ANALYSIS

**3.1 Introduction**

This chapter outlines the processes that have been carried out to produce the analysis required for modeling the artifact. In theory a detailed, solid analysis will intern produce a ‘good’ project and therefore the processes in this chapter are indispensable. This chapter will contain an outline of how the research material has been analyzed which in tern was then modeled to produce a set of key entities.

To start this chapter, it discusses the system development life cycle and methodology which will be used in the development of the software artifact.

**3.2 Methodology Selection**

The development life cycle and methodology selected for this project is the traditional life-cycle model more commonly known as the Waterfall method with a prototype approach. There are numerous reasons why this approach has been suggested. Frenzell (1999, p.236) suggests that the Waterfall approach offers many advantages such as complex activities becoming easier to understand, various interactions between developer and client are easier in an incremental environment and it makes the system easier to evaluate on an interim basis.

With the advantages that are defined here it can be concluded that this methodology suits the project and this report has been structured to support this. The reasoning behind the use of a prototype approach is due to the ease at which it could be implemented into the life cycle.

At each stage of the life cycle any function could be evaluated and then passed back a stage to be re-thought and redeveloped thus increasing the quality of the artifact.

Requirements gathering

And systems analysis

System Design

Implementation

Deployment of System

Testing

Evaluation and Maintenance

*Figure 3.1: The Waterfall Model*

**3.3 Research and Information Gathering**

A key step for analysis was the identification of key entities and attributes from the research data of the university information management system. This was a key process that required a high standard of work to ensure the entities and attributes were considered important, reliable and useful not only now but also in the future.

This was possible by looking at trends within the data. It makes sense to make any mistakes before any development has begun as mistakes are much easier to repair at the analysis stage rather than during development.

A simple tabular format was chosen as the method to document the research findings. The potential volume of data that could be identified meant that it made sense to limit the analysis to a selection of solutions. From this a broad range of solutions were identified and then filtered and cut down to finally produce a sampler data set which could be heavily focused on and thus produce the quality of data needed.

The method used for documenting the research had been devised to allow quick and easy access with as little hassle as possible. The reason for this was due to the nature of how the research was carried out.

As new elements were identified in different university information management solutions it was felt using a tabular format new attributes of various entities could be entered with ease in there own row and the analysis could then be carried out in a sequential process down the list of data items. This made more sense than having an organized ‘clump’ of data for each solution. This method could be considered crude and it was not based around any solid research method however it achieved the desired results.

**3.4 Identification of Key Entities Attributes**

Using the research material that has been discussed above, key entities and attributes were identified.

At the end of the system analysis, possible suggestions and inputs were made of what was felt needed to be included in the system development.

This task was a key milestone in the project as it ensured that the set of entities and attributes selected produced concrete sampler data set which the rest of the project would ultimately be built upon.

**3.5 Reasons for Final Attribute Selection**

These have been summarized and explained in this section. The main areas which are covered consist of Setup, Personals, Academics, Assessments, Users Management, Registry, Finance, Library and Hostel Accommodations.

**3.5.1 Setup**

This module is responsible for setting up parameters that would be consumed. Such duties involve activating or deactivating an account, creating schedules, and/or granting necessary privileges. This module is available mainly to the administrator.

*Approvals*: This entity handles the approval or rejection of records by designated officers.

*Area of Specialization*: This entity is responsible for managing the different areas of discipline of a staff in the community.

*Countries*: This entity is responsible for location or country where a student or staff came from.

*Course Numbering*: This entity handles the university course numbering system.

*Course regulations*: This entity administers the specific rules that guides each course

*Courses*: This entity is responsible for managing the courses in the university.

*Departments*: This entity is responsible for managing the departments in the university.

*Designations*: This entity is responsible for granting or revoking privileges to users within a specific period of time.

*Durations*: This entity is used to enforce time constraint on a project. Projects can involve examinations, semester, and so on.

*Entry requirement*: This entity set rules or criteria for enrollment into the university.

*Faculties*: This entity is responsible for managing the university faculties.

*Grading system*: This is a mathematically-driven entity that set up and manages the university grading system that can be used to compute students’ results.

*LGAs*: An entity linked to the country entity- responsible for local government of where a staff/student came from

*Parameters*: This entity is used to identify smaller entities (entities that do not require having a standalone entity in the database).

*Permissions*: This entity is responsible for managing set of rules or permission based on student/staff role in the system

*Schedules*: This entity is used setting up and managing the university time table for time-table management.

*Sub Courses*: This entity is responsible for managing various courses curriculum per semester and per year.

*Universities*: This entity manages information about the university.

**3.5.2 Personals**

This module sole purpose is to manage personal records of both staff and students. Management of records includes an orderly collation of the records, a systematic processing of the given data, and an efficient storage for easy retrieval.

