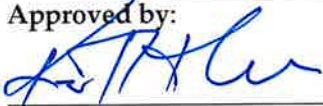



District of Columbia Department of Health <h2 style="text-align: center;">Agile Software Development and DevOps</h2>		PROCEDURE 735.000 Implementing Office: Office of the Director/Office of Information Technology Training Required: Yes Originally Issued: July 11, 2018 Revised/Reviewed: <p style="text-align: center;">JUL 27 2022</p> (Second Revision)
Approved by:  LaQuandra S. Nesbitt MD, MPH; Agency Director	Review by Legal Counsel:  Phillip Husband, Esq.; General Counsel	Effective Date: JUL 27 2022 Valid Through Date: JUL 27 2025

I. Authority	Reorganization Plan No. 4 of 1996; Mayor’s Order 1997-42; This SOP shall supersede and replace SOP 735.000 Information Technology Life Cycle, issued on October 28, 2019.
II. Reason for the Policy	<p>Information technology (IT) is a critical element of daily operations at DC Health. IT solutions enable DC Health to collect, analyze, secure, report, and share data that support public health policies and programs. To ensure that DC Health is a continuously improving organization, IT solutions must keep pace with ever-changing requirements at the program level. Recognizing the need to deliver IT solutions more quickly, the DC Health Office of Information Technology (DC Health IT) has adopted the Agile methodology. This methodology delivers an IT solution incrementally in a series of rapid development cycles, or sprints, executed in weeks rather than months. A protocol is necessary to operationalize this methodology within DC Health IT.</p> <p>In addition, it is essential that the gathering of functional requirements be organized, that those requirements are documented and validated by program teams, and that DC Health IT only commits to deliver the features that it can deliver on specified dates. As such, DC Health IT will organize all software development and configuration activities using a DevOps solution.</p>

<p>III. Applicability</p>	<p>This SOP shall apply to all DC Health employees, contractors, interns, and summer youth employees. This SOP shall also apply to vendors that are party to an agreement with DC Health to develop or configure an IT solution. These individuals are referred to collectively herein as “employees” or “DC Health employees.”</p>
<p>IV. Policy Statement</p>	<p>Software development and configuration is part of the portfolio of DC Health IT. The Chief Information Technology Officer (CITO) is the accountable manager over all tasks assigned to DC Health IT. Within DC Health IT, agile software development and configuration is delegated to the Data and Applications Program Manager.</p> <p>All software development and configuration will occur in the context of an IT project. An IT project will formally launch in a kickoff meeting and be documented in the Master Portfolio Listing (MPL). See SOP 702.000 Master Portfolio Listing for additional guidance on the MPL. The Data Applications and Program Manager will approve and schedule a project in collaboration with the CITO and the Senior Deputy Director (SDD) of the administration that manages the business process(es) to be automated/improved.</p> <p>The Data and Applications Program Manager will assign an IT Lead to each IT project. The IT Lead may be a Business Analyst (BA) or a Solution Architect (SA) and may be a DC Health employee or a contractor. The Data and Applications Program Manager has the discretion to determine which projects are most appropriate to be led by a BA or an SA depending on the skillsets required. The Data and Applications Program Manager may assign additional developers to support the project on a case-by-case basis. This includes temporary assignments to support specific deliverables, or ensure key deadlines are met.</p> <p>The SDD of the lead administration, or designee, will designate a product owner for every project. A single product owner is preferred, but the duties of this role may be shared among a small team if time constraints, or essential skills sets dispersed among several employees, contraindicate a single product owner. The product owner is ideally an employee who has working</p>

