एन एम डी सी
NMDC

# SOFTWARE CONTROL POLICY



| | | | | | |
|---|---|---|---|---|---|
| **Author** | | **Document name** | | **Date of first issue** | |
| **Owner** | C & IT Department | **Document ref. no.** | | **Date of latest re-issue** | |
| **Version** | 1.1 | **Page** | 1 of 7 | **Date of next review** | |
| **Issue Status** | Under Review/ Live | **Security classification** | Internal use only | **Reviewer** | |

एन एम डी सी
NMDC

# SOFTWARE CONTROL POLICY

**VERSION CONTROL**

| Revision no. | Date of issue | Prepared by | Reviewed by | Approved by | Issued by | Remarks |
|---|---|---|---|---|---|---|
| | | | | | | |
| | | | | | | |
| | | | | | | |

एन एम डी सी
NMDC

# SOFTWARE CONTROL POLICY

## OBJECTIVE

Software, being a critical resource, should be maintained and supported systematically during its lifetime. The company has defined the procedures for acquisition, design and development as well as maintenance of software in a standardized manner. These procedures and methods delineate the various aspects of specifications, configuration, development and procurement cycle of the software while ensuring that they are of required quality with appropriate controls built in them and that they meet the desired business objectives.

## SCOPE

This domain addresses policies and procedures related to the software life cycle management of all software resources. This policy applies to all the NMDC staff, information resources as well as the application and systems software.

## RESPONSIBILITY

For all types of software managed within the company, the respective Systems Executives and the Department Heads along with the Managers are responsible for decision making, initiation, execution and monitoring of various procedures included in this policy.

## POLICY RULES

**Commented [BA1]:** Processes and necessary documentation to be done by the software developer team. If outsourced, these clauses should be included in the contract as part fo activities to be carried out.

### Software Acquisition

**1. Acquisition planning phase:**

This phase defines NMDC's strategy, initiating planning process, and establishing general practices for planning of software acquisition. The objectives behind acquiring the software are identified, critically reviewed and finalized.

a) A cost benefit analysis needs to be done to determine if the application can\should be developed in-house or should be outsourced
b) The final objectives are then translated into procurement strategy, which essentially involves deciding on the type of the software to be acquired, i.e. commercial off-the shelf or customized application.
c) A software acquisition process suitable for the company has been established and documented. It includes contracting methods, administrative responsibilities, legalities involved and assignment of Systems Executives for these jobs. These technical services are hired in case of non-availability of such skills within the company.
d) The functionalities that the software requires are to be defined, analyzed and frozen. This would also involve minimum, but critical functional specifications that a new system must address and provide for the yardsticks for a specific software quality. These requirements are then translated into formal 'Request for proposals' (RFPs).
e) A verification of other hardware and software support will be verified by the C&IT department before realizing the contract

# SOFTWARE CONTROL POLICY

**2. Contracting phase:**

a) The C&IT Department and the Security Administrator of the application identify prospective vendors, who provide documentation of the software, demonstrate the functionality features and offer formal proposals.
b) The supplier data along with credentials and past performance is reviewed.
c) Contract requirements detailing expected quality, acceptable performance and criteria, contract provisions, payments options and their direct linkage to deliverables are prepared.
d) The contracts are finalized after obtaining the concurrence from legal department.
e) Once the proposals are received, they are to be evaluated against the desired and critical functionality features, quality specifications, payment terms, future maintenance service and support from vendors.
f) At the end of this stage, the supplier, the quotation and other terms are finalized.

**3. Implementation phase:**

This is the phase of managing the contract and the project during software implementation and monitoring the progress of the project from time to time. The performance of the supplier is benchmarked against the time scale and deliverables mutually agreed with the vendors. Management should create an environment within the company, which would accelerate the process of implementation. Any deviation from the implementation plan is to be reported to the HoD and Manager in-charge of monitoring this activity.

**4. Software acceptance phase:**

This phase involves the testing and evaluation of the software, maintaining controls over the testing process and establishing the acceptance process. The testing is divided into systems level test and user acceptance testing (UAT). Some of the users identified for UAT should include the members involved in the planning stage.

a) A comprehensive test plan is to be prepared well in advance considering all critical requirements and quality benchmarks. It is important that the test plans are based on most current performance standards and benchmarks agreed upon with the supplier.
b) The test process is to be documented along with test plans and essentially includes the processes for detecting the differences between the existing and required conditions and evaluating the system features such as portability, performance, speed, functionality, security, data integrity and compatibility.
c) Final user acceptance tests are done only after field level testing to verify performance, quality and controls within the system. End users are required to carry out testing.
d) The test plans, the process and test results are documented in detail and the final sign-off from users is obtained.
e) The final payment is not to be made to the vendors until it is certified that all the software deliverables meet the contract specifications and that all acceptance criteria are satisfied.

