**Chapter 2**

**ANALYSIS PHASE**

**A. Requirement Analysis**

This is the process of determining user expectations for the proposed system. This involves frequent communication with system users/respondents to determine specific feature expectations. The proponent conducted an interview from students, faculty and alumni who previously developed a study similar to proposal.

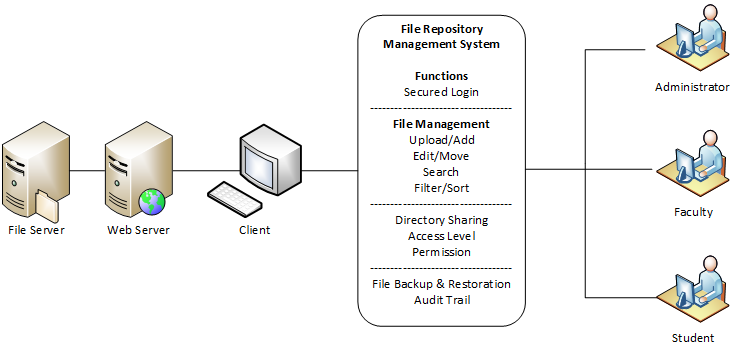
**Summary of the Complete Requirements for the Proposed System:**

Following is a summary of the list of complete requirements defined for the “File Repository Management System”:

1. The administrator, faculty and student should able to login to the system.
2. The system should be able to provide a user friendly interface and comprehensive logical flow of pages which consists of main menu, administrator menu, faculty menu and student menu.
3. There are three user account types based on their level of security namely admin, faculty, and lastly student.
4. The system should have a basic function of file management and operation namely add, edit, delete and search.
5. Every student must be currently enrolled in a current academic year per semester to be able to access the system. This will be done by the administrator.
6. The administrator should manage first the students based on the current enrollee master list. Their account will be activated and ready to use during laboratory activities.
7. The administrator should load the standard directory structure every subjects that have laboratory activities.
8. The student should have the following functions over their file management interface:

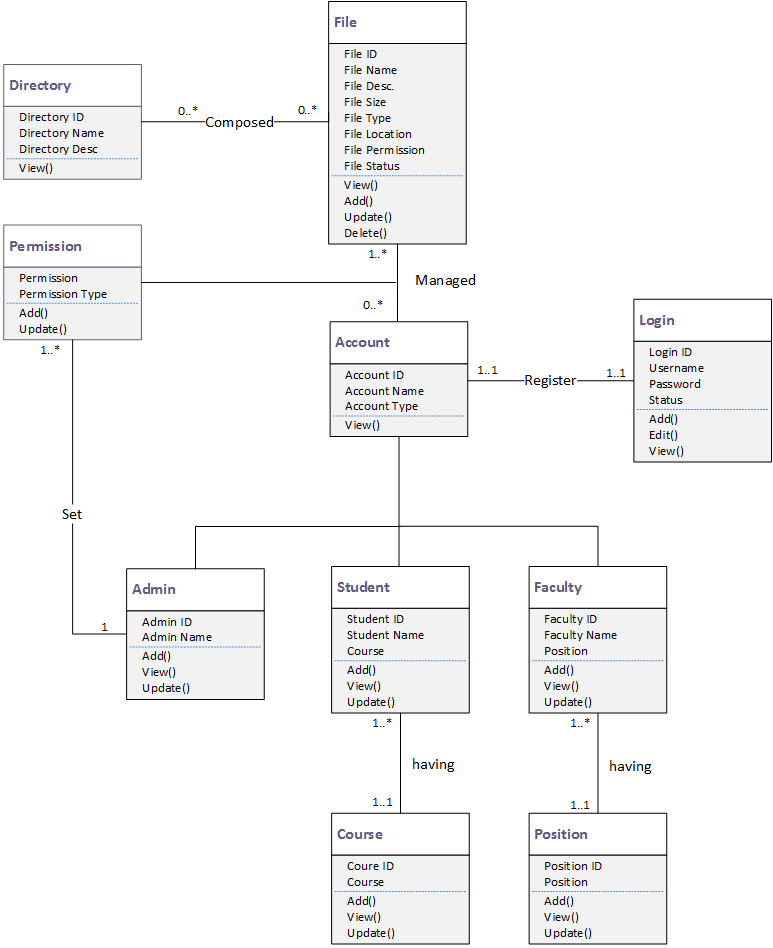
* upload file function which limited only to file type and file sizes that is set by the administrator;
* move file function from one directory to another;
* file deletion and it will automatically remove from the server;
* file indexing and searching to be able find any files;
* file filtering and sorting;
* file restoration and automatic backup;

1. The faculty should have setting control over their directory or folder. These controls are directory lock, directory permission and directory sharing to the students. It also provides a task notification function from student to faculty in every activities they do.
2. All the file related documents, activities, laboratory and references that set by the administrator should save or store in the file server.
3. **System Design and Implementation**
   1. **Detailed Architecture of the System**

****

**Figure 1**

**Architectural Diagram**

* 1. **UML Class Diagram**

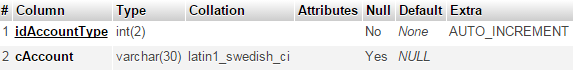
**Figure 2**

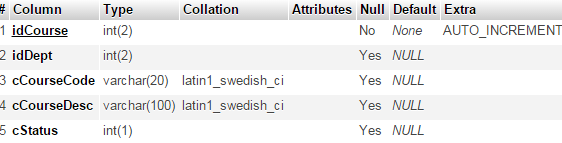
**UML Class Diagram**

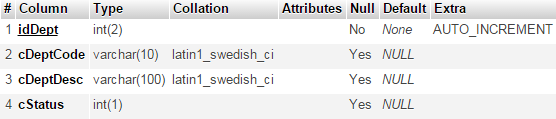
* 1. **Data Dictionary**

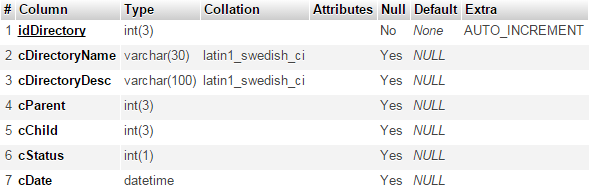


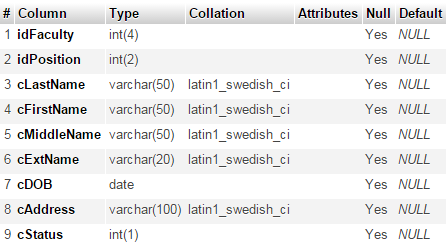
**Table 1: Academic Year**

**Table 2: Account Type**

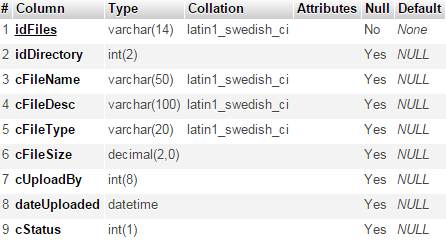
**Table 3: Course Information**

**Table 4: Department Information**

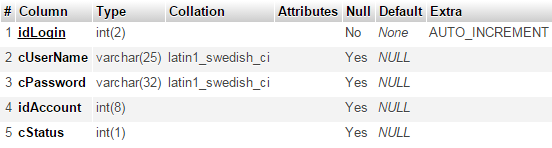
**Table 5: Directory Information**

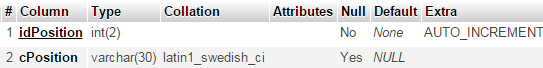
****

**Table 6: Faculty Information**

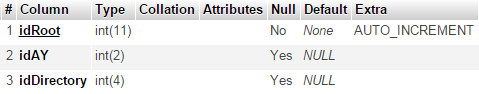
****

**Table 7: File Information**

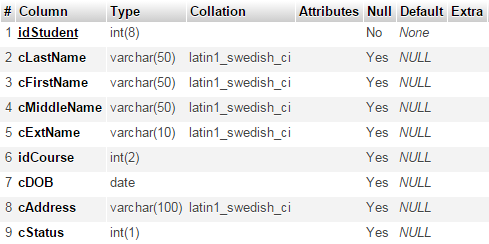
**Table 8: Login Information**

****

**Table 9: Position Information**

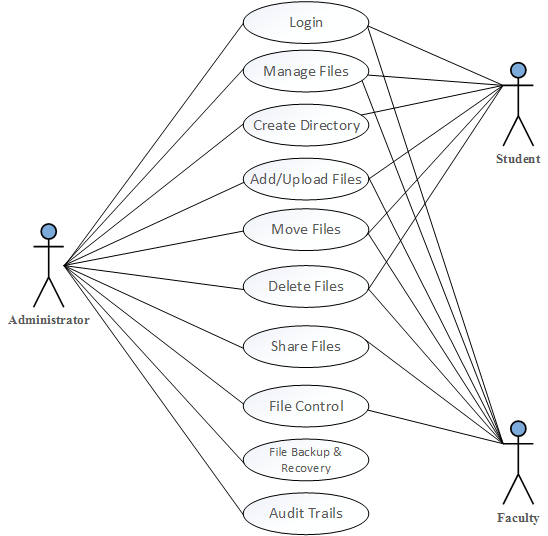
****

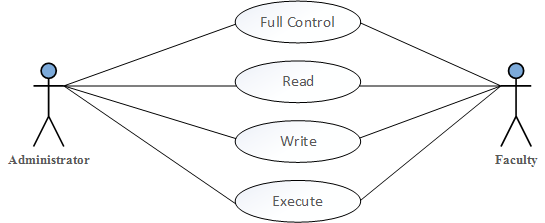
**Table 10: Root Directory**

****

**Table 11: Student Information**

* 1. **Use Case Diagram**







**Figure 3: Use Case Diagram**

* 1. **Use Case Diagram Descriptions**

|  |  |
| --- | --- |
| Use Case ID: | 001 |
| Use Case Name: | Manage Files |
| Actors: | Administrator, Faculty and Student |
| Description: | The process allows the actors to manage their files. |
| Pre-Conditions: | The standard directory structure exist in database |
| Post-Conditions: | The actors can manage their directory and files |
| Normal Flow: | 1. They can view the directory tree of their files.  2. They can see the operation of their files |