	<p>knowledge of the business process being automated/improved and is well-versed in the desired future-state process.</p> <p>All projects will be documented in the DC Health DevOps solution. The DevOps solution will organize incremental scheduled releases (“sprints”) into epics and user stories. The Data and Applications Program Manager, or designee, will set a sprint schedule on a case-by-case basis with consideration to the availability of resources, the relative urgency of the requirements, the size or complexity of the build, and/or any other relevant factors. At a minimum, the sprint schedule will consist of the following milestones in each sprint:</p> <ol style="list-style-type: none">1. Roadmap development. The deadline for having the next sprint’s epics prioritized and defined.2. Sprint defined. All requirements gathering meetings have been held, user stories have been drafted, and have been approved by the product owner;3. User Acceptance Testing (UAT).4. Deployment cadence. The schedule for deploying sprint deliverables into the production environment, e.g., every two weeks, or every four weeks. <p>The product owner is responsible for the following:</p> <ol style="list-style-type: none">1. Ensuring all functional requirements are communicated to the IT Lead and the development team, either by communicating those requirements directly, or by referring the IT Lead to the appropriate program staff;2. Attending the roadmap meetings, and ensuring that the epics for each sprint are prioritized according to the most urgent and mission-critical needs;3. Approving all user stories by a pre-determined deadline in every sprint, correcting any requirements in a user story that have been captured incorrectly;4. Assigning a team member to test each user story requiring a business-side tester during UAT. <p>The IT Lead and the IT team are responsible for the following:</p> <ol style="list-style-type: none">1. Assigning sprint points to all user stories;2. Deciding upon the tool/platform to be used in solutioning a requirement;3. Developing all user stories in a development environment;
--	---

4. Completing unit testing in the quality assurance (QA) environment to identify bugs;
5. Fixing identified bugs in the development and QA environments;
6. Migrating the new content into the UAT environment through the DevOps solution in time for the UAT phase of the sprint to begin.
7. Creating a test script for each user story requiring a business-side tester.
8. Creating a listview in the DevOps solution of user stories requiring a business-side tester.

Assigned testers will test each user story to which they are assigned. UAT shall be completed by close-of-business two business days prior to the planned deployment. Testers will use test scripts to test an assigned user story and ensure that the application is performing as expected. If the test does not produce expected behavior, the tester will log a bug in the DevOps platform with detailed information on the unexpected behavior, including a screen shot if possible. If the test produces expected behavior, the tester will accept the user story in the DevOps platform.

The IT Lead and IT team will endeavor to resolve all identified bugs by the morning of the planned deployment and refer back to the tester for re-testing. User stories that are not yet accepted on the day of deployment may be held out of the deployment at the discretion of the product owner in coordination with the IT Lead.

The Data and Applications Program Manager has the discretion to order smoke testing for any code migrated into the production environment.

The product owner may request a hotfix to deploy outside of the normal sprint schedule. Hotfixes should be highly targeted in scope, and should be reserved for the following scenarios:

1. Deploying a mission critical user story into production that was not ready to deploy in the previous sprint due to needing additional bug fixes; or
2. Deploying a fix to a bug that was only discovered in production after the user story was deployed; or

3. Deploying a user story to respond to a time-sensitive deliverable such as an emergency request from DC Health leadership.

The CITO has the discretion to order a security review of any new or enhanced application. DC Health IT will comply with all security reviews requested by the District of Columbia Office of the Chief Technology Officer (OCTO) and comply with all OCTO security policies and directives.

The IT Lead is responsible for ensuring that release notes are sent to all users of the application at the time of a new release being deployed. At a minimum, release notes will include:

1. A revised entity relationship diagram;
2. Screenshots of new or enhanced features;
3. Written instructions accompanying the screenshots;
4. Any new or enhanced data mappings for integration or reporting purposes.

The Data and Applications Program Manager has the discretion to arrange end user training in addition to the release notes based upon the need, demand, and capacity to provide such training.

DC Health IT shall ensure that all users have access to a support ticket queue. DC Health IT will designate a team to staff both Tier 1 and Tier 2 support. Each Tier may be assigned to either DC Health IT or a contract vendor. The CITO, or designee, will make a case-by-case decision on how to staff support needs for each application in the portfolio.

