development work. Benefits may be financial, operational, and technical. Organizations managing their benefits have smaller cost overruns.

Good practices recommend developing a benefit realization plan to assess the benefits of system implementations. Such plans define and monitor the realization of benefits as a system is released to users, and after completion.

We assessed whether the sampled initiatives established and maintained benefits. We also determined whether development teams regularly monitored the realization of benefits.

We found that the initiatives that requested capital funding from the Deputy Ministers' Committee on Digital and Data had defined benefits within their business cases. However, they had the following issues:

- They were not easily measurable;
- Their alignment with the original business was unclear; and
- Development teams did not monitor benefits throughout the initiatives.

The other initiatives did not record or monitor benefits. Without benefit realization plans, development teams cannot report on the success of their system implementations. It prevents their **sponsors** from making well-informed decisions.

A **Sponsor** is an individual or group responsible for financing a project and approving key decisions. They set the project towards successful completion and realization of benefits. Senior ministry executive(s) often take this role.

We recommend that ministries establish a process to manage and monitor benefit realization. This includes:

- Updating business cases if initial benefits have lost their relevance; and
- Concluding a post-implementation review to assess the realization of benefits at the end of system implementations (refer to section 5.3).

Government guidance

Since 2021, the OCIO has been developing requirements to help ministries create their benefit realization plan. In addition to expected benefits, the OCIO requires ministries to define success metrics to measure benefit realization and report on it at the end of their system implementations. It will require more frequent reports soon. These requirements are mandatory for system implementations requesting capital funding to the Deputy Ministers' Committee on Digital and Data and good practices for other system implementations.

2.3 Governance

Effective governance is another driver of system implementation performance. Governance is a set of practices enabling oversight of a system implementation by establishing the decision-making framework. The framework defines roles, responsibilities, and accountabilities for team members.

We assessed whether ministries identified an executive sponsor with adequate authority to lead the sampled initiatives and assembled a team of professionals to execute the development and implementation.

Executive sponsors' commitment is essential to go through the challenges of any system implementation, such as funding pressures and organizational resistance. We found that the initiatives had a sponsor at the appropriate level of authority, and who actively engaged in the initiatives.

Ministry teams assigned to a system implementation should also hold adequate skillsets and experience to maximize performance. We found that ministries were diligent in assembling their teams, especially assigning a Project Manager or **Product Owner**.

A **Product Owner** holds the vision for the future system and prioritize tasks. They are empowered to make decisions. A leader from the main program area often takes this role.

Governance Models

Governance practices depend on the selected system development approach. For instance, the OCIO's *Governance for Agile Teams* draft document notes that **standing governance committees** are not always appropriate. They are suitable for predictable initiatives where the circumstances are clear and simple, such as system implementations suitable for Waterfall. For complex problems where Agile is more suitable, standing committees may be slow to react and have the least information to solve the problem.

A **standing governance committee** is a formal group of stakeholders, chaired by the initiative's sponsor, who meet regularly to monitor an initiative's progress, make strategic decisions, and solve issues.

The OCIO is developing a governance model, the Alliance Team, for Agile system implementations to replace traditional governance committees. In this model, an Alliance Lead works alongside the Product Owner to engage with:

- **Sponsors**, which should generally be the Assistant Deputy Minister in charge of the program area, Ministry Chief Information Officer, and Chief Financial Officer; and
- **Stakeholders**, which may be within their ministries and across Government to ensure major decisions are well supported. Stakeholders may include the OCIO and OCG.

