



State of New Hampshire
Department of Safety
Division of Emergency Services and Communications

Addendum to RFP 2015-170 "Next Generation 911 (NG9-1-1)"
Advertised: February 11, 2015

Official RFP Link: [RFP DOS 2015-170](http://das.nh.gov/purchasing/specRFP.asp?rfpID=9420)
<http://das.nh.gov/purchasing/specRFP.asp?rfpID=9420>

Date of Addendum: 3/19/15

Designator of Addendum: Addendum E

RE: NH Department of Information Technology (DoIT) Standards and Requirements

The following are State of New Hampshire Department of Information Technology standards and requirements.

APPENDIX G-1 Application Security

IT Security involves all functions pertaining to the securing of State Data and systems through the creation and definition of security policies, procedures and controls covering such areas as identification, authentication and non-repudiation.

This shall include but is not limited to

- Develop software applications based on industry best practices and incorporating information security throughout the software development life cycle
- Perform a Code review prior to release of the application to the State to move it into production. The code review may be done in a manner mutually agreeable to the <VENDOR> and the State. Copies of the final, remediated results shall be provided to the State for review and audit purposes
- Follow change control process and procedures relative to release of code
- Develop applications following security-coding guidelines as set forth by organizations such as, but not limited to Open Web Application Security Project (OWASP) Top 10, SANS Common Weakness Enumeration (CWE) Top 25 or CERT Secure Coding.

- Make available to the for review and audit purposes all software development processes and require training for application developers on secure coding techniques.

APPENDIX G-2 TESTING REQUIREMENTS

All testing and acceptance addressed herein shall apply to testing the System. This shall include planning, test scenario development, Data, and System preparation for testing, and execution of unit testing, System integration testing, conversion/migration testing, installation testing, performance, and stress testing, Security review and testing, and support of the State during user Acceptance Testing (UAT).

G-2.1 Test Planning and Preparation

The overall Test Plan will guide all testing. The Vendor provided, State approved, Test Plan will include, at a minimum, identification, preparation, and Documentation of planned testing, a requirements traceability matrix, test variants, test scenarios, test cases, test scripts, test Data, test phases, unit tests, expected results, and a tracking method for reporting actual versus expected results as well as all errors and problems identified during test execution.

It is crucial that client training and testing activities not be abbreviated in order to meet Project Implementation Schedules. Therefore, the State requires that the testing activities be represented both in terms of effort and duration.

Vendors must disclose in their proposals the scheduling assumptions used in regard to the Client resource efforts during testing.

State testing will commence upon the Vendor Project Manager's certification, in writing, that the Vendor's own staff has successfully executed all prerequisite Vendor testing, along with reporting the actual testing results, prior to the start of any testing executed by State staff.

The State will commence its testing within five (5) business days of receiving Certification from the Vendor that the State's personnel have been trained and the System is installed, configured, complete, and ready for State testing. The testing will be conducted by the State in an environment independent from the Vendor's development environment. The Vendor must assist the State with testing in accordance with the Test Plan and the Work Plan, utilizing test and live Data to validate reports, and conduct stress and performance testing, at no additional cost.

G-2.2 Testing

Testing begins upon completion of the Software configuration as required and user training according to the Work Plan. Testing ends upon issuance of a letter of UAT Acceptance by the State.

Vendor must demonstrate that their testing methodology can be integrated with the State standard methodology.

