



REPUBLIC OF THE PHILIPPINES
NATIONAL ECONOMIC AND DEVELOPMENT AUTHORITY

Office Circular No. 03 -2025

January 22, 2025

TO : **All Concerned**
 SUBJECT : **Information Systems and Software Development Guidelines**

1. RATIONALE

As part of its digital initiatives to improve its Information and Communications Technology (ICT) processes, the National Economic and Development Authority (NEDA) is adopting an information system (IS) that is responsive to the challenges of current and emerging technology-related systems. Through the Information and Communications Technology Staff (ICTS), NEDA shall implement automated business processes using harmonized standards for programming and software development.

These systems include the application of standard and high-quality source code across NEDA, including code review, application of tools, system stacking, testing, and other quality control measures, among others. These will improve processes, systems, outputs, and creativity, while minimizing errors, costs, and miscommunication.

2. COVERAGE

This Office Circular (OC) shall apply to all Offices and staffs within the NEDA Central Office (NCO) and NEDA Regional Offices (NROs) concerning the development and implementation of IS and software.

3. OBJECTIVES

The guidelines aim to:

- Establish standards and procedures for the development and/or acquisition of IS projects in the NCO and NROs, whether developed in-house or outsourced;
- Ensure that all IS and software developed or acquired are effective and efficient, meeting the criteria of functionality and cost-effectiveness; and
- Provide a clear understanding of the roles of key stakeholders (internal and external) in the development, acquisition, implementation, and maintenance of IS and software.

4. DEFINITION OF TERMS

- 4.1. **Agency** – refers to the National Economic and Development Authority, or any of its offices.
- 4.2. **Information Systems (IS)** – refer to organized systems designed to collect, process, store, and distribute information to support decision-making, coordination, control, analysis, and visualization within an organization or across multiple organizations.
- 4.3. **Software** – refers to a collection of instructions, data, or programs used to operate computers and perform specific tasks.
- 4.4. **Project** – refers to the development of IS or software solutions aimed at meeting specific business needs and objectives.
- 4.5. **Information Systems Strategic Plan (ISSP)** – refers to the plan that contains the agency's overall strategy which involves medium-term (3–5-years) planning for its ICT thrusts, strategies and programs for development.
- 4.6. **Process Owner** – refers to the individual/group who holds responsibility for overseeing and managing a specific process within an organization, ensuring its efficiency, effectiveness, and alignment with organizational objectives.
- 4.7. **Open Source** – refers to software whose source code is made freely available for use, modification, and redistribution by anyone.

5. PROJECT TYPES

Projects are categorized based on their development and implementation approach, as follows:

- a. **In-House Project** – refers to IS or software development projects developed by internal developers or computer programmers employed by NEDA, whether on a job hire, contractual, or permanent basis, in both NCO or NROs.
- b. **Outsourced Project** – refers to IS or software development projects that are contracted to external developers or consultants not employed by NEDA.

6. GENERAL GUIDELINES

6.1. Project Scoping

- a. The project scope shall be clearly defined and documented during the initiation phase. It shall outline the objectives, deliverables, timelines, and required resources.
- b. Both in-scope and out-of-scope items shall be explicitly stated to prevent misunderstandings and scope creep.
- c. Stakeholders shall be actively consulted during the project scoping phase to ensure that all requirements are thoroughly considered.
- d. For in-house projects, the project scope shall be established during the initiation process.

- e. For outsourced projects, the scope shall be clearly defined in the terms of reference, with the process owner leading the scope definition in collaboration with key ICTS personnel.
- f. The project scope document shall be reviewed and approved by all relevant stakeholders before the project begins.
- g. Any changes to the project scope shall adhere to a formal change management process to evaluate the impact on timelines, costs, and resources.

6.2. Creation of Project Development Team

- a. A dedicated project development team shall be formed for each IS and software development project. For smaller projects, one (1) developer and one (1) counterpart from the end-users may be sufficient.
- b. For in-house projects, the team shall consist of key personnel from the ICTS, along with relevant staff members involved in the project.
- c. For outsourced projects, the external provider shall establish its own project development team. The terms of reference for these projects shall include a provision requiring bidders to designate personnel and specify their qualifications.
- d. Team members may take on multiple roles depending on the project scope.
- e. Process and data owners shall always be included in the project development team to ensure successful implementation and monitoring.
- f. Attached as **Annex A** are the detailed composition, functions, and qualifications of the project development team, which can be customized based on the specific needs of each project.