Recommendations:

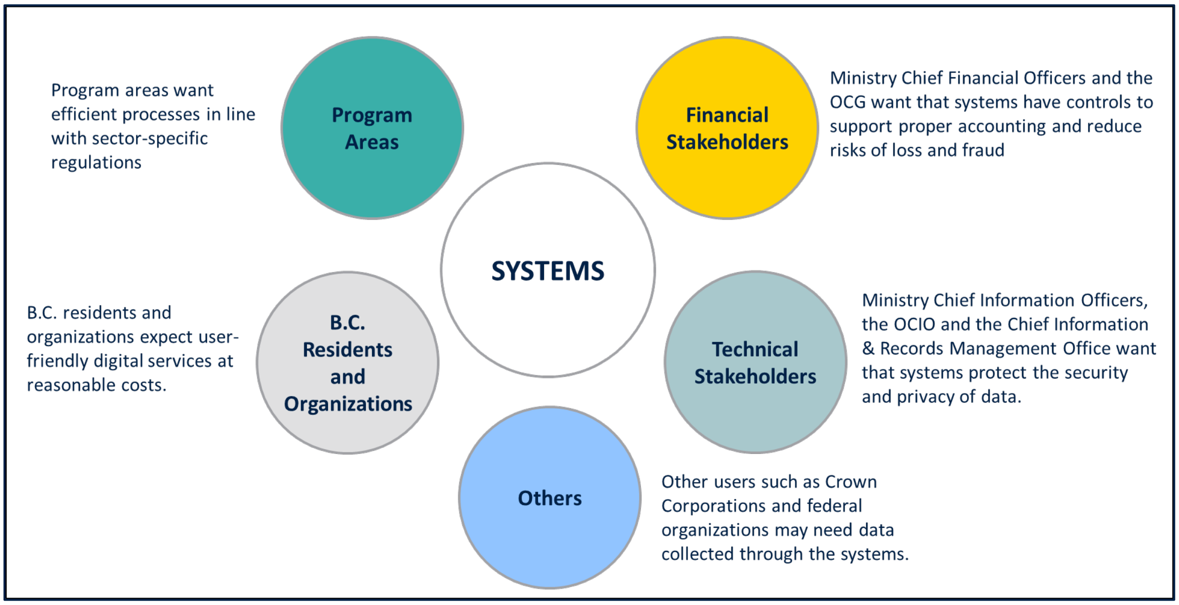
(1) Ministries should establish a list of key records and approvals to maintain and monitor throughout system implementations.

(2) Ministries should require that their development teams establish benefit realization plans and monitor them, including updating their business cases (or equivalent records) when needed and conducting post-implementation reviews.

3.0 Identifying Requirements

System requirements define the operational and technical features of the intended system. Users' and stakeholders' expectations for a future system may differ and sometimes compete, especially in a complex government context (Figure 3).

Figure 3 – System Implementation Stakeholders


Program areas want efficient processes in line with sector-specific regulations
Program Areas
Financial Stakeholders
Ministry Chief Financial Officers and the OCG want that systems have controls to support proper accounting and reduce risks of loss and fraud
SYSTEMS
B.C. residents and organizations expect user-friendly digital services at reasonable costs.
B.C. Residents and Organizations
Technical Stakeholders
Ministry Chief Information Officers, the OCIO and the Chief Information & Records Management Office want that systems protect the security and privacy of data.
Others
Other users such as Crown Corporations and federal organizations may need data collected through the systems.

Source: IAAS

Involving users and stakeholders throughout the system implementation stages is vital to ensure the intended system will meet their needs and expectations. Through the requirements stage development teams should:

- Identify the users and stakeholders of the future systems. They can be either internal or external to the ministry owning the system;
- Consult users and stakeholders to record their needs and expectations, including operational, technical, and compliance requirements; and
- Manage the set of requirements throughout the time of the system implementation.

Incomplete or unclear requirements increase the likelihood of:

- Non-compliance with government financial policies, and privacy and security standards;
- Delays and additional costs; and
- Low user acceptance.

3.1 Stakeholder Consultation

Engaging parties of a future system beyond the main program area is important as systems commonly serve a diverse stakeholder group. While the program area provides the core requirements, development teams must also consider financial, security and privacy requirements.

Development teams must identify stakeholders upfront and record their needs to build a substantial set of requirements. They must regularly confirm their stakeholder landscape as their understanding of the scope of the system implementation may improve with time.

We assessed whether the sampled initiatives' teams took action to understand their stakeholders upfront and engaged them in the requirement-gathering process.

We found that development teams engaged their main program areas' executives and staff in gathering requirements, providing a solid foundation for their system implementation. Some teams also reached out to stakeholders outside Government.

