



University
of Houston
Clear Lake

UCT Application Development Lifecycle

UCT Business Applications

February 24, 2016

Table of Contents

Planning Phase.....	1
Analysis Phase.....	2
Design Phase.....	3
Implementation Phase.....	4
Software Development.....	4
Product Testing.....	5
Product Implementation.....	6
Maintenance Phase	7

Planning Phase

The planning phase includes the following components:

- Project Request
- Feasibility Study
- Project Plan

The project request is often handled by management. If accepted, the project will proceed to the feasibility study where developers and management will discuss the project and determine whether or not it is feasible to proceed. If the project is feasible, a project plan will be created before proceeding to the analysis phase.

The scope and depth of the feasibility study and project plan vary based on the complexity of the project. For simple projects the project request is nearly the same as the project plan, which allows us to proceed directly to the Analysis Phase and begin the requirements gathering process, whereas large and more complicated projects often involve multiple discussions before proceeding to the Analysis Phase.

The project plan will include the defined scope and purpose of the project, the stakeholders, and an estimate of the project's timeline for completion along with the basic work breakdown. Large projects may define a series of milestones within the project plan, breaking the project's scope into a series of projects. These milestones will focus on implementing the requirements and features over a series of releases.

Analysis Phase

The analysis phase includes the following components:

- Requirements Definition
- Use Cases Defined
- Data and Process Models Defined

The requirements definition is produced through a series of meetings and interviews with the stakeholders. During these meetings and interviews, we gain an understanding of the existing business processes and workflows, identify areas for improvements, and learn the nature of the data and information involved with the project. With the scope and goals of the project in mind, this information is used to help create a strong set of requirements that will be used to define the use cases, data models, and process models.

The use cases will define the set of activities to be performed within the project. These can be automated events, or user-driven events within the system. The use cases will rely heavily on understanding the workflow of the business process and the expectations of the stakeholders on how the system will work. The use cases can also have a strong impact on how the data and process models are designed.

The process of defining the data and process models is dependent on the requirements definition and use cases being created, reviewed, and approved by the stakeholders. These models define the key data fields, data types, events, and workflows of the system and how they are related. The models are then used to define the database models and application APIs during the Design and Implementation phases of the project. Any changes to the requirements and/or use cases will trigger a review of the data and process models and likely result in modifications to the program's design and implementation.

Design Phase

The design phase includes the following components:

- Architecture Design
- Database and File Specifications
- Program Design

The architecture design will identify what hardware, databases, software, and other technologies to be used to develop and implement the project. A large number of projects that we work with are database applications with web interfaces that allow us to leverage existing hardware, servers, databases, and other services. Some projects require the implementation of new hardware, servers, etc. Once the architecture design is established, the project will proceed to the database/file specifications and the program's design phases.

The database specifications will result in the creation of an entity relationship diagram (ERD) that documents the names of tables and views, along with their columns, data types, and constraints. This diagram is always produced prior to the implementation phase, as it has a direct impact on how the program's software will be written. During the implementation phase, prior to project's production deployment, the diagram may be modified several times based on debugging, testing, and other findings. The diagram is also reviewed and approved by database administrators prior to implementation and deployment of the project in a production environment at the end of the implementation phase.

The file specifications is similar to that of the database specifications, expect that it will deal with the data structures in a flat-file format. This could be plain text, binary, CSV, XML, JSON, or other proprietary formats based on the needs of the project.

The program design is an optional document, as it's closely related to the database and file specifications. The project's API created during the implementation phase will document how the application interfaces with the database and file specifications. The program design document is often created when the project requires interaction with a third party system and will explain how the project interfaces with the external system. An example would be how a credit card processing system will interface with a third party payment processor for confirming payment attempts and acknowledging success/failure messages between the systems.