6.3. Methodology

- a. The Agency shall adopt the Agile Software Development Methodology for both in-house and outsourced IS and software development projects, emphasizing iterative development, cross-functional team collaboration, and effective adaptation to change. The approval of the ICTS Director shall be required for any changes to the methodology.
- b. Detailed information regarding the phases and processes of the proposed methodology can be found in **Annex B**.

6.4. System Ownership

- a. The design, source code, and any licenses acquired during the development and completion of the project, regardless of its type, along with all information and data generated by the system, shall remain as exclusive property of the agency.
- b. Projects completed at the NCO that are downloaded or duplicated in the NROs may not be modified without the authorization and approval of the ICTS Director.
- c. In terms of intellectual property rights, these Guidelines shall enforce Section 178.3 of Republic Act No. 8293, also known as the Intellectual

Property Code of the Philippines, and shall adhere to Clause 35, General Conditions of Contract, as outlined in the Philippine Bidding Documents for consultancy services.

6.5. Development Tools and Technology

The Agency shall use a hybrid strategy by using open-source programming languages, platforms, and tools in the development of IS and software projects. This will allow for cost-effective software solutions for the Agency without incurring costly licensing expenditures.

7. RESPONSIBILITIES

7.1. Executive Committee/Project Sponsors

- a. Approve project-related documents and act as the decision-making body for significant issues encountered during the project lifecycle; and
- b. Shall consist of Undersecretaries, Assistant Secretaries, and Directors from the offices involved in the project.

7.2. ICTS and Regional IT Staffs

- a. Provide critical technical expertise in the development of new IS;
- b. Review project proposals submitted by external consultants involved in the development of IS;
- c. Provide staging and production environments to the project development team;
- d. Be responsible for the administration and security of the server after deployment of the project; and
- e. Serve as designated system administrators of the implemented IS.

7.3. Process/Data Owners

- a. Provide all necessary data requirements or specific processes essential for the project;
- b. Act as end-users and participate in all testing activities;
- c. Collaborate with the project development team, and provide critical feedback; and
- d. Serve as specific system owners.

8. SEPARABILITY

If any provision of these Guidelines or the application of such a provision to any person or circumstance is declared invalid, the remainder of these Guidelines or the application of such provision to other persons or circumstances shall not be affected by such declaration.

9. REVIEW/AMENDMENT

These Guidelines shall be reviewed and updated or amended periodically as may be necessary.

10. REPEALING CLAUSE

All provisions of existing NEDA issuances inconsistent with these Guidelines are hereby repealed, amended, or modified accordingly.

11. EFFECTIVITY

These Guidelines shall take effect immediately.


ARSENIO M. BALISACAN, PhD
Secretary



ANNEX A

Project Development Team Composition and Qualifications

The project development team must include at least the following members with their corresponding functions:

1. **Project Manager**
 - a. Oversees the entire project development process;
 - b. Ensures adherence to programming standards;
 - c. Tracks compliance with established guidelines;
 - d. Monitors project deliverables;
 - e. Addresses any deviations from requirements and timelines;
 - f. Supervises development team members;
 - g. Sets the overall direction of the project; and
 - h. Must be assigned by the business process owner and the ICTS to manage the project's functional and technical aspects, respectively.
2. **Information Systems (IS) Analysts**
 - a. Coordinate the gathering, refining, and prioritization of user requirements;
 - b. Develop design specifications based on user needs;
 - c. Identify appropriate technology stacks for the project; and
 - d. Maintain constant communication with the process owner throughout the project.
3. **Business Analysts**
 - a. Typically represent the end-users;
 - b. Coordinate with IS Analysts to discuss the business process flow;
 - c. Explore opportunities to streamline processes; and
 - d. Work on transforming business processes into a computerized system.
4. **Software Developers**
 - a. Translate design specifications into code;
 - b. Be responsible for actual programming of the system; and
 - c. Develop information systems based on the requirements provided by process owners.
5. **Quality Analysts**
 - a. Ensure that the software meets the specified requirements and quality standards;
 - b. Conduct technical reviews of the software;
 - c. Perform software testing to identify issues and ensure functionality; and
 - d. Enforce software standards throughout the development process.
6. **Documentation Analysts**
 - a. Be responsible for producing user documentation;
 - b. Assist the team throughout the project timeline; and

- c. Provide logistical support for the project by organizing team meetings and drafting meeting notes and highlights.