We also found that development teams, who did not have to submit a business case to the OCIO, did not complete and record their stakeholder identification and engagement process. This led to omitting some stakeholders. For instance, they did not always engage early and throughout the initiatives with:

- Central agencies such as the OCIO and OCG;
- Ministries' finance executives; and
- Ministry Security and privacy officers.

Development teams often engaged these experts and authorities once the development had substantially advanced. This resulted in delays and the inability to comply with policies and standards on time (refer to section 5.2).

We recommend that development teams conduct and maintain a stakeholder analysis and ensure adequate engagement early and throughout their system implementations, in accordance with the approaches selected.

3.2 Managing Requirements

Once they have identified stakeholders, development teams can record the system requirements. The development of robust requirements is critical to achieving the intended outputs.

We assessed for the sampled initiatives whether their development teams captured requirements aligning with the business case, and with acceptance criteria.

We found that development teams were diligent in identifying and recording the program areas' system requirements. These requirements addressed the objectives of the business case. Most complex initiatives used discovery sessions with users to identify their program requirements, sometimes facilitated by user researchers.

Maintaining Requirements

Requirements may evolve in terms of priority, complexity, and feasibility. Various reasons explain changes, such as:

- Stakeholders uncover requirements after the initial set of requirements. In our sample, a development team underestimated the complexity of integrating the Government's authentication process into its future system.
- Opportunities to develop additional features may arise. For instance, a development team saw an opportunity to integrate its system with the Province's Corporate Financial System after starting the development work. This removed manual data entries.
- Development teams may review the prioritization of requirements as they manage the developers' workload and timeframe.

For these reasons, development teams should revise their sets of requirements as system implementations progress. In Agile, the Product Owner conducts this process between each sprint. In Waterfall, it requires greater formality as any changes may require adjusting the design and development already completed (Figure 1).

We assessed whether development teams had processes to maintain their requirements, including prioritizing them. We found that the Agile initiatives actively maintained and reviewed their requirements over time. Product Owners continuously refined the requirements and prioritized them to meet the sprint objectives, which were agreed upon with stakeholders. They did so to complete the minimum viable product as early as possible and fix existing defects.

For the two Waterfall initiatives, development teams did not prioritize their requirements. For the initiative that had additional requirements, the development team submitted a new service request to their vendors. Service requests included a technical and budgetary assessment.

Recommendations:

(3) Ministries should require that development teams conduct and maintain a stakeholder analysis and ensure adequate engagement through their system implementations, in accordance with the approach selected.

4.0 Developing Future Systems

Once development teams have recorded and prioritized system requirements, they convert them into working pieces of the future system. While doing so, development teams must comply with government standards. They should also follow industry good practices to ensure the quality of the design and development. It is important because development teams:

- Will transfer the responsibility for the systems to operations teams once completed. Following standard practices will help the latter teams maintain future systems.
- Must ensure the security and effectiveness of future systems. To do so, they must control and test all functionalities they develop.
- Need to ensure that future systems can pass a series of security, privacy and financial assessments required by policies and standards. This is to ensure the systems do not insert unacceptable risks (refer to section 5.2).

Without adequate design and development procedures, the security, reliability, and sustainability of future systems are at risk. It may lead to terminating failing system implementations, causing reputational and financial damages.

4.1 Design & Development

Design and development activities use significant effort and resources. Agile and Waterfall manage these activities differently. Waterfall focuses on defining and approving comprehensive system design plans before implementing them. Conversely, Agile focuses on breaking down these activities into small portions of the system. This allows development teams to design, develop, and fix their work quickly.

We assessed whether development teams designed and developed their systems in line with government standards and industry good practices.

Architects

We found that for half of the sampled initiatives, the development teams did not involve their ministries' system architecture experts in the major design decisions. Other teams involved these experts through review committees, and/or post-implementation technical reviews.

Engaging with system architects early and through implementation helps build systems that are efficient, resilient and that comply with standards and good practices. As mentioned in section 3.1, development teams should engage with their ministry system architects at the key stages of their system implementations.

