SYSTEMS DEVELOPMENT LIFE CYCLE (SDLC) STANDARDS

POLICY

The Information Technology department at Sherman College of Chiropractic, is responsible for developing, maintaining, and participating in a Systems Development Life Cycle (SDLC) for Sherman College system development projects. All entities at the College, engaged in systems or software development activities, must follow the Sherman College SDLC. This SDLC is detailed in the Sherman College Systems Development Life Cycle (SDLC) Standards below.

Additionally, the following apply:

- All software developed in-house which runs on production systems must be developed according to the Sherman College SDLC Standards. At a minimum, a software development plan should address the areas of preliminary analysis or feasibility study; risk identification and mitigation; systems analysis; general design; detail design; development; quality assurance and acceptance testing; implementation; and post-implementation maintenance and review. This methodology ensures that the software will be adequately documented and tested before it is used in conjunction with critical and/or sensitive Sherman College information.
- All development work shall exhibit a separation between production, development, and test environments, and at a minimum have at least a defined separation between the development/test and production environments unless prohibited by licensing restrictions or an exception is made. These separation distinctions allow better management and security for the production systems while allowing greater flexibility in the pre-production environments.
- Where these separation distinctions in environments have been established, development, and QA/test staff must not be permitted access to production systems unless absolutely required by their respective job duties/descriptions.
- All application/program access paths utilized in development or testing, other than the formal user access paths, must be deleted or disabled before the software is moved into production.
- Documentation must be kept and updated during all phases of development from the initiation phase through implementation and ongoing maintenance phases. Additionally, security considerations should be noted and addressed through all phases.
- All software and web applications that create, manage, use, or transmit Level I information, as defined by the Sherman College Data Classification Policy, must be developed and maintained solely by Sherman College Information Technology. Other development work involving Level II and Level III information may be done outside of Sherman College IT provided the Sherman College Systems Development Life Cycle (SDLC) Standards are followed.

SYSTEMS DEVELOPMENT LIFE CYCLE (SDLC) STANDARDS

Sherman College Systems Development Life Cycle (SDLC) Standards
All systems and software development work done at the Sherman College of Chiropractic shall adhere to industry best practices with regard to a Systems (Software) Development Life Cycle. These industry-standard development phases are defined by ISO/IEC 15288 and ISO/IEC 12207. The minimum required phases and the tasks and considerations within these Systems development phases are outlined below. All of the following sub-tasks and considerations, as listed in the below respective standard development phases, are mandatory if the system or software development deals with Level 1 data (see Policy 3305 Data Classification) in any way. Otherwise, the sub-tasks and considerations are recommended steps within the required standard development phases.

System Initiation
- A need or opportunity is defined.
- Concept proposal is made.
- An initial feasibility study is conducted.
- A project charter (if necessary) is formulated.

System Requirements Analysis
- Analyze user needs and develop user requirements.
- Create a detailed Functional Requirements Document.
- Break down the system, process, or problem into discrete units or modules and utilize diagrams and other visual tools in order to analyze the situation or need.
- Any security requirements must be defined.

System Design
- This phase transforms the requirements into a Design Document.
- The functions and operations of the system or software being designed are described in detail.
- A risk analysis should be done between the System Requirements and System Design phases.
- A final design review should be done to ensure the design addresses practicality, efficiency, cost, flexibility, and security.

System Construction (Procurement)
- This phase entails the transformation of the detailed design documents into a finished product or solution.
- Manual and automated testing at a unit or module level is done throughout this phase by the system or software developers. Security considerations are taken into account during testing.

SYSTEMS DEVELOPMENT LIFE CYCLE (SDLC) STANDARDS

- A third-party product may be utilized as a system or software solution if it best fits the user requirements and is more practical from a budgetary and/or resource perspective. However, all of the next phases should be followed regardless of whether the solution was developed in-house or purchased.

System Testing and Acceptance

- This phase should validate or confirm that the developed system or software meets all functional requirements as captured during the System Requirements Analysis phase.
- Representatives separate from the development group should conduct internal Quality Assurance (QA) testing.
- Representative(s) from the user group should conduct user acceptance testing.
- Documentation during testing should detail and match testing criteria to specific requirements.
- While unit and module testing should be done throughout the entire SDLC, this phase entails holistic testing of the finished product and the final acceptance testing by the user(s).
- Final security assessment testing is now conducted.
- Any problems identified during the previous phases must be resolved or remediated before implementation.

System Implementation

- The finished, tested, and user-accepted system or software is moved from the testing environment to production.
- All tools, code, or access mechanisms used for development or testing of the system or software must be removed from the software that is being moved into a production environment.
- Any necessary user training should be done prior to or during this phase.

System Maintenance

- This phase is the ongoing life of the system or software. Unlike the other phases, this phase only ends when the system or software is decommissioned.
- A customer/user support structure and any other necessary operational support processes should be in place.
- Any planned changes to the system or software should be scheduled, communicated, and documented.
- Continuous security penetration testing is conducted on the system or software throughout its life cycle at regularly scheduled intervals.
- Mandatory security testing is conducted when any major configuration or architecture change is made.

SYSTEMS DEVELOPMENT LIFE CYCLE (SDLC) STANDARDS

Consequences
Use of College electronic information resources contrary to this policy or applicable federal, state, or local law is prohibited and may subject the user to disciplinary action including, but not limited to, suspension of the user's access to the electronic information resources. Users also should be aware of other possible consequences under college policies and federal, state, or local laws, particularly those related to computer crime and copyright violation.

Recourse/Appeals
If a department or individual feels that actions taken under this policy have been inappropriate, a review of the decision may be requested. The review will be conducted by the provost and executive vice president. If, after the review, there is still a disagreement with the decision, appeals should be directed through the existing procedures established for employees and students.