



Rancho Los Amigos National Rehabilitation Center

INFORMATION MANAGEMENT SERVICES

POLICY AND PROCEDURE

SUBJECT: APPLICATION DEVELOPMENT
PROCEDURES

Policy No.: 310
Supersedes: 12/9/91
Revision Date: July 2000
Page: 1 of 4

I. PURPOSE:

To establish a specific protocol for developing and modifying information management systems for the purpose of improving patient outcomes and individual and hospital performance in patient care, governance, management and support processes.

II. POLICY:

A standardized set of procedures is used by Information Management Services (IMS) for information management system development and system modifications.

III. PROCEDURES:

1. **Information Systems Project Initiation.** All projects are subject to the information systems project initiation processes and approval by the Executive Council, as outlined in policy A133
2. **Project Review.** The Director, Data Administration reviews all project requests from the project initiation process before resources are allocated to the project. The Director, Data Administration and Manager, Systems Development/Integration review the impact of the project on the IMS infrastructure.
3. **Assign Project Leader.** The initiating party of the project should assign a Project Leader to coordinate and lead the development of the project. IMS will assign the appropriate developer for the programming of the application.
4. **Identify Key Members.** It is the responsibility of the Project Leader to assemble the principle members for a project team. This team should consist of the requestor(s), end-users, management, programmer/developer, and any persons offering valuable input into the design of the system.
5. **Initial Design Meeting.** The Project Leader will assemble the necessary key members of the project team to review the request and supporting documentation. It will be the responsibility of the Project Leader to direct the meeting. There may be additional meetings, as needed throughout the design and development cycle of the system. The initial discussion will address, but not be limited to the following areas of concern:
 1. Goals – The request is reviewed to establish expectations and goals. Identify alternative solutions to the objectives, if necessary.
 2. System Requirements – Determine hardware and software requirements (e.g., multi-user versus stand-alone, storage space, printers, application software, communication devices, cabling,

interfaces, performance requirements, etc.). After establishing the proposed system requirements, determine additional equipment that need to be procured by the requestor.

3. Define Input – Determine where the data to be input into the system will originate. Data may come from various sources such as forms, downloads, diskettes, tapes, etc.
4. Evaluate Current Systems – Review any manual or automated systems that are used currently by the requestors that are to be replaced or enhanced by the request.
5. Define Output – Establish what output will be expected of the system, e.g., current and proposed reports, forms, electronic transfers, interfaces, etc.
6. Logic – Ascertain various ways in which the data will need to be manipulated to achieve the required output.
7. Users – Determine which users will be included in the testing of the prototype system.

5. System Design.

1. Identify Unique Data Elements – Separate input data into the individual elements. Determine key fields and the necessary indexes. Determine the types and lengths of the data elements, e.g., alpha, numeric, etc.
2. Establish Rules – Based on user requirements and restrictions, determine rules associated with individual data elements to ensure system integrity, e.g., checking if a value is valid, mandatory fields, specific input formats, etc.
3. Develop Program Specifications – Develop detailed descriptions of the flow of the system input, output, and processing logic. Conduct a walk through of the external system design with the requestor for accuracy and incorporate necessary changes. Develop flow charts and narrative that detail the logic to a level that will enable the programmer to begin programming directly from the completed program specifications documentation.
4. Security Specifications – Develop necessary security precautions to protect the confidentiality of data.
5. Develop Conversion Plan – If data is to be imported from an existing system, design a methodology for converting the data to be loaded into the new system. Include downtime and impact analysis.
6. Data Modeling – Establish logical relationships between various data elements. Determine how the elements will be arranged in relational tables, as needed.

6. System Development.

1. Tables – Design and test functionality of individual tables by the system.
2. Screens – Design and test the input and output screens.
3. Output – Develop and test the output features of the system, e.g., forms, reports, tapes, diskettes, etc.

-
4. Program Application – Create the necessary programming and menu applications that comprise the system.
 5. Rules – Incorporate data entry rules to ensure the data and system integrity, e.g., mandatory fields, minimum and maximum value ranges, format types, value verification.
 6. Document Application – Begin development of user and technical documentation to assist in training and implementation.
 7. Debug System – Evaluate the system’s ability to achieve the desired results with test data. Test when applicable: the rules, output, interface capabilities, menus, screens, etc.
- 7. System Test.**
1. Review Documentation – Review the application and associated documentation with the Manager System Development/Integration and/or the Director, Data Administration.
 2. Install Prototype – Install a test system for the end-users. Ensure that the necessary equipment is available to perform a thorough evaluation.
 3. Train End-Users – Review the user application guide for clarity and content with delegated end-users. Train the data entry and other necessary personnel on the system operation. This training will consist of such areas as: data input and editing procedures, requesting reports, backing up the system, etc.
 4. Test and Evaluate Results – The test system will be utilized for a specific period of time. Testing will consist of using the initially developed test data. The testing will determine if the operating logic is consistent with the program specifications. The data entry rules and requirements will be tested using intentional erroneous input. Communication, interfaces, security procedures, downloading, and uploading of any input/output processing will be analyzed. Output will be compared with expected and current operations for correctness. Upon completion of testing, findings will be reviewed with the appropriate project team members.
 5. Implement Changes – Incorporate necessary changes as generated from the system test results before the implementation process. Review the changes with the requestor.
- 8. Finalize System Documentation.** User/System Documentation – Incorporate any changes from the test results in the appropriate system documentation. The documentation typically contains user guides, desk instructions, and other similar documents, which will be maintained and kept up-to-date throughout the useful life cycle of the system. Additionally, include any operations documentation such as technical specifications, application code, table structure, etc. Include instructions for any processes that may be needed, but usually not done on a daily basis, include backup procedures, batch processing, etc.
- 9. Requester Acceptance.** Before installing final system, review results of testing and subsequent changes with the requestor.

10. Implementation.

1. Deliver Documentation – Provide users with the finalized version of the system documentation.
2. Convert Data – When applicable, perform the necessary conversion in accordance with the data conversion plan and import the data into the system.
3. Install Equipment – If new hardware and/or software have been ordered, it may be necessary to schedule several activities related to its installation, checkout, acceptance testing, and operator/user familiarization.
4. Training – Primary attention is given to the interface each individual user has with the system including; how to prepare input and access the system, how to read the output produced by the system, and how to correct any error that may occur. Emphasis is placed on any activities they may expect to perform in support of the system such as backing up the application software and data.