Records

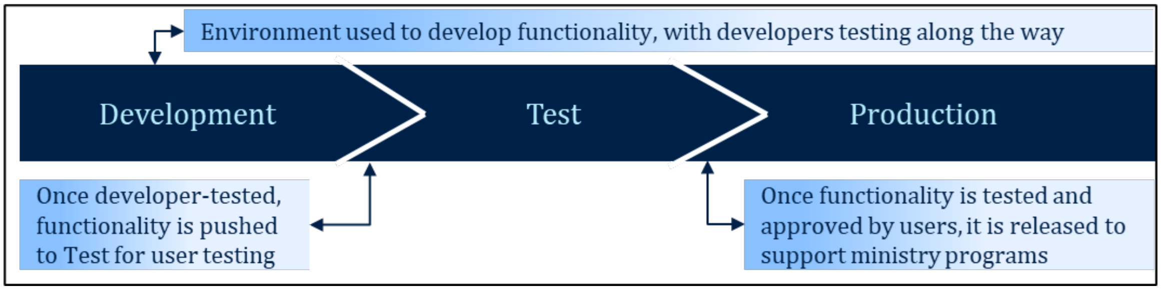
We also found that development teams, especially the Agile ones, did not maintain a sufficient level of system documentation, including architecture decisions. Maintaining good system architecture records ensures knowledge continuity and facilitates future maintenance and enhancements.

4.2 Development Controls

Development teams implement controls into their development process to build systems that can support government services. By doing so, developers can follow a repeatable process through which their work is reviewed and tested before being released to users. Key development controls include:

- Moving portions of code between successive and segregated environments of the systems. This ensures that they only release thoroughly tested and approved code to users (Figure 4).
- Quality assurance activities executed by the development team such as peers reviewing code to uncover defects and technical testing (e.g., testing of interfaces, integrations, security).

Figure 4 – Migration of system functionalities through successive environments


Environment used to develop functionality, with developers testing along the way
Development
Test
Production
Once developer-tested, functionality is pushed to Test for user testing
Once functionality is tested and approved by users, it is released to support ministry programs

Source: IAAS

We assessed whether development teams built controls into their development processes, and the effectiveness of those controls. We found that they conducted quality assurance activities during their system development. We also identified some control deficiencies.

Version Control Systems

Development teams used version control tools to control the movement of the code between environments. These tools record and track code sections, their reviews, and approvals. Development teams also conducted code peer review and technical tests automated in the version control tools. This is good practice.

For two system implementations, the teams did not integrate the approval workflow in the version control tools, which track code sections, with the one in the work management tools, which track system requirements. It created a risk of inconsistency between the tools. It also increases the likelihood of features being released into production without all expected approvals. We found instances where a development team released code sections in production without the expected status and approvals. Therefore, ministries should implement controls ensuring that deployments into production receive and record reviews and approvals.

Segregation of duties

We also found instances where development teams did not establish adequate segregation of duties. Some developers and business analysts held administrator privileges allowing them to modify the production environment, bypassing the version control tool.

As good practice, developers and business analysts should not access the production environment directly, so that all changes to a system can be controlled by a second individual. This control prevents:

- The release of untested and unapproved functionalities in production; and
- Unexpected impacts on ministry operations.

We recommend that ministries review their system privileges to confirm appropriate segregation of duty is in place.

Vendor monitoring

We found one instance where a ministry had limited oversight of its vendor maintaining and enhancing its system. The ministry identified several instances of non-compliance with the OCIO's security standards (i.e., segregation of duties, and protection of sensitive information). A quality plan was also still in draft form. We recommend that the vendor maintains a quality management plan and provides a periodic third-party review report to the ministry.

Recommendations:

(4) Ministries should implement controls that ensure that deployments into production environments receive and record appropriate reviews and approvals.

(5) Ministries should ensure that system administrator access enforces segregation of duty.

(6) Ministries should require their vendors to maintain a quality management plan and provide third-party review reports periodically, when appropriate.