Each product owner is responsible for disseminating the process for requesting service and enhancements to end users. At a minimum, the product owner will:

1. Ensure that at least two users of any application have permissions to create a support ticket;
2. Ensure everyone with those permissions are trained in creating support tickets.

Any employee out of compliance with any part of this SOP may be subject to commensurate disciplinary action.

<p>V. Definitions & Acronyms</p>	<p>Agile- A software development methodology in which an IT solution is implemented incrementally over a succession of short development cycles called sprints. Each sprint adds to the features and functionality of the IT solution.</p> <p>BA- Business Analyst</p> <p>Bug- An error, flaw or fault in a computer program or system that causes it to produce an incorrect or unexpected result, or to behave in unintended ways.</p> <p>Bug fix- A revised program file or patch that corrects a software bug.</p> <p>CITO- Chief Information Technology Officer</p> <p>Code coverage- A software testing metric that determines the number of lines of code that is successfully validated under a test procedure, which in turn, helps in analyzing how comprehensively a software product is verified.</p> <p>Code review- A software quality assurance activity in which one or several people check a program mainly by viewing and reading parts of its source code after implementation or as an interruption of implementation. At least one of the persons must not be the code's author. The persons performing the checking, excluding the author, are called "reviewers".</p> <p>DC Health IT- DC Health Office of Information Technology</p> <p>Development environment- A workspace for developers to make changes to the application separate from a live environment. The development environment is where all work occurs to develop a user story prior to migration to the testing environment.</p> <p>DevOps- A set of practices that combines software development (Dev) and IT operations (Ops). It aims to shorten the systems development life cycle and provide continuous delivery with high software quality. DevOps is complementary with Agile software development; several DevOps aspects came from the Agile methodology.</p>
---	--

	<p>Enhancement- A change to an existing IT solution. Examples of enhancements include, but are not limited to: adding a new feature or functionality, correcting a problem or bug, or improving the user experience. Enhancements are facilitated in a sprint, or a series of sprints, each producing a new iteration.</p> <p>Entity Relationship Diagram- A graphical representation that depicts relationships among people, objects, places, concepts or events within an information technology (IT) system. An entity relationship diagram uses data modeling techniques that can help define business processes and serve as the foundation for a relational database.</p> <p>Epic- A large user story that cannot be delivered as defined within a single iteration or is large enough that it can be split into smaller user stories. An epic consists of the requirements for a more cohesive IT deliverable such as a feature or component.</p> <p>Expected behavior- An IT solution performing as the user story indicates it should perform.</p> <p>Feature branch- A copy of the main codebase in the development environment where an individual or team of software developers can work on a new feature until it is complete.</p> <p>Functional requirement- The specifications of an IT product's function. Functional requirements define what precisely a software must do and how the system must respond to inputs. Functional requirements define the software's goals, meaning that the software will not work if these requirements are not met.</p> <p>Functionality testing- A type of software testing that validates the software system against the functional requirements/specifications. The purpose of functional tests is to test each function of the software application by providing appropriate input and verifying the output against the functional requirements.</p> <p>Hotfix- A deployment outside the normal sprint schedule. Hotfixes are executed ad hoc to correct specific problems requiring time sensitive solutions.</p>
--	--

	<p>Integration test- Integration tests determine if independently developed units of software work correctly when they are connected to each other.</p> <p>Load test- A type of performance testing that simulates a real-world volume of transactions on any software, application, or website.</p> <p>MPL- Master Portfolio Listing</p> <p>OCTO- District of Columbia Office of the Chief Technology Officer</p> <p>Performance test- A non-functional software testing technique that determines how the stability, speed, scalability, and responsiveness of an application holds up under a given workload.</p> <p>Production environment- The set of computers where finished, user-ready software is deployed and executed. The production environment is the only environment where live data corresponding to real persons, including personally identifiable information may be input and/or stored.</p> <p>Pull request- An event that takes place in the development environment when a developer is ready to begin the process of merging new code changes with the main project repository.</p> <p>QA environment- Quality assurance environment. A close simulation of the production environment where unit testing occurs.</p> <p>Regression test- A type of software testing to confirm that a recent program or code change has not adversely affected existing features.</p> <p>SDD- Senior Deputy Director</p> <p>Smoke Testing- A software testing process that determines whether the deployed software build is stable or not in the production environment.</p>
--	---