7. Network Administrators

- a. Be responsible for delivering essential network services for the information systems project;
- b. Oversee backup operations to ensure data safety; and
- c. Implement crucial network security protocols to safeguard the security and integrity of data within the system.

8. Technical Support and Maintenance Staff

- a. Be involved in the maintenance of existing and fully implemented software; and
- b. Be responsible for bug fixing and updating technology stacks.

Below are the suggested minimum qualifications for the members of the project team:

Member	Education/Work Experience	Skills
Project Manager	<ul style="list-style-type: none"> • Bachelor's degree in Computer Science, Information Technology, Business Administration, or related field; • PMP (Project Management Professional) or similar certification is preferred; • 5-7 years of experience in project management within the IT sector; and • Proven track record of successfully managing complex projects from inception to completion. 	<ul style="list-style-type: none"> • Strong leadership and team management skills; • Excellent communication and interpersonal skills; • Proficiency in project management tools (e.g., MS Project, JIRA); • Strong organizational and multitasking abilities; and • Risk management and problem-solving skills.
IS Analyst	<ul style="list-style-type: none"> • Bachelor's degree in Computer Science, Information Systems, or related field; and • 3-5 years of experience in information systems analysis or a similar role. 	<ul style="list-style-type: none"> • Strong analytical and problem-solving skills; • Proficiency with database management and data modeling;

		<ul style="list-style-type: none"> • Familiarity with various software and hardware systems; • Knowledge of SQL and other programming languages; and • Good understanding of business processes and workflows.
Business Analyst	<ul style="list-style-type: none"> • Bachelor's degree in Business Administration, Computer Science, Information Systems, or related field; and • 3-5 years of experience as a business analyst or in a related role. 	<ul style="list-style-type: none"> • Strong analytical and critical thinking skills; • Excellent communication and presentation skills; • Proficiency in business analysis tools (e.g., MS Visio, JIRA); • Ability to gather and document requirements; and • Understanding of business process modeling and improvement.
Software Developer	<ul style="list-style-type: none"> • Bachelor's degree in Computer Science, Software Engineering, or related field; and • 2-5 years of experience in software development. 	<ul style="list-style-type: none"> • Proficiency in programming languages (e.g., Java, PHP, C++, Python, JavaScript); • Knowledge on MVC framework (e.g., Laravel, CodeIgniter); • Strong understanding of software development life cycle (SDLC); • Experience with version control systems (e.g., Git); • Knowledge of databases and SQL; and

		<ul style="list-style-type: none"> • Problem-solving and debugging skills.
Quality Analyst	<ul style="list-style-type: none"> • Bachelor's degree in Computer Science, Information Systems, or related field; and • 2-4 years of experience in quality assurance or software testing. 	<ul style="list-style-type: none"> • Proficiency in QA methodologies, tools, and processes; • Experience with automated testing tools (e.g., Selenium, QTP); • Strong analytical and problem-solving skills; • Attention to detail; and • Understanding of software development life cycle (SDLC) and agile methodologies.
Documentation Analyst	<ul style="list-style-type: none"> • Bachelor's degree in Technical Writing, Information Systems, English, Communications, or a related field; and • 2-4 years of experience in technical writing or documentation. 	<ul style="list-style-type: none"> • Strong writing, editing, and proofreading skills; • Proficiency with documentation tools (e.g., MS Word, Adobe Acrobat); • Ability to create clear, concise, and user-friendly documentation; • Excellent organizational and time management skills; • Basic understanding of technical concepts and terminology; and • Ability to work collaboratively with technical and non-technical teams.
Network Administrator	<ul style="list-style-type: none"> • Bachelor's degree in Computer Science, Information Technology, or related field; 	<ul style="list-style-type: none"> • Strong understanding of network protocols and technologies;

	<ul style="list-style-type: none"> • Relevant certifications (e.g., Cisco Certified Network Associate [CCNA], CompTIA Network+); and • 3-5 years of experience in network administration or IT infrastructure. 	<ul style="list-style-type: none"> • Experience with network hardware and software; • Proficiency in network security measures and best practices; • Problem-solving and troubleshooting skills; and • Ability to manage multiple tasks and projects simultaneously.
Technical Support and Maintenance Staff	<ul style="list-style-type: none"> • Bachelor's degree in Computer Science, Information Technology, or related field; • Relevant certifications (e.g., CompTIA A+, Microsoft Certified IT Professional [MCITP]); and • 1-3 years of experience in technical support or IT maintenance. 	<ul style="list-style-type: none"> • Strong technical and troubleshooting skills; • Excellent customer service and communication skills; • Familiarity with various operating systems (e.g., Windows, macOS, Linux); • Knowledge of hardware and software troubleshooting; and • Ability to work under pressure and handle multiple tasks simultaneously.