5.0 System Implementation and Evaluation

At implementation, development teams release their systems to users into production (Figure 4). Issues during this phase may:

- Impair ministry operations;
- Impact the rest of the ecosystem;
- Trigger data loss; and
- Open security vulnerabilities.

Development teams following Waterfall should plan the implementation of the whole system carefully. They should anticipate a system **rollback** in case of significant issues. To reduce the risk, Agile teams plan this stage as a routine activity, releasing small portions of a future system to users at the end of each sprint.

Rollback is the process of reversing any changes made during a release into an environment.

Development teams should also engage with users and stakeholders to verify that future systems meet their requirements and are free of major issues before implementation.

5.1 User Acceptance

Development teams must receive confirmation from the users that a new system conforms to their requirements before implementation in production. User acceptance testing enables:

- Development teams to see the system they have been developing working in close to real-life conditions, identifying and fixing defects; and
- Users to familiarize themselves with the future system, easing their adoption.

Development teams should record the testing results and identify defects. As users discover defects, they should agree with development teams on those needing remediation before implementation and those that can wait. The new system should only be released in production after receiving users' endorsements.

We assessed whether development teams establish an effective process to involve users in the acceptance testing, including whether:

- Tests were conducted in close to real-life conditions;
- Testing results were recorded and defects were prioritized for remediation; and
- Users approved the remediation plans.

We found some good practices and control deficiencies. For all sampled initiatives, development teams requested users to test the systems. This is a good practice. Most initiatives had appropriate test cases, close to real-life conditions, and tracked their testing results.

Records

As already mentioned in sections 2.1 and 4.2, we found that one development team did not keep adequate records of testing results and acceptance in the requirement tracking tool. We also found instances where the development team released system sections in operations without the program area's acceptance.

Acceptance Testing in Agile

Some Agile methodologies do not include acceptance testing. Instead, development teams demonstrate the new features to the users and let them try in production to generate feedback. Development teams then consider the feedback and apply it during the next sprint(s). This approach requires seamless communication between development teams and users and transparent task prioritization.

One sampled system implementation followed a similar approach and faced the following challenges:

- As it did not require structured testing but let users try the future features in the test environment on their own time, user engagement was not consistent throughout the initiative.
- Users did not always hear back from the development team about their feedback. They were also unsure whether the team had addressed the defects they had reported.
- Other communication channels were not always effective as discussions tended to be too technical for users to provide valuable inputs.

This created a gap between users' expectations and what the development team was working on. It could have threatened the adoption of the system. Therefore, we recommend that Agile development teams implement a transparent process for users to record and track their feedback. They may invite users to share their feedback and confirm acceptance of new features in the work management tool, and/or receive notifications about the progress. The benefits will be:

- Keeping users aware of what development teams are working on;
- Sharing responsibility for success between users and development teams; and
- Easing user adoption of future systems.

Testing Conditions

Development teams must organize the acceptance testing in a system environment that is separate but representative of the production environment. The objective is to provide testers with close to real-life conditions without disrupting ministry operations. We found that all sampled initiatives conducted testing in a dedicated test environment.

Test Data

For testing to be close to real-life conditions, development teams should load **production-like** data in the test environment. The OCIO's security standards define strict conditions to use **actual data** for testing. Indeed, there is a risk that non-authorized individuals access **sensitive, financial**, or **personal information**, during testing.

Production-like data is data intended to simulate live, actual data being used in Production. Tools exist to mask or mimic production data without duplicating it.

We found that half of the development teams used actual data for testing. They did so without masking and expected controls in place, leading to some personal information being easily accessible. As this does not comply with the OCIO's security standards, we recommend that ministries ensure their development teams are familiar with the Government's policies and standards and confirm compliance.

Actual Data include:
- **Sensitive**: ongoing investigations or judgements, financial figures, passwords
- **Financial**: Credit card information, banking account numbers
- **Personal**: citizens' names, addresses, social insurance numbers.