Implementation Phase

The design phase includes the following components:

- Software Development
 - Establish Repository
 - Data Models
 - System Interface
- Product Testing
 - Test Plans
 - Product Testing
- Product Implementation
 - Migration Plans
 - User Documentation and Training Plans
 - Change Management Plan
 - Deployment

The implementation phase is often the most time consuming phase as it encompasses the majority of the work to be completed. Before software development can begin, any architectural requirements for development and testing purposes must be resolved. This may involve coordination between systems administrators and database administrators to create the server, database, and other environments necessary. The process of creating these environments will be documented for use during the product's implementation.

Software Development

The software development phase includes the creation of the project's repository which will provide change management and versioning of the software and supporting documents, the creation of the data models and supporting code base, and the development of the software's interface.

The repository may be a Subversion or Git repository. The product's documentation will be placed within a Documentation folder, and the application's code base will be placed within a separate folder, often named after the project itself or an appropriate acronym. The repository allows the code to be distributed between multiple developers for team development. Developers are responsible for checking in code on a regular base and communicating changes with other developers assigned to the project.

The data models are often created as its own software project or library, especially for large projects. This data model will create the objects that reflect the previously defined database and file specifications, as well as the CRUD (Create, Read, Update, and Delete) operations required by the project's use cases and workflows. The database models will be used to create a series of SQL scripts that establish the required tables, views, and other database objects.

The systems interface will implement the data model and establish the user interface and/or automated processes required by the project. This is often a web-based interface, but may include a console or other graphical user interface.

Product Testing

Product testing is initiated after the software development phase has implemented all of the required features. The project's code base is deployed to a testing environment with its own database. Staging data will be imported as necessary to provide a functioning environment for users to use the system.

Test plans are created to help guide the testing of the key components and features, and is designed to help ensure that everything is functioning as expected by the stakeholders. The test plans will address the testing of data entry workflows, reports, and other components of the system to ensure that data is not only being entered and displayed correctly, but is being processed correctly as well. The test plan may be broken up into different plans for different groups of users, often by access-level. For example, Administrators will have an additional administrative test plan, and non-administrative users will only have a test plan that only focuses on the components they would normally have access to.

The stakeholders and users will review the test plan and use them to test the system. Any findings are documented and reported back to the software developers for review and resolution. Based on the findings, the project may return to the Design Phase or returning to the Software Development portion of the Implementation phase. Once all issues are addressed, product testing will repeat until all issues are resolved and the stakeholders are satisfied with the product.

Product Implementation

Once product testing has been completed, the project will move forward with implementation in a production environment. At this point the systems administrators will proceed with implementing any architectural requirements needed by the project. Database administrators will review and approve any database diagrams and create the schemas using SQL scripts provided by the developers.

If the project involves upgrading an existing system, or converting from one system to another, a migration plan will be created and used to proceed with the migration. This could include flat file export/import scripts, SQL scripts, or other custom processes used to convert data from one format to another.

User documentation and/or training plans will also be created for the product depending on the target audience and stakeholder requirements. These documents will also be used by support staff during the maintenance phase to help troubleshoot any user-reported issues.

A change management plan may be created to document how change requests, bugs, and other issues will be handled by management and support staff. If certain features are scheduled for a future release, the change management plan will address these as well.

After all required documents are produced, they will be reviewed by management and the project stakeholders for approval. Once everything is approved, the project will be deployed to its production environment, often involving coordination between systems administrators, database administrators, support staff, developers, and the stakeholders. When a database is involved, the production database credentials will be communicated between the database and systems administrators, and the software developers will not have access to them. Software developers will only know the name of the connection pool, since it will be included in the application's configuration for deployment.

Once the project is deployed, it will be reviewed and tested by the software developers, stakeholders, and other users to ensure it is functioning correctly. Once functionality is confirmed, the project will enter the maintenance phase.

Maintenance Phase

As the project enters the maintenance phase, support documents are created and delivered to the support staff. These documents will address any common questions and include instructions on how to troubleshoot known issues. Guidelines for handling bug reports and change requests will also be included in the support plan to help facilitate communication between the appropriate stakeholders, developers, and management staff. These documents may be updated over time to include new questions and instructions as necessary.

When bugs are reported and confirmed by developers, the project will repeat the analysis/design/implementation phases as required to resolve the issues and deploy any fixes to the problem.