ANNEX B

Information Systems (IS) Development Methodology

The preferred approach for IS development is the Agile Software Development methodology. This methodology follows an iterative and incremental process, allowing for flexibility and adaptability to change. It emphasizes the continuous delivery of functional software, promotes collaboration, and prioritizes customer feedback. The following outlines the key phases and processes involved in Agile Software Development:

1. Initiation Phase

This phase, also known as pre-planning or project kickoff, involves categorizing the project into one of the following types:

- a. **New IS Development Project:** Focused on developing a completely new information system or software to automate existing manual business processes. All new IS development and acquisition projects are required to be included in the NEDA ISSP.¹ Projects not included in the ISSP will undergo the review and approval of the ICTS Director.
- b. **IS Enhancement Project:** Involves significant modifications to the functionality, features, performance, or interfaces of existing IS. These projects may continue using the same environment as the current system.
- c. **IS Re-engineering Project:** Entails redesigning the data, architecture, and logic of existing systems to perform the same functions more efficiently.
- d. **IS Acquisition:** Refers to the procurement of IS or software, which may be customized to meet the Agency's specific needs.

The project begins by defining its vision, objectives, and scope. Key stakeholders are identified, and the project development team is established to ensure alignment with the project's goals. An initial detailed IS requirement document is created, outlining the features and user stories that will guide the development process. Additionally, all project-related activities should be documented in the chronology of events.

2. Planning Phase

In this phase, detailed planning is conducted for each sprint. The team prioritizes features from the detailed IS requirement document, estimates tasks, and sets clear sprint goals to ensure focused development. A high-level release plan is also established to guide future iterations and key deliverables.

¹ <https://dict.gov.ph/wp-content/uploads/2024/02/Department-Circular-No.-001-s.-2024.pdf>

3. Execution Phase

The project development team builds the product incrementally through sprints. Daily stand-ups, continuous integration, and frequent testing ensure smooth progress. Code is developed, tested, and integrated iteratively, allowing flexibility for changes and adaptations.

During the execution phase, the following elements are to be adhered to:

a. Technology

- Programming Language: PHP – Chosen for its versatility, scalability, and strong support in web development.
- Framework: Laravel – A powerful, flexible framework that accelerates development with its extensive tools, features, and MVC architecture, ensuring clean and efficient coding.
- Database Management System: MySQL/MariaDB – Reliable and widely used relational databases, known for their performance, scalability, and ease of use.
- Web Server: Apache HTTP Server – A stable and flexible web server that efficiently handles web content delivery and offers robust performance.
- Frontend Technologies: HTML, CSS, JavaScript – Essential for creating interactive, responsive, and user-friendly web interfaces.
- JavaScript Frameworks: jQuery, Ajax, Vue.js – Enable advanced frontend functionalities and dynamic content interaction, improving the user experience and responsiveness of web applications.
- Server Operating System: Windows/Linux Server – Provides a solid and secure platform for hosting and managing web applications, offering flexibility and compatibility with various technologies.

b. Coding Style and Conventions

- Use descriptive and meaningful names for variables, functions, classes, and methods.
- Use `lowercase_with_underscores` for variables, `camelCase` for functions, and `StudlyCase` for classes and interfaces.
- Use consistent naming conventions, indentation, and formatting throughout the codebase.
- Use an indentation of 4 spaces for PHP code.
- Keep lines within a maximum length of 120 characters to improve readability.
- Place opening braces on the same line as the control structure declaration and use consistent spacing.
- Write meaningful comments to explain the purpose and functionality of code blocks.