<p>Unit Testing</p>	<p>Application components are tested on an individual basis to verify that the inputs, outputs, and processing logic of each application component functions without errors. Unit Testing is performed in either the development environment or a testing environment.</p> <p>The goal is to find errors in the smallest unit of Software. If successful, subsequent integration testing should only reveal errors related to the integration between application components.</p>
<p>System Integration Testing</p>	<p>a.) Validates the integration between the individual unit application components and verifies that the new System meets defined requirements and supports execution of interfaces and business processes. The Systems Integration Test is performed in a test environment.</p> <p>b.) Emphasizes end-to-end business processes, and the flow of information across applications. It includes all key business processes and interfaces' being implemented, confirms data transfers with external parties, and includes the transmission or printing of all electronic and paper documents.</p> <p>c.) The State will conduct System Integration Testing, utilizing scripts developed, as identified in the Test Plan, to validate the functionality of the System and its interfaces. The State will also use System Integration Testing to validate modifications, fixes and other System interactions with the Vendor supplied Software Solution.</p>
<p>Conversion /Migration Validation Testing</p>	<p>The Conversion/Migration Validation Testing should replicate the entire flow of the converted data through the Software Solution. As the Software Solution is interfaced to legacy or third-party applications, the testing verifies that the resulting converted legacy data performs correctly.</p>
<p>Installation Testing</p>	<p>Application components are installed in the System test environment to test the installation routines and are refined for the eventual production environment. This activity serves as a dry run of the installation steps in preparation for configuring the production System.</p>
<p>User Acceptance Testing (UAT)</p>	<p>The User Acceptance Test (UAT) is a verification process performed in a copy of the production environment. The User Acceptance Test verifies System functionality against predefined Acceptance criteria that support the successful execution of approved business processes.</p> <p>a.) The Vendor's Project Manager must certify in writing, that the Vendor's own staff has successfully executed all prerequisite Vendor testing, along with reporting the actual testing results prior to the start of any testing executed by State staff.</p>

	<p>b.) The State will be presented with a State approved Test Plan, test scenarios, test cases, test scripts, test data, and expected results, as well as written Certification of the Vendor's having completed the prerequisite tests, prior to the State staff involvement in any testing activities</p> <p>c.) UAT will also serve as a performance and stress test of the System. It may cover any aspect of the new System, including administrative procedures such as backup and recovery. The results of the UAT provide evidence that the new System meets the User Acceptance criteria as defined in the Work Plan.</p> <p>d.) Upon successful conclusion of UAT and successful System deployment, the State will issue a letter of UAT Acceptance and the respective Warranty Period shall commence as described in Section H-25.10.1: Warranty Period.</p>
<p>Performance Tuning and Stress Testing</p>	<p>Vendor shall develop and document hardware and software configuration and tuning of System infrastructure as well as assist and direct the State's System Administrators and Database Administrators in configuring and tuning the infrastructure to support the software throughout the project</p> <p>Performance Tuning and Stress Testing</p> <p><u>Scope</u></p> <p>The scope of performance testing shall measure the system level metrics critical for the development of the applications infrastructure and operation of the applications in the production environment. It will include the measurement of response rates of the application for end-user transactions and resource utilization (of various servers and network) under various load conditions. These response rates shall become the basis for changes and retesting until optimum system performance is achieved.</p> <p>The application transactions shall be identified with specific roles and selected transactions shall be recorded for the performance measurements. These will be compared to baselines to determine if object and/or system performance increases as changes are made.</p> <p>Performance testing shall consider the full scope of the application infrastructure with emphasis on the most heavily used or shared transactions. Performance testing of the application will profile the identified user transactions and assist in the identifying performance gaps to improve the most critical parts of the applications.</p> <p>Performance testing and tuning shall occur in the final production environment and shall use a copy of the final production database to provide the best results.</p> <p>Vendor must lead this effort. Responsibilities include identifying appropriate tunable parameters and their default and recommended settings, developing scripts, which accurately reflect business load and</p>

coordinating reporting of results.

Test types

Performance testing shall use two different types of tests to determine the stability of the application. They are baseline tests and load tests

Baseline Tests: Baseline tests shall collect performance data and load analysis by running scripts where the output is broken down into business transactions or functions. The test is like a single user executing a defined business transaction. During baseline testing, each individual script is run to establish a baseline for transaction response time, throughput and other user-based metrics. Usually each business transaction is executed multiple times during a single test run to obtain an average for the user-based metrics required for the performance testing evaluations. It must be noted that changes made to the code after baseline testing is completed will skew the results collected to date. All effort will be made to provide a code test base that is tested in the environment for problems prior to the establishment of the baseline, which are used in future testing and tuning efforts. Any changes introduced into the environment after performance testing has started can compromise the accuracy of the results and will force a decision to be made whether baseline results need to be recreated.

Load Tests: Load testing will determine if the behavior of a system can be sustained over a long period of time while running under expected conditions. Load tests helps to verify the ability of the application environment under different load conditions based on workload distribution. System response time and utilization is measured and recorded.