**5. Using and distribution of the software:**

एन एम डी सी
NMDC

# SOFTWARE CONTROL POLICY

After the approval of Security manager and Department head, the distribution of the software to the other locations would be done. This is the phase when the software is implemented and used in live environment. An effort is made to identify both positive and negative aspects of the acquisition process particularly deviations, lacunas and lessons learnt. The corrective action taken is documented.

Along with the assessment of user satisfaction, an estimate is made to determine the maintenance efforts after putting the software in use. A suitable Annual Maintenance Contract (AMC) is signed with the vendor. While evaluating the supplier performance, the results are maintained for future reference.

## Security of system files

### 1. Control of operational software

To minimize the risk of corruption of operational systems, the following controls are considered.

a) The nominated Systems Executive only updates the operational program libraries after appropriate authorization from the System Administrator of the respective application and the respective Departmental Manager.
b) Executable code is not to be implemented on an operational system until evidence of successful testing and user acceptance is obtained, and the corresponding program source libraries are updated.
c) An audit log is maintained of all updates to operational program libraries.
d) Previous version of software is retained as a contingency measure.
e) Vendor supplied software used in operational systems is maintained at a level supported by the supplier. Any decision to upgrade to a new release will consider the security of the release, i.e. the introduction of new security functionality or the number and severity of security problems affecting this version. Software patches are applied when they can help to remove or reduce security weaknesses.

### 2. Protection of system test data

Test data is protected and controlled. System and acceptance testing usually requires substantial volumes of test data that are as close as possible to operational data. The use of operational databases containing personal information is to be avoided. If such information is used, it is depersonalized before use. The following controls are applied to protect operational data, when used for testing purposes:

a) The access control procedures, which apply to operational application systems, also apply to test application systems.
b) There is separate authorization each time operational information is copied to a test application system.
c) Operational information is erased from a test application system immediately after the testing is complete.
d) The copying and use of operational information is logged to provide an audit trail.

### 3. Access control to program source library

एन एम डी सी
NMDC

# SOFTWARE CONTROL POLICY

In order to reduce the potential for corruption of computer programs; strict control is maintained over access to program source libraries, which are as follows:

a) Where possible, program source libraries shall not be stored on operational systems.
b) Systems Executives responsible for the application will approve accesses to the program source library of the application.
c) IT support team members do not have unrestricted access to program source libraries
d) The updating of program source libraries and the issuing of program sources to programmers is to be performed only by the nominated Systems Executives upon authorization from Security Administrator for that application.
e) Program listings are held in a secure environment.
f) An audit log is maintained of all accesses to program source libraries.
g) Old versions of source programs are archived, with a clear indication of the precise dates and times when they were operational, together with all supporting software, job control, data definitions and procedures.
h) Maintenance and copying of program source libraries is subject to strict change control procedures

## Security in Development and Support Processes

### 1. Concept Proposal

The Concept Proposal is the first document to be completed in the Systems Development Life Cycle (SDLC). The purpose is to highlight where strategic goals are not being met or where mission performance needs to be improved. The Project Manager prepares the Concept Proposal for NMDC. It should include:

a) Originator
b) Description of investment
c) Mission/Goal of investment
d) Existing Structure
e) Benefits
f) Funding

### 2. Conversion Plan

The Conversion Plan describes the strategies involved in converting data from an existing system to another hardware or software environment. It is appropriate to re-examine the original system's functional requirements for the condition of the system before conversion to determine if the original requirements are still valid. This would also be applicable for the acquisition stage.

Following strategies may be used for conversion of system hardware, software, and data.

a) **Hardware Conversion Strategy:** This section describes the strategy to be used for the conversion of system hardware, if any. Describe the new (target) hardware environment, if appropriate.
b) **Software Conversion Strategy:** Any conversions to any existing software which may be required.
c) **Data Conversion Strategy:** This section describes the data conversion strategy, data quality assurance, and the data conversion controls.

d) **Data Conversion Approach:** Any specific data preparation requirements and the data that must be available for the system conversion. If data will be transported from the original system, provide a detailed description of the data handling, conversion, and loading procedures. If these data will be transported using machine-readable media, describe the characteristics of those media.

e) **Interfaces:** In the case of a hardware platform conversion the interfaces to other systems may need reengineering. This section will describe the affected interfaces and the revisions required in each.

f) **Data Quality Assurance and Control:** The strategy to be used to ensure data quality before and after all data conversions. This section will also describe the approach to data scrubbing and quality assessment of data before they are moved to the new or converted system. The strategy and approach may be described in a formal transition plan or a document if more appropriate.

g) **Conversion Risk Factors:** This section describes the major risk factors in the conversion effort and strategies for their control or reduction. Descriptions of the risk factors that could affect the conversion feasibility, the technical performance of the converted system, the conversion schedule, or costs should be included. In addition, a review should be made to ensure that the current backup and recovery procedures are adequate as well as operational.