	<p>Sprint- A rapid development cycle during which a limited and defined set of requirements is incorporated into an IT solution. An IT solution may go through several sprints before all requirements are fully realized.</p> <p>Sprint points- An approximation of the effort necessary to deliver a user story to completion. Sprint points are assigned to all user stories in a sprint to ensure that promised deliverables align with available capacity and projects remain on schedule.</p> <p>Static code analysis- A tool that allows a developer to detect code inefficiencies and errors before they create a larger problem in the software application. Static code analysis can automatically monitor developers' code and enforce implementation best practices. IT leadership can review the results and prioritize rules depending on programmatic needs as well as decide whether code can be moved to an upper environment or not.</p> <p>Test script- A set of instructions that will be performed on the system under test to test that the system functions as expected.</p> <p>Testing environment- A setup of software and hardware for the testing teams to execute test cases. In other words, it supports test execution with hardware, software and network configured.</p> <p>Tier 1 Support- The support level responsible for basic customer issues. The first job of a Tier 1 specialist is to gather the customer's information and to determine the customer's issue by analyzing the symptoms and figuring out the underlying problem. This level should gather as much information as possible from the end user. Once the identification of the underlying problem is established, the specialist can begin sorting through the possible solutions available. Technical support specialists in this group typically handle straightforward and simple problems while possibly using some kind of knowledge management tool.</p> <p>Tier 2 Support- Tier 2 technical support services are responsible for assisting Tier 1 personnel in solving basic technical problems and for investigating elevated issues by confirming the validity of the problem and finding solutions related to these more complex issues.</p>
--	--

	<p>UAT- User Acceptance Testing. UAT is conducted on a product by the end users of the product for approval for production release.</p> <p>User story- The basic functional increment of IT development and configuration work. Each user story is expected to yield, once implemented, a contribution to the value of the overall product. User stories are subdivisions of an epic.</p>
<p>VI. Procedures</p>	<p>Procedure A: Roadmap Setting</p> <ol style="list-style-type: none"> 1. The IT Lead, the product owner, and any other relevant stakeholders will convene the roadmap meeting per the sprint schedule. 2. The attendees will identify the epics for the next two upcoming sprints. 3. The attendees will define the high-level requirements for each epic to inform how the epic will be sized. 4. The IT Lead will list the size of each epic (large, medium, small). <p>Procedure B: Requirements Gathering</p> <ol style="list-style-type: none"> 1. The IT Lead and product owner will collaborate to designate workgroups for each epic in the sprint. 2. The IT Lead will schedule requirements meetings to occur in the allotted window of time listed in the sprint schedule. 3. The IT Lead, or designee, will facilitate all requirements meetings. 4. If additional information is required from the product owner and other program employees, they will provide that information by the deadline for requirements submission in the sprint schedule.

5. The IT Lead, or designee, will draft the user stories in the DevOps solution by the assigned deadline in the sprint schedule.
6. The product owner will review the user stories in the DevOps solution, approving those that are correct, and noting any corrections to the functional requirements as applicable.
7. The IT Lead will revise user stories as necessary per the feedback from the product owner.
8. The IT Lead will document any technical specifications in the user story as necessary to guide the developers in solutioning the requirement.
9. The IT Lead will assign developers to work on user stories.