*Addresses*: This entity is responsible for managing addresses in the system

*Area of Expertise*: This entity manages the area of expertise of staff

*Biodata*: This is information relating to a particular person and his or her financial, professional, or educational history, stored in a database used to identify such a person within a group. A student of ACU for example should have a name(s), sex, date of birth, place of birth, and so on. These are his/her biodata.

*Certificates*: This entity is responsible for managing awarded certificates.

*Choices*: This entity manages the individual choice (first and second of courses) of students for seeking admission into the university.

Other entities in this module include:

* Computer Skills
* Declaration
* Emails
* Emergencies
* Employment Histories
* Hobbies
* Religions
* Sports

**3.5.3 Registry**

This module is responsible for handling student enrollment, admissions, examination permissions, hostel accommodation permission etc.

*Admission*: The admission entity manages the admission process of the system. It makes use of the record filtering process for short-listing admitted students.

*Exams*: The exams entity is responsible for short listing names of students that are qualified for entrance examination into the university based on the set criteria e.g. POST UME.

*Students*: This is the entity manages registration of both newly admitted students into the university program and registration of returning student (already admitted students) per session.

*Tickets*: This module handles the generation of scratch card details for registration purposes. It has details like Serial No & Pin Code.

**3.5.4 Assessment**

This module is used to manage the day to day attendance/academic records activity of a particular student.

*Attendance*: This entity manages the attendance rate of a student. Attendance could be managed academically (e.g. class, exam, laboratory) or administrative (e.g. faculty forum, accommodation).

*Records*: This entity manages the daily assessment of a student in the particular course. Records are updated into individual account to ease computation at regular intervals.

**3.5.5 Academics**

This module is manages students course registration processes per semester and per session.

*Registrations*: Academic registrations are carried out on semester basis identifying subjects a student would undertake and their respective lecturers.

*Signing Hierarchy*: This entity manages the final signing hierarchy of student’s course registration form e.g. after lecturers of various courses might have finish signing, Head of Department would finally sign followed by Faculty officer followed by the registrar followed by the dean etc.

*Submissions*: This entity handles the final submission of student registration form to the department and the faculty.

**3.5.6 Finance**

This module is responsible for managing both the financial records of a student/other university transactions. Each student account has a financial account embedded in it. This FA keeps tracks of student finances from fees, levies, and even refundable. And

*Payment Plan*: This entity among others is responsible for managing the payment plan of students school fess. The plan can be categorized into installmental or full.

*Refunds*: This entity manages the refunding of an overpaid school fees or as the case may be.

**3.6 Requirements**

Based on the issues and objectives outlined in the previous sections a set of functional and nonfunctional requirements have been identified. These requirements will be used to aid the design and implementation stages to produce a system which meets the objectives of this project.

**3.6.1 Functional Requirements**

Functional requirements are defined by Bennet (1999) cited by Melia (2003, p.2) as “What a system does or is expected to do”

The system is expected to:

* Be developed using a vendor specific software (Visual Studio .NET (C#)) which has been identified in the review chapter.
* Provide sufficient functions to view and query data populated in the back end database.
* Provide a means for information to be provided via the internet.
* Have an intuitive user friendly user interface.
* Incorporate a public page for all non-administration (Students/Staff) users to view.
* Be maintainable so that it can be expanded to allow for future products or services in the technology which will need to be recorded and analyzed.
* Be password and username protected so that only authorized users can update, delete or add any records from within the database.
* Meet with all copyright licenses and any data protection act laws that may cover the information stored by the RDBMS.

**3.6.2 Non-Functional Requirements**

Non functional requirements are defined by Bennet (1999) cited by Melia (2003, p.2) as “Aspects of the system that are concerned with how well it provides the functional requirements”

The system will provide the above requirements by:

* Implementing a centralized database system developed and then administrated on the Microsoft Sql Server 2008 Standard Edition.
* The system will be programmed in ASP.Net (C#) a well known Object Oriented Programming language this should be adaptable to any future internet extensions.
* The web interface will be required to contain many functions, all of which are for public, student and admin use. These functions can be distinguished using login information.
* Providing a front end for all database data manipulation functions to be carried out through C# code.

**3.7 Summary**

In this chapter we have discussed the analysis processes that have been used. The results of these processes will cascade down the Waterfall life cycle and contribute to the design of the system.

In doing this a main objective of this project has been achieved. The research that has supported the analysis was conducted using no particular method; however the process chosen has achieved the desired results.

It was interesting to find out how many areas each of the university information management system could be divided up into.

**3.8 Key Points**

* The analysis section has been focused mainly on bringing together the research material and focusing attention on identifying the important and crucial attributes which were required to be of modeled to a high standard.
* Due to the infancy of the system researched, research has been a struggle with recourses being scarce. It is hoped that this project will help start a wave of future projects that will help improve and restructure the university management information system.
* The data has been analyzed to the extent the database and further stages can be designed.