### 3. Functional requirements study

The functional requirements document (FRD) is a formal statement of an application's functional requirements. It serves the same purpose as a contract. The developers agree to provide the capabilities specified. NMDC agrees to find the product satisfactory if it provides the capabilities specified in the FRD.

The functional requirements describe the core functionality of the application. This section includes the data and functional process requirements. The sub stages of the following stages would be

**a)** **Data Requirements:** Describe the data requirements by producing a logical data model, which consists of entity relationship diagrams, entity definitions, and attribute definitions. This is called the application data model. The data requirements describe the business data needed by the application system. Data requirements do not describe the physical database.

**b)** **Functional Process Requirements:** Process requirements describe what the application must do. Process requirements relate the entities and attributes from the data requirements to the users' needs. Process requirements may be expressed using data flow diagrams, text, or any technique that provides the following information about the processes performed by the application:

- Context
- Detailed view of the processes
- Data (attributes) input to and output from processes
- Logic used inside the processes to manipulate data
- Accesses to stored data
- Processes decomposed into finer levels of detail

**c)** **Security:** The security section describes the need to control access to the data. This includes controlling who may view and alter application data. State the consequences of the following breaches of security in the subject application:

- Erasure of contamination of application data

— Disclosure of NMDC`s secrets
— Disclosure of privileged information about individuals

It will be ensured that the compatibility to the LDAP server and strong Authentication server exists in the security requirements.

**d) Audit Trail:** List the activities that will be recorded in the application's audit trail. For each activity, list the data to be recorded.

**e) Recoverability:** Recoverability is the ability to restore function and data in the event of a failure. Answer the following questions in this section:

— In the event the application is unavailable to users (down) because of a system failure, how soon after the failure is detected must function be restored?
— In the event the database is corrupted, to what level of currency must it be restored? For example "The database must be capable of being restored to its condition on no more than one hour before the corruption occurred."
— If the process site (hardware, data, and onsite backup) is destroyed how soon must the application be able to be restored?

## 4. Implementation

Refer to the Implementation section

## 5. Separation of Development and Operational Facilities

Development and test activities may cause problems, e.g. unwanted modification of files or system environment, or of system failure. The level of separation that is necessary, between operational, test and development environments, to prevent operational problems needs to be considered. Development and testing activities may cause unintended changes to software and information if they share the same computing environment. Separating development, test and operational facilities is essential to reduce the risk of accidental change or unauthorized access to operational software and business data. The following controls are considered:

a) Development and operational versions of the software run on different computer processors, or in different domains or directories.
b) Development and testing activities are separated.
c) Compilers, editors and other system utilities are not accessible from operational systems.

Different log-on procedures are used for operational and test systems, to reduce the risk of error. Users are encouraged to use different passwords for these systems, and menus should display appropriate identification messages.

Development staff has access to operational passwords where controls are in place for issuing passwords for the support of operational systems. Controls are in place to ensure that such passwords are changed after use.

## 6. Change control procedures

एन एम डी सी
NMDC

# SOFTWARE CONTROL POLICY

Formal change control procedures are to be followed. Modifications to software packages are discouraged. As far as possible, and practicable, vendor-supplied software packages are used without modification. Where it is deemed essential to modify a software package, the following points are considered:

a) The risk of built-in controls and integrity processes being compromised.
b) Whether the consent of the vendor is obtained. Alternatively, obtaining formal approval for detailed proposals before work commences
c) The possibility of obtaining the required changes from the vendor as standard program updates.
d) The impact if the company becomes responsible for the future maintenance of the software as a result of changes. If changes are deemed essential the original software is retained and the changes applied to a clearly identified copy. All changes are fully tested and documented, so that they can be reapplied if necessary to future software upgrades.
e) Identifying all computer software, information, database entities and hardware that require amendment.
f) Ensuring that the authorized user accepts changes prior to any implementation.
g) Ensuring that implementation is carried out with minimal or no business disruption.
h) Maintaining a version control for all software updates.
i) Maintaining an audit trail of all change requests.

Systems Executives responsible for each application ensure that security and control procedures are not compromised, that support team members are given access only to those parts of the system necessary for their work, and that formal agreement and approval for any change is obtained. Changing application software can impact the operational environment.