The OCIO's security standards recommend not using any sensitive or personal information for testing a system. Where testing requires this data, development teams must remove or mask the data. In rare cases where testing must use sensitive or personal data, a list of conditions applies. They include:

- Receiving approval from the program area's executive management and the Ministry Chief Information Officer;
- Completing a specific Privacy Impact Assessment (PIA) and a Security, Threat, and Risk Assessment (STRA) (refer to section 5.2); and
- Reducing the presence of this data in the test environment and access to it to the minimum necessary.

5.2 Security, Privacy, and Financial Controls Assessments

The system stakeholder landscape is complex in the public sector. Through legislation or policy, several authorities across Government have responsibilities over ministry systems.

These authorities have implemented system assessments to protect government data and guide development teams to comply with legislation and policy. These assessments are also a valuable source of information for ministry executives. Main system assessments in Government include:

- **Privacy Impact Assessment (PIA):** development teams, with their Ministry Privacy Officer's support, complete it to determine if a system meets the requirements of the *Freedom of Information and Protection of Privacy Act*. They must submit their assessment to the Corporate Information and Records Management Office and address any findings prior to sign off.
- **Security Threat and Risk Assessment (STRA):** development teams, with their Ministry Information Security Officer's support, complete it to identify security risks and relevant actions to take. They must submit Statements of Acceptable Risks to the OCIO for sign-off.
- **Financial Risk and Controls Review (FRCR):** development teams must submit to the OCG an independent report that assesses the adequacy of the system's financial controls before implementation. A second review within three years of operation is also required. The objective is to reduce risks of loss, error, or fraud and to confirm compliance with accounting standards.

Policies and standards require completing the assessments before implementation. Development teams should engage with experts and initiate these assessments early to identify requirements and potential issues. It is important to do so in Agile as risks may appear as each sprint releases new functionalities regularly.

We assessed whether development teams underwent these assessments as relevant to their systems, completed them on time, and implemented the recommendations.

As we mentioned it in section 3.1, we found that development teams did not always engage with their ministry's privacy, security, and financial experts early in the process. These individuals can:

- Identify system requirements and controls before design work starts; and
- Determine whether the circumstances require undertaking these assessments.

We found that half of the sampled initiatives did not have all these assessments completed before implementation. In one case, the development team noted that completing these assessments would have impacted their timeframe. They completed them after implementation. Another team had no record supporting the decision to forego the assessments. Finally, the last team was not aware of the FRCR requirement.

Tracking

We also found that two development teams could not provide an update on the actions taken to implement the recommendations contained in these assessments. Therefore, we could not confirm that ministries had addressed the identified risks.

We recommend that development teams confirm with their experts on the requirements to undertake PIA, STRA, and FRCR for the sampled systems that did not have them completed. We also recommend that ministries track the implementation of FRCR, STRA and PIA action plans and communicate their status to their stakeholders.

5.3 Implementation

A system is ready for implementation once it satisfies users and stakeholders. Sponsors should provide their approvals. In the case of financial systems ministry finance executives and the OCG must approve the systems before implementation. They base their approvals on the FRCR report.

In Waterfall, implementations are generally larger and rarer than in Agile, so the risk of issues and impact on operations is higher. Success depends on detailed planning of activities to follow and steps to take in case of issues. In Agile, teams implement small system portions at the end of each sprint. This makes this process a lower-risk one.

We assessed whether development teams took adequate implementation steps in accordance with the methodologies adopted.

We found that most development teams had undergone some planning activities and recorded them. However, they did not always finalize and agree upon their plans. We also found that most teams could not provide a record of executive approvals. Poor record management impairs the accountability and transparency of government decisions.

Post-Implementation Review

Development teams should conduct a post-implementation review to assess the success of their system implementations. Success can be determined by looking back at the initial objectives of system implementations and expected benefits (refer to section 2.2).

At a more granular level, the post-implementation review should:

- Identify any defects after implementation and develop action plans.
- Include any outstanding issues from PIA, STRA and FRCR; and
- Identify opportunities for development teams to mature their own practices.

We assessed for the sampled initiatives, whether development teams conducted post-implementation reviews and involved system stakeholders.

We found that development teams generally overlooked this phase. Only one team performed an adequate assessment that included engaging users, linking to the objectives in the business case, and recording it in an OCIO template.