Procedure C: Coding and Configuration

1. The developer will write code in the development environment as assigned per the functional requirements in the user story.
2. The developer will commit the code to a feature branch and create a pull request.
3. The IT Lead will conduct a code review and approve the pull request once the code is suitable to move to the QA environment. The IT Lead has the discretion to engage another IT employee to conduct a second code review if the additional feedback would help validate the new code.

Procedure D: Unit Testing

1. The assigned tester will run a static code analysis using a tool approved by the CITO.
2. The tester will identify any issues leading to an unacceptable score on the static code analysis and report back to the IT Lead.

3. The IT Lead will direct the developer to address any discovered issue.
4. The assigned tester will conduct the following tests on the new code in the QA environment, reporting back any issues to the IT Lead:
 - a. Functionality testing;
 - b. Regression testing;
 - c. Integration testing;
 - d. Load testing;
5. At the discretion of the CITO, the IT Lead will order performance testing across integration boundaries.
6. The IT Lead will migrate the approved new code to the UAT environment for user acceptance testing.
7. The IT Lead, or designee, will create a test script for each user story requiring program-level testing.

Procedure E: User Acceptance Testing

1. The product owner will assign testers to all user stories that require program-level testing.
2. The IT Lead will create a listview in the DevOps solution that includes all user stories needing program-level testing, and the assigned tester.
3. The designated testers will test their assigned stories using the test scripts provided within the user story record in the DevOps solution.
4. The tester will document any bugs identified and flag the user story with the bug in the DevOps solution, including as much detail as practicable to guide the developer in designing a bug fix.
5. A developer will review all reported bugs, triage, and design a bug fix in the development environment.

	<ol style="list-style-type: none">6. The developer will migrate the bug fix into the testing environment.7. The developer will notify the tester to re-test the user story.8. The tester will re-test any user stories with bug fixes in place.9. The tester will change the status of the user story to accepted if no bugs are found, or if the bug is remedied through a bug fix.10. The product owner and IT Lead will review any user stories not passing UAT in time for deployment. They will choose from among the following four options for disposition of those user stories:<ol style="list-style-type: none">a. Deploy the user story to production with the bug intact because the bug does not preclude using the feature;b. Not deploy the user story and defer it to the following sprint;c. Not deploy the user story and defer it to a hotfix;d. Cancel the user story completely.11. The product owner will provide a written go order to the IT Lead to authorize deployment to production. The go order will stipulate any user stories that will be held out of the deployment due to unresolved bugs.12. The IT Lead will migrate the approved user stories into the production environment.13. The IT Lead, or designee, will supply the release notes to the application's users at the time of deployment of a new release. <p>Procedure F: Service Request</p> <ol style="list-style-type: none">1. A user identifies a bug, outage, or other issue in the application.
--	--

	<ol style="list-style-type: none"> 2. The user will create a support ticket, detailing the issue in as minute detail as they can including, if possible, a screenshot of the error. If the user does not have permission in the application to create a support ticket, he/she will engage a supervisor or colleague with those privileges to create the support ticket. 3. The Tier 1 Support Team will triage the issue, addressing any problems in their scope of duties, escalating all others to the Tier 2 Support Team. 4. The Tier 2 Team will initiate discovery on the issue, attempting to recreate the error in a non-production environment. 5. If the root cause of the issue is user error, the Tier 2 Team will provide technical assistance to the user submitting the ticket. 6. If the issue is not user error, and is reparable in real time, the Tier 2 Team will initiate a bug fix. 7. If the issue will require an enhancement, the Tier 2 Team will create a user story in the DevOps solution referencing the support ticket number. 8. The Product Owner will coordinate with the IT Lead to assign a priority to the new user story (e.g. the upcoming sprint, or a hotfix).
<p>VII. Contacts</p>	<p>Chief Information Technology Officer</p> <p>Data and Applications Program Manager</p>
<p>VIII. Related Documents, Forms and Tools</p>	<p>DC Health IT DevOps solution</p> <p>Master Portfolio Listing</p>