### 7. Trojan code

Trojan code is designed to affect a system in a way that is not authorized and not readily noticed and not required by the recipient or user of the program. Where Trojan code is a concern, the following controls are to be considered:

a) Buying programs only from a reputable source.
b) Buying programs with source code so the code may be verified.
c) Using evaluated products.
d) Inspecting all source code before operational use, wherever possible.
e) Controlling access to, and modification of, code once installed.
f) Use staff of proven trust to work on key systems.

### 8. Outsourced Software Development

Where software development is outsourced, the following points are considered:

a) Selection of vendor based on software quality standards such as SEI-CMM or ISO standards.
b) Licensing arrangements, code ownership and intellectual property rights.
c) Certification of the quality and accuracy of the work carried out.
d) Escrow arrangements in the event of failure of the software developer.
e) Rights of access for audit of the quality and accuracy of work done.

f) Contractual requirements for quality of code.
g) Testing before installation to detect Trojan code.

## 9. Software Maintenance process

Following are the various stages of managing and executing the software development activities:

a) **Identification of problem:** This is the main source of input and initiator to the maintenance process. Each software Change Request (CR) is classified as 'Accepted' or 'Rejected'. In this phase, software modifications are identified, classified and assigned an initial priority ranking. The maintenance approach is classified into corrective, adaptive, preventive and emergency with preliminary estimate of efforts required. The CR entries are assigned a unique number and maintained in a repository by the respective Systems Executives.
b) **Analysis of request:** The validated and approved CR mentioned above along with the system and project documentation are used to study the feasibility and scope of modifications to the software. At the end of this stage, a preliminary plan for design, implementation, test and rollout is prepared. This essentially includes various parameters such as impact of modifications, alternate solutions, and workarounds and their prototyping, analysis of conversion requirements and coding efforts, safety, security and controls implications, human resource, short-term and long-term costs and the net value derived out of the modification to the software.
c) **Solution design stage:** In this phase, the current system and project documentation, existing software and databases along with the output of above phases are used for designing the modifications to the system. This stage involves identifying affected modules and other software, modifying system documentation as well as data flow diagram, file relationships, database dictionary, workflow charts etc. and creating test cases for the modified software. The test involves the plans for user acceptance as well as system regression testing. The user and system manuals, which are required to be changed for these modifications, are identified at this stage.
d) **Implementation:** In this phase, the results of the design phase, the current source code, project and system documentation as modified with latest changes are used to drive the implementation efforts. At this stage, the unit testing and integration testing is conducted as per the test plans. The test process and results are documented to state functional, performance, and usability and control effects of the modification. During the implementation, risk analysis and review is to be conducted regularly rather than at the end. Regression and system testing is carried out to ensure that modified code does not introduce faults that did not exist prior to maintenance activity. Acceptance tests are carried out on fully integrated systems and the users of the software conduct these. They conduct the software functionality or interoperability audit.
e) **Roll-out/commissioning the system:** This involves installing the new programs on live environment, notifying the users about changes provide training and hand over version release document. The old programs are archived and entire programs and database files are backed up before installing the programs. The user sign-off is obtained after successfully implementing the changes.

## 10. Precautions with Software Upgrades

**Commented [BA2]:** Can be done through MDM.

There are various instances where software on systems requires special attention. Some of them are as below:

एन एम डी सी
NMDC

# SOFTWARE CONTROL POLICY

a) Software Patches that might have to be implemented from time to time are tested prior to being put into the live environment. Similarly, hot fixes are also tested and only those necessary are applied to the systems.
b) Version upgrades are preferably carried out by the vendor, who has been identified as having the expertise, after signing escrow and implementation agreements with him.
c) User training is carried out in order to make them aware of the upgraded software and the patches/hotfixes applied. It is the responsibility of the Systems Executives to ensure that users are made aware of the changes in their respective environments.

## User Related

1. All software upgrades, patches, hot fixes and new software as identified by NMDC in Annexure 1 are pushed to the user systems by the C&IT Department.
2. Users cannot edit/uninstall any software listed in Annexure 1.
3. Users can only install software from the MDM software app/google play/apple app store/
4. Users will require permission from RO1 for installing any other software from any other source. The software installation will be subject to checking by the anti-virus installed in the system.

## ANNEXURE 1:

The company will maintain a list of required software depending on asset type. This list will be updated based on business requirements of NMDC

| Asset Type | Software to be installed |
|---|---|
| Laptop | |
| | |
| | |
| | |
| Desktop | |
| | |
| | |
| | |
| Smart Phone (BYOD) | |
| | |
| | |
| | |