- If the system is developed by an outsourced consultant, code documentation should be available and submitted to ICTS for proper handover.

c. Version Control and Integration

- Create a central repository for your project using a version control system like Git or Bitbucket.
- Establish the repository structure, including branches for development, testing, and production releases.
- Write clear and descriptive commit messages summarizing the changes made.
- Implement a code review process to maintain code quality and consistency.
- Tag important milestones, releases, or versions of your software to mark significant changes.

d. Testing Practices

- To ensure that the developed system aligns with the business requirements of the end-users and staff, user testing should be conducted 1-2 weeks prior to the system's release. If end-users do not comply with this testing requirement, adjustments to the agreed target release date may be necessary.

e. Performance Considerations

- Proper data typing ensures the right kind of data stored in a field and adds data validation and integrity in the database. Each data has specific size requirement to optimize query execution.
- Use proper indexing to improve query performance.
- Limit attributes selected in each query based on data needed in a function. Never use '*' or select all for any production queries.
- Designing table relationships is essential in all IS. Relationships involve joining multiple tables which can be resource-intensive, especially for large datasets. Properly indexed foreign keys can improve the performance of join operations.
- Ensure proper structure and clear relationships in JOIN clauses between database tables.
- Clean up HTML document by proper placing of styles/CSS and JavaScript.
- Optimize CSS performance by minimizing bloated CSS.
- Minifying CSS, JavaScript, and HTML can help eliminate unnecessary characters within a file like indentations.
- Optimize images by resizing the image or using image optimization tools.

f. Security Practices

- Apply input validation and proper error handling to prevent common vulnerabilities like SQL injection and Cross Site Scripting.
- Use CSRF protection, input validation, and authentication middleware.
- Encrypt sensitive data using encryption algorithms such as DES or Blowfish.
- Regularly perform Vulnerability Assessment and Penetration Testing to identify and address vulnerabilities in the application.

g. Accessibility

- Design user interfaces with accessibility in mind, ensuring compatibility with screen readers and keyboard navigation.
- Use semantic HTML markup and ARIA attributes to enhance accessibility for users with disabilities.

h. Deployment and Configuration

- Maintain separate configurations for development, staging, and production environments.
- Store sensitive information like API keys and database credentials securely using environment variables or a secrets management system.
- Use dependency management tools like Composer, npm, pip, or Maven to manage project dependencies.

i. Legal and Licensing Considerations

- Utilize exclusively open-source technologies for all IS development endeavors.
- Ensure that your project complies with the terms of the licenses of all third-party dependencies and libraries used.

4. Review and Retrospective Phase

At the end of each sprint, the team reviews the completed work with stakeholders to gather feedback. A retrospective meeting is held to reflect on the sprint's successes and areas for improvement, driving continuous process refinement.

5. Release Phase

Once significant features are completed, the product is prepared for release. Final testing, debugging, and deployment are conducted. Documentation and user training may also be provided to facilitate the system's smooth adoption.

6. Continuous Feedback and Iteration

After the release, user feedback is collected to refine and improve the system. The detailed IS requirement document is updated accordingly, and the team continuously enhances the development process, adapting to evolving requirements in subsequent sprints.

7. Project Acceptance

Project acceptance is an ongoing process throughout the development lifecycle, based on defined acceptance criteria for each user story to meet stakeholders' expectations at the end of every sprint. The product owner and stakeholders regularly review and approve features during sprint reviews, enabling early feedback and adaptation. Final acceptance builds on this iterative process, ensuring the product delivers the intended business value. Attached as **Annex C** is the template for project acceptance, which is one of the required output documents for this phase.

8. Project Close-out

The project close-out marks the final stage of IS development. In this phase, all project activities are completed, and the project is formally concluded. A project close-out meeting is held to ensure all essential documents, handovers, lessons learned, and challenges encountered by the team are addressed. For outsourced projects, the consultant must provide a detailed user manual and technical documentation prior to the close-out.

Annex D contains the combined form templates for the chronology of events, sprint plan, and user feedback monitoring.

ANNEX C

PROJECT ACCEPTANCE FORM DOCUMENT

[To be accomplished by the Information and Communications Technology Staff (ICTS)/Finance and Administrative Division (FAD)/Development Research Division (DRD) or Outsourced Developer]

PROJECT ACCEPTANCE AND SIGN-OFF FORM	
Project Name:	
Start Date:	Completion Date:
Project Manager (Staff/Division):	End-User (Staff/Division):
Project Description: (Brief description of the project)	
Project Deliverables: (Identify the project deliverables)	
By signing this document, I acknowledge that we have delivered all the stated deliverables to the requesting party at the agreed quality levels.	By signing this document, I acknowledge that we have received all the stated deliverables from ICTS at the agreed quality levels.
Project Team Members & Signature:	End-User's Representative Name and Signature:
Date:	Date:
ICTS Director:	End-User Director:
Date:	Date: