

**Drake University  
Computer Information Systems  
Software Development Life Cycle  
Policy and Procedures**

Applications Covered Under this procedure are:

DUSIS which includes: SCT Banner including Self Service, Bookshelf and Online Help, SCT Banner Xtender Solutions, SCT Workflow, Eprint

Luminis which includes: SCT Luminis Content Management System, SCT Luminis Platform System

AdAstra Room Scheduling

Blackboard

RDBMS and Applications servers such as Oracle, Oracle Application Server, MicroSoft SQLServer.

Custom developed applications regardless of programming language.

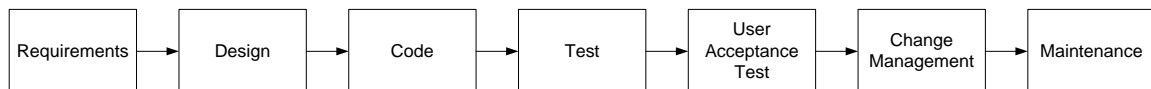
## **Software Development Life Cycle (SDLC) Defined**

The software development life cycle (SDLC) is the formal process followed by a software development team to move business applications from concept to implementation. The SDLC applies to all application development within the CIS team at Drake University.

In order to enter the SDLC one of two events must first occur. The first is a work request initiated by one of the functional business units and approved and assigned by the CIS Director. The second is a trouble ticket initiated by the help desk. Work requests are initiated when new function is required that does not yet exist. A trouble ticket is initiated when existing function is broken. Only trouble tickets that require code changes will follow the SDLC.

## **The SDLC Process – Work Requests**

Application development initiated by a Work Request must completely follow the phases of the SDLC. No phase of the SDLC may be skipped and they must take place in the order presented. However, during the SDLC process it may be necessary to start the process over or return to one of the previous phases of the process. If this happens, the steps of the process must again occur in the order presented from the phase it was determined it was necessary to return to.



**Drake University  
Computer Information Systems  
Software Development Life Cycle  
Policy and Procedures**

### ***Requirements Phase***

The requirements phase covers all of the activities involved in discovering, documenting and maintaining a set of requirements for a computer-based system. There are functional requirements and non-functional requirements. For example, a functional requirement may be to identify a person in the database and update their record accordingly. The non-functional requirement might be to do so in 4 seconds or less.

Requirements are identified by the project stakeholders. Stakeholders are people who will be affected by the system and who have direct or indirect influence on the system requirements. Stakeholders will normally include the initiator of the work request, however it is important to recognize that other stakeholders will exist when the system is integrated across functional business units. For example, a set of requirements to change the way telephone numbers are processed and stored would impact most functional business units.

There are 3 basic steps in the requirements phase:

- Requirements Elicitation – The system requirements are discovered through consultation with stakeholders.
- Requirements Analysis and Negotiation – The requirements are analyzed in detail, and a formal negotiation process occurs involving all the different stakeholders to decide which requirements will be accepted. Additionally, not all requirements may be delivered during the same project phase. Delivery of requirements in different project phases will be determined by the priority of the requirement and the date it is needed as well as resources available to do the development.
- Requirements Validation – A careful check of the requirements for consistency and completeness.

At Drake, requirements are formally defined by the stakeholders in the Work Request. The Work Request document is available to the programmer/analyst for discussion and more detailed definition of requirements. However, the Work Request also defines the scope of the project. The scope of work should never change unless a new Work Request is issued and approved by the CIS Director. No other formal requirements document is required for Banner related application development. Other requirements documents for development outside of Banner may be required.

### ***Design Phase***

The Design Phase covers the activities of the programmer/analyst to create or modify the system to meet the functional and non-functional requirements. The design phase may

**Drake University  
Computer Information Systems  
Software Development Life Cycle  
Policy and Procedures**

include many types of technical documentation. Documentation required will be determined by the CIS director. Additionally, the CIS Director may require a formal design review on larger projects. The CIS Director will notify the programmer/analyst when the assignment to the work request is made whether or not a design review will be required and what documents the programmer will be expected to produce.

***Code Phase***

The Code Phase covers the activities of the programmer/analyst to develop the design into functioning code. Code related to the SCT Banner system should follow SCT standards and guidelines for customizing and augmenting the Banner system. These are defined in the General Technical Reference Manual which can be found in the Banner Bookshelf. Code not related to the SCT Banner system should follow any standards and guidelines set forth by the CIS Director. In both cases, additional standards may be imposed.

***Test Phase***

The Test Phase covers the activities of the programmer/analyst to ensure that the code will function to meet the requirements, the code will not break function that currently exists in the system and that the code will not drain system resources and performs as required by the defined non-functional requirements.

Formal test plans and scripts are not required by Drake. However, the programmer is expected to perform all testing necessary to ensure the code performs as expected. Formal test plans and scripts may be required in the future if it is determined that testing being performed by the programmers is not adequate.

***User Acceptance Test Phase***

The User Acceptance Test Phase covers all of the activities under which the stakeholders perform to ensure the new software meets their requirements. Formal test plans and scripts are not required by Drake. However, the stakeholders are expected to perform all testing necessary to ensure the code performs as expected. Formal test plans and scripts may be required in the future if it is determined that testing being performed by the stakeholders is not adequate. In all cases, the user is required to sign off to the CIS director that the code has been tested and is ready for promotion into the production environment.

***Change Management Phase***

The Change Management Phase includes documentation of the change requirements and promotion of code to the production environment. See [Banner Production Change](#)

**Drake University  
Computer Information Systems  
Software Development Life Cycle  
Policy and Procedures**

[Management Policies and Procedures 1.1](#) for detailed information on Change Management.

### ***Maintenance Phase***

Maintenance covers all activities that occur on the computer system after the code has been placed into production through the Change Management process. Once the code is placed into production, all non-conformities to documented requirements or other problems should be reported to the help desk at extension 3001 or [helpdesk@drake.edu](mailto:helpdesk@drake.edu). [The Banner Production Problem Resolution Policy and Procedure](#) should then be followed. Any requirements that were not originally documented in the Work Request are not trouble tickets. New Work Requests will need to be issued for additional requirements. For example, a report was developed by CIS following the SDLC. After using the report for a short period, the user determines that they need to see additional information on the report and they need to see the data sorted in a different order. These requirements were not part of the original work request and therefore a new Work Request is required.

### **The SDLC Process – Trouble Tickets**

The SDLC process for Trouble Tickets is the same as the SDLC process for Work Request with the following exception. Instead of building new function which is the purpose of a Work Request, a trouble ticket is to report a problem with existing functional or non-functional requirements. This is commonly known as a bug. The trouble ticket should define how the software in production is not performing according to requirements. The trouble ticket will serve as the “requirements” document. Not all trouble tickets will result in performance of the SDLC process. Only those trouble tickets that require code changes are required to follow the SDLC process.