Another development team displayed some elements of a post-implementation review, such as ongoing issue management and gathering user feedback. However, without a thorough approach to guide the exercise, the tasks lacked structure and objectivity, and potential lessons learned may have been lost.

Without a comprehensive assessment, ministries may face various issues:

- They may disband development teams before fixing outstanding issues.
- Ministry executives and funding authorities may be unable to determine the actual value of the investments.
- They may not identify and address process inefficiencies.

As mentioned in section 2.2, we recommend development teams perform and record a post-implementation review as part of their benefit realization plans.

Recommendations:

(7) Ministries using Agile approaches should implement a transparent process for users to record and track their feedback on development work.

(8) Ministries should ensure their development teams are familiar with the Government’s policies and standards and confirm compliance, including Privacy Impact Assessments, Security Threat and Risk Assessments, and Financial Risk and Controls Reviews.

(9) Ministries should consult their experts on the requirements to complete Privacy Impact Assessment, Security Threat and Risk Assessment, and Financial Risk and Controls Review for the sampled systems that did not have them completed.

(10) Ministries should track the implementation of action plans from Privacy Impact Assessments, Security Threat and Risk Assessments, and Financial Risk and Controls Reviews and communicate the status to their stakeholders.

6.0 Ministry Frameworks

Within the Government, ministries continuously develop new systems or enhance existing systems. These initiatives require well-defined practices since they must comply with similar policies and standards and often face similar challenges.

A framework is a comprehensive approach that provides some standardization. Regardless of the circumstances of each system implementation, including the selected approach, a ministry-level framework can increase success rates by:

- Implementing processes that align with industry good practices across multiple system implementations;
- Improving internal controls and risk management activities, including ensuring compliance with policies and standards; and
- Increasing the quality of information supporting executive oversight and the transparency and accountability of ministry decisions.

We assessed whether the three sectors selected for this audit had developed, maintained, and enforced a framework for their system implementations.

We found that the sectors did not have any actively-used frameworks to support the sampled initiatives. Two sectors had published guidance aligned with Waterfall, but they had stopped maintaining them and development teams did not use this information. In the absence of a framework, we found that:

- Expectations in terms of processes, record management, and overall rigor differed between system implementations within the same sector.
- Successes were often reliant on a few leading individuals' knowledge, expertise, and commitment. Losing one of them could have led to system implementation into failure without a framework to fall back on.
- Ministries relied on their vendors' methodologies. Vendors are not often familiar with government policies and standards and may adjust their practices to meet their financial interests.

We found that one sector began developing a framework since our sample.
Its framework is adequate because it:

- Supports Waterfall, Agile and hybrid approaches; only Waterfall is in operation.
- Defines requirements according to the system implementations' risk profile. A higher-risk initiative has more stringent requirements to meet than a lower-risk initiative.
- Uses a tool to monitor progress and record key information.

This framework only applies to system implementations that benefit from the sector's information management branch's support. Other implementations within the sector do not benefit from it, which limits value.

We recommend that this sector carries on developing its framework by implementing Agile and hybrid approaches and scoping in any significant system implementations in the sector. For other sectors, we recommend they develop a system development framework and leverage existing government resources.

Government Resources

The Ministry of Citizens' Services has developed technical resources to support the ministry's system implementations. These resources include:

- **Service Design:** The Government Digital Experience Division offers Service & Content Design services. It has drafted a *Service Design Playbook* to guide ministry teams in the early stages of a system implementation, identify and validate program needs, and confirm executive commitment (link).
- **System Design & Delivery:** The OCIO is drafting guides to help ministries develop modern systems. It has shared the current versions of its *Hosting and Application Development Framework, Modernization Playbook* (link), and *Design & Delivery Playbook* (link).
- **GitHub:** The OCIO provides ministries with a tool to share code and coding practices. It also supports automated testing of code.
- **Common Components:** The OCIO is building a library of common components to help ministries develop systems quickly and provide a consistent online experience.

