|  |  |  |
| --- | --- | --- |
| Document ID  **ITSW105** | Title  **SOFTWARE PROGRAMMING** | Print Date  **mm/dd/yyyy** |
| Revision  **0.0** | Prepared By  **Preparer’s Name / Title** | Date Prepared  **mm/dd/yyyy** |
| Effective Date  **mm/dd/yyyy** | Reviewed By  **Reviewer’s Name / Title** | Date Reviewed  **mm/dd/yyyy** |
|  | Approved By  **Final Approver’s Name / Title** | Date Approved  **mm/dd/yyyy** |

**Policy:** All software products developed by the company must meet company standards for user interface, program structure, system interface, toolset, and configuration.

**Purpose:** To effectively and efficiently carry out the plans of the software designer.

**Scope:** All software products and updates released by the company.

**Responsibilities:**

The Software Designer is responsible for transforming the system requirements developed by the systems analyst into programming instructions and communicating the overall design approach to the team.

The Software Programmer is responsible for reviewing the programming instructions created by the software designer and using those instructions to design each software component.

**Procedure:**

### 1.0 PROGRAMMING STANDARDS

1.1 Members of the software development department should create basic programming standards, to be followed in programming all of the Company’s software. The Company encourages all programmers to participate in setting these standards: welcoming programmers’ input encourages teamwork and often makes programmers more willing to follow the standards.

1.2 Programming standards typically address:

* Conventions in the use of the programming tools;
* A locking strategy that defines how and when record locking is imposed;
* A strategy for handling messages in the various types of software components;
* Programming style guides for the various types of software components; and
* Strategies for dealing with inefficiencies and limits of the programming tools, software environment, or equipment.

1.3 Software Engineering Standards are used very early in the development of the project plan. See procedure ITSW101 PROJECT DEFINITION for additional information.

### 2.0 PROGRAMMING TASKS

* 1. The software designer divides the programming tasks among the members of the programming team and then conducts a series of meetings to introduce the design to each member of the team.
  2. The software designer presents the software design to the entire programming team at an initial “kick-off” meeting. This meeting explains to the programmers the context for each programming task. For a large system, multiple meetings might require several sessions over the course of the programming effort.
  3. The software designer explains each programming task with each programmer who has been assigned to it. This meeting clarifies the specifications to eliminate misunderstandings.

### 3.0 SOFTWARE DEVELOPMENT

3.1 The programmer reviews the programming instructions created by the software designer, and uses these instructions to design each software component.

3.2 If the programming instructions require the programmer to modify existing software components, the programmer checks those components out of the software component library. See ITSW109 SOFTWARE RELEASES AND UPDATES procedure for information about the software component library.

3.3 The programmer programs the software components. After completing each component, the programmer unit tests it to ensure that it:

* Complies with programming standards;
* Runs error free;
* Meets all the requirements listed in the software design specification; and
* The programmer corrects any errors the unit test identifies.

3.4 The programmer checks all new and modified software components into the software component library.

3.5 The software designer reviews and tests all software components to ensure that they comply with programming standards. Any variances shall be noted and steps 3.1 through 3.5 repeated until all variances are resolved.

### 4.0 PROGRAMMING REVIEWS

4.1 Throughout the project life cycle, the project team conducts team reviews of the work products being built. The types of reviews may vary from formal design reviews to informal code walk through, according to the plan set by the project team, to ensure best use of time spent on the review.

* 1. It is important to ensure that all people involved in the project understand the content of a given work product, and identify any changes needed in the work product before starting work on other work products that depend on it.
  2. ITSW105-1 WORK PRODUCT REVIEW CHECKLIST contains items to consider when planning, conducting, and following up actions after a review. The project leader and/or Project Manager should review the work product, using ITSW105-1 as a guide.

**Forms:**

* ITSW105-1 WORK PRODUCT REVIEW CHECKLIST

**References:**

* 1. **ISO/IEC 12207:2008 – SYSTEMS AND SOFTWARE ENGINEERING – SOFTWARE LIFE CYCLE PROCESSES**
  2. **IEEE 12207-2008 – SYSTEMS AND SOFTWARE ENGINEERING – SOFTWARE LIFE CYCLE PROCESSES**

This ISO standard describes the major component processes of a complete software life cycle and the high-level relations that govern their interaction. It establishes a software life cycle architecture based on two principles, modularity of processes and responsibility for processes. There are three process classes in the ISO software life cycle: primary (such as acquisition and operations); supporting (such as documentation and configuration management); and organizational (such as infrastructure and training). Each life cycle process is made up of activities, and each activity is further subdivided into tasks. The standard is based on ISO quality management principles.

The IEEE version of 12207 is more closely aligned with the ISO standard than it was in previous versions.

For more information, visit the ISO web site at <http://www.iso.org/iso/catalogue_detail.htm?csnumber=43447> or the IEEE web site at <http://standards.ieee.org/findstds/standard/12207-2008.html>.

**Revision History:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision** | **Date** | **Description of Changes** | **Requested By** |
| 0 | mm/dd/yyyy | Initial Release |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**ITSW105-1 WORK PRODUCT REVIEW CHECKLIST**

|  |  |  |
| --- | --- | --- |
| **WORK PRODUCT REVIEWS** | | |
| **ID** | **ITEMS TO BE CONSIDERED** | **RESPONSE** |
|  | Have the work products to be reviewed been identified? |  |
|  | Has the type of review been selected? Alternatives include:   * Informal walk through by several team members * Technical review by project team members and stakeholders * Inspection by project team members(and perhaps others) |  |
|  | Have the goals of the review been established? |  |
|  | Has a moderator/facilitator been selected? |  |
|  | Has a review package been developed and distributed to the participants with ample review time? The review package should include at least the following:   * Work product to be reviewed; * Related templates, guidelines, other background information; and * Forms with which to record defects, questions, issues. |  |
|  | Have results of the review been used to update the work product? |  |
|  | Have the goals of the review been reviewed to determine success? |  |
|  | Has process been reviewed to identify any improvements? |  |

[This page intentionally left blank]