**Table 12 Manage Files**

|  |  |
| --- | --- |
| Use Case ID: | 002 |
| Use Case Name: | Create Directory |
| Actors: | Administrator, Faculty and Student |
| Description: | The process allows the actors to create their own directory |
| Pre-Conditions: | The new directory does not exist in the database and file server. |
| Post-Conditions: | The system will save the new directory information and allow them to see it. |
| Normal Flow: | 1. The system will generate new id for directory.  2. The actors will enter the details. |

**Table 13 Create Directory**

|  |  |
| --- | --- |
| Use Case ID: | 003 |
| Use Case Name: | Add / Upload Files |
| Actors: | Administrator, Faculty and Student |
| Description: | The process allows the actors to add or upload their files. |
| Pre-Conditions: | The new file does not exist in the database and file server. |
| Post-Conditions: | The system will save the new file information and allow them to see it. |
| Normal Flow: | 1. The system will generate new id for files.  2. The actors will enter the details. |

**Table 14 Add/Upload Files**

|  |  |
| --- | --- |
| Use Case ID: | 004 |
| Use Case Name: | Move Files |
| Actors: | Administrator, Faculty and Student |
| Description: | The process allows the actors to move their files to desired directory. |
| Pre-Conditions: | The files exist in the database and file server. |
| Post-Conditions: | The system will save the new file location and allow them to see it. |
| Normal Flow: | 1. The actors will enter the details. |

**Table 15 Move Files**

|  |  |
| --- | --- |
| Use Case ID: | 005 |
| Use Case Name: | Delete Files |
| Actors: | Administrator, Faculty and Student |
| Description: | The process allows the actors to delete or remove their files. |
| Pre-Conditions: | The new files exist in the database and file server. |
| Post-Conditions: | The system will remove the file information and create a log file for audit trail. |
| Normal Flow: | 1. The administrator will see the deleted files in event log |

**Table 16 Delete Files**