The OCIO is also developing a *Code of Practice* to expand on the Digital Principles (refer to section 1.2). As we identified several control deficiencies during this audit, we recommend that the OCIO continues collaborating with ministries in developing government resources. This will help ministries strengthen their system development practices.

Recommendation:

(11) Ministries should create or continue developing their system development framework and leverage existing government resources to develop them.

(12) The OCIO should collaborate with ministries to develop or update government resources to support ministries' system implementations.

Appendix A – Summary of Recommendations

1	Ministries should establish a list of key records and approvals to maintain and monitor throughout system implementations.
2	Ministries should require that their development teams establish benefit realization plans and monitor them, including updating their business cases (or equivalent records) when needed and conducting post-implementation reviews.
3	Ministries should require that development teams conduct and maintain a stakeholder analysis and ensure adequate engagement through their system implementations, in accordance with the approach selected.
4	Ministries should implement controls that ensure that deployments into production environments receive and record appropriate reviews and approvals.
5	Ministries should ensure that system administrator access enforces segregation of duty.
6	Ministries should require their vendors to maintain a quality management plan and provide third-party review reports periodically, when appropriate.
7	Ministries using Agile approaches should implement a transparent process for users to record and track their feedback on development work.
8	Ministries should ensure their development teams are familiar with the Government's policies and standards and confirm compliance, including Privacy Impact Assessments, Security Threat and Risk Assessments**,** and Financial Risk and Controls Reviews.
9	Ministries should consult their experts on the requirements to complete Privacy Impact Assessment, Security Threat and Risk Assessment, and Financial Risk and Controls Review for the sampled systems that did not have them completed.
10	Ministries should track the implementation of action plans from Privacy Impact Assessments, Security Threat and Risk Assessments, and Financial Risk and Controls Reviews and communicate the status to their stakeholders.
11	Ministries should create or continue developing their system development framework and leverage existing government resources to develop them.
12	The OCIO should collaborate with ministries to develop or update government resources to support ministries' system implementations.

Appendix B – Sampled System Implementations

Sector	Cost	Timelines	Description of Systems
Economy	$95K	Sept. 2019 – Oct. 2019	This system processes forestry workers' applications for financial assistance to help them retire early. The program had an initial budget of $40 M over three years. An additional allocation of $18 M was provided as part of Stronger BC. Budget 2022 provided $185 M over three years to provide supports for forestry workers, industry, communities, and First Nations affected by old growth logging deferrals. A portion of the $185M is allocated to the program. Actual spending to date has been lower than the budgeted amounts.
Justice and Public Safety	$4.9M	March 2018 – March 2023	The system modernized the Province's management of disputes between landlords and tenants. Citizens or organizations that request the sector's involvement in their disputes pay a fee, generating about $2 M of revenues annually.
	$6.5 M	June 2017 – Sept. 2022	This system enhanced provincial services supporting victims, including financial assistance and benefits to victims of violent crimes. The sector also uses it to manage service providers of community programs. The sector pays about $63 M every year through the processes depending on this system.
Natural Resources	$145K	Dec. 2019 – June 2022	The sector enhanced an existing system to process payments for licenses, permits and other authorizations, historically managed through a legacy system. It also developed system integrations, including with the Province's Corporate Financial System. The sector processes about $49 M of revenues annually through the enhancements made to the system.
	$6.2 M	May 2018 – Dec. 2021	This system enhances the Province's oversight of mines. The sector processes about $11 M of application and inspection fees through the data and processes relying on the system.

	$200 K	Jan. 2020 – Aug. 2021	The sector enhanced an existing system to allow proponents to bid electronically on petroleum and natural gas rights. A Court ruling paused the program soon after the enhancement became operational.

Appendix C – Abbreviations

Government or Province	Government of British Columbia
CPPM	Core Policy and Procedures Manual
FRCR	Financial Risk and Controls Review
IAAS	Internal Audit and Advisory Services
Initiatives	Sampled system implementations
OCG	Office of the Comptroller General
OCIO	Office of the Chief Information Officer
PIA	Privacy Impact Assessment
STRA	Security Threat and Risk Assessment
System	Information Technology System