Tuning

Tuning will occur during both the development of the application and load testing. Tuning is the process whereby the application performance is maximized. This can be the result of making code more efficient during development as well as making tuning parameter changes to the environment.

For infrastructure tuning, parameters will be identified for all components prior to undertaking the load testing efforts. This should include a list of the variables, their definitions, the default settings, range of acceptable settings and the settings as testing begins. This will permit the team to identify the areas of most potential gain and a starting point. Tuning is a process which is repeated until the team feels that the systems are running at or near optimum performance.

Implementing Performance and Stress Test

Performance and Stress test Tools must be provided by the Vendor for this effort. Consideration must be give to licensing with respect to continued use for regression testing. If the Vendor is familiar with open source low/no

cost tools for this purpose those tools should be identified in your response.

Scheduling Performance and Stress Testing

Vendor shall perform test planning. The steps for planning include identification of application functionality as well as what percentage of normal daily use is represented by each function. This information will become the foundation for scripting so that tests closely represent what loads in production will look like.

Vendor shall provide definition and expectations from testing. This definition should include who is in charge of testing and coordinating results, anticipated run times, logs required for tracking, their locations and which technician is responsible to track and provide them following each test to the team.

Initial test runs shall be completed to establish that the tests and data sets can be run to completion without errors. The ratio of types of transactions which makeup the test shall be reviewed prior to the beginning of testing and then again once testing has begun to make sure that testing accurately reflects the system performing in production.

Initial tests shall be used to establish a baseline from which all subsequent tests will be compared. Tests will be considered for baseline status once two of them have been run within 2% of each other in key and overall performance areas. No changes to the test scripts or data sets (with the exception of restores after each test) can be done to the test environment once tuning has begun so as to not damage the comparison to baseline results. The systems must be restarted prior to each test run to assure all cache is cleaned out. All effort will be made to run these tests at a time when system and network infrastructure utilization doesn't impact the results. Tests will be run in close proximity to our infrastructure to eliminate the public network from our environment.

Post-test reporting and result assessment will be scheduled following each test. The team will compare these results to the baseline and a determination must be made to make additional changes to the parameter being tuned or return to the prior configuration and select another parameter to tune while keeping in mind that significant changes to any one parameter may require the retesting of some others. Careful work on identifying dependencies up front should minimize this impact.

If defects are identified in the application during testing, they will be recorded; however, changes to the application code should be avoided if possible so as not to affect baseline comparisons. If a change to the application is required new baselines will be established (and possibly the execution of prior tests to validate changes with the new application) before testing can continue.

When performing capacity testing against a GUI the focus will be on the

	<p>ability of the interface to respond to user input.</p> <p>During stress/load testing the tester will attempt to stress or load an aspect of the system to the point of failure. The goal being to determine weak points in the system architecture. The tester will identify peak load conditions at which the program will fail to handle required processing loads within required time spans.</p> <p>During Performance testing the tester will design test case scenarios to determine if the system meets the stated performance criteria (i.e. A Login request shall be responded to in 1 second or less under a typical daily load of 1000 requests per minute.). In both cases, the tester will determine the capacity of the system under a known set of conditions.</p>
<p>Regression Testing</p>	<p>As a result, of the user testing activities, problems will be identified that require correction. The State will notify the Vendor of the nature of the testing failures in writing. The Vendor will be required to perform additional testing activities in response to State and/or user problems identified from the testing results.</p> <p>Regression testing means selective re-testing to detect faults introduced during the modification effort, both to verify that the modifications have not caused unintended adverse effects, and to verify that the modified and related (possibly affected) System components still meet their specified requirements.</p> <p>a.) For each minor failure of an Acceptance Test, the Acceptance Period shall be extended by corresponding time defined in the Test Plan.</p> <p>b.) The Vendor shall notify the State no later than five (5) business days from the Vendor's receipt of written notice of the test failure when the Vendor expects the corrections to be completed and ready for retesting by the State. The Vendor will have up to five (5) business days to make corrections to the problem unless specifically extended in writing by the State.</p> <p>c.) When a programming change is made in response to a problem identified during user testing, a regression Test Plan should be developed by the Vendor based on the understanding of the program and the change being made to the program. The Test Plan has two objectives:</p> <ol style="list-style-type: none"> 1. validate that the change/update has been properly incorporated into the program; and 2. validate that there has been no unintended change to the other portions of the program. <p>d.) The Vendor will be expected to:</p> <ol style="list-style-type: none"> 1. Create a set of test conditions, test cases, and test data that will validate that the change has been incorporated correctly; 2. Create a set of test conditions, test cases, and test data that will validate that the unchanged portions of the program still operate correctly; and 3. Manage the entire cyclic process.

e.) The Vendor will be expected to execute the regression test, provide actual testing results, and certify its completion in writing to the State prior to passing the modified Software application to the users for retesting.

In designing and conducting such regression testing, the Vendor will be required to assess the risks inherent to the modification being implemented and weigh those risks against the time and effort required for conducting the regression tests. In other words, the Vendor will be expected to design and conduct regression tests that will identify any unintended consequences of the modification while taking into account Schedule and economic considerations.

In their Proposals Vendors must acknowledge their responsibilities for regression testing as described in this section.

Security Review and Testing

IT Security involves all functions pertaining to the securing of State Data and Systems through the creation and definition of security policies, procedures and controls covering such areas as identification, authentication and non-repudiation.

All components of the Software shall be reviewed and tested to ensure they protect the State's hardware and software and its related Data assets.

Service Component	Defines the set of capabilities that:
Identification and Authentication	Supports obtaining information about those parties attempting to log onto a system or application for security purposes and the validation of users
Access Control	Supports the management of permissions for logging onto a computer or network
Encryption	Supports the encoding of data for security purposes
Intrusion Detection	Supports the detection of illegal entrance into a computer system
Verification	Supports the confirmation of authority to enter a computer system, application or network
Digital Signature	Guarantees the unaltered state of a file
User Management	Supports the administration of computer, application and network accounts within an organization.
Role/Privilege Management	Supports the granting of abilities to users or groups of users of a computer, application or network
Audit Trail Capture and Analysis	Supports the identification and monitoring of activities within an application or system
Input Validation	Ensures the application is protected from buffer overflow, cross-site scripting, SQL injection, and unauthorized access of files

and/or directories on the server.

In their proposal, the Vendors must acknowledge their responsibilities for security testing. Tests shall focus on the technical, administrative and physical security controls that have been designed into the System architecture in order to provide the necessary confidentiality, integrity and availability. Tests shall, at a minimum, cover each of the service components. Test procedures shall include Penetration Testing (pen test) or code analysis and review. Prior to the System being moved into production, the Vendor shall provide results of all security testing to the DESC IT staff for review and acceptance. All Software and hardware shall be free of malicious code (malware).

Penetration Testing shall include:

11.3 Implement a methodology for penetration testing that includes the following:

- Is based on industry-accepted penetration testing approaches (for example, NIST SP800-115)
- Includes coverage for the entire CDE perimeter and critical systems
- Includes testing from both inside and outside the network
- Includes testing to validate any segmentation and scope-reduction controls
- Defines application-layer penetration tests
- Defines network-layer penetration tests to include components that support network functions as well as operating systems
- Includes review and consideration of threats and vulnerabilities experienced in the last 12 months
- Specifies retention of penetration testing results and remediation activities results.

11.3.1 Perform *external* penetration testing at least annually and after any significant infrastructure or application upgrade or modification (such as an operating system upgrade, a sub-network added to the environment, or a web server added to the environment).

11.3.2 Perform *internal* penetration testing at least annually and after any significant infrastructure or application upgrade or modification (such as an operating system upgrade, a sub-network added to the environment, or a web server added to the environment).

11.3.3 Exploitable vulnerabilities found during penetration testing are corrected and testing is repeated to verify the corrections.

11.3.4 If segmentation is used to isolate the CDE from other networks, perform penetration tests at least annually and after any changes to segmentation controls/methods to verify that the segmentation methods are operational and effective, and isolate all out-of-scope systems from in-scope systems.

If you have any additional questions, please feel free to contact:

Robert Brown, IT Manager
33 Hazen Drive
Concord, NH 03305
(603) 271-6911
rbrown@e911.nh.gov



Bruce G. Cheney, Director

3/19/15
Date