|  |
| --- |
| **Project: Admission Office Software** |
| **Requirement and Analysis Document** |
|  |
| This project aims at providing PKFokam Institute of Excellence with a Software to support the management of the academic maters of the University |
|  |
| **Author: Mr. Herman Mekontso** |
| **Lecturer CS/IT – PKFokam Institute of Excellence, CS Researcher** |
|  |



Table of contents

[1 – Project overview 3](#_Toc90296798)

[1.1 – Purpose of the project 3](#_Toc90296799)

[1.2 – Modular decomposition of the project 3](#_Toc90296800)

[1.2.1 Students management 3](#_Toc90296801)

[1.2.2. Staff management 3](#_Toc90296802)

[1.2.3 Grades management, results consolidation and transcripts generation 3](#_Toc90296803)

[1.2.4 Attendance management 4](#_Toc90296804)

[1.2.5 Courses Management 4](#_Toc90296805)

[1.2.6 Program Management 4](#_Toc90296806)

[1.2.7 Settings Module 4](#_Toc90296807)

[2. Current status of the project 4](#_Toc90296808)

[3. Improvements and Future work 5](#_Toc90296809)

[3.1 Improvements 5](#_Toc90296810)

[3.2 Future work 5](#_Toc90296811)

[3.2.1 On the software 5](#_Toc90296812)

[3.2.2 Parallel applications/modules that could be developed around the application (exploiting data from the database) 5](#_Toc90296813)

# 1 – Project overview

## 1.1 – Purpose of the project

The software we intent to design is an admission office manager that will assist admission officers in their daily tasks. In effect, having noticed the number and the diversity of responsibilities entrusted to these people, they have to be assisted by automated tools in order to be efficient. Daily activities of admission officers can be grouped in 6 modules plus a settings module:

* The management of students;
* The management of the staff (lecturers and administration staff);
* The management of grades, results, and transcript generation;
* The management of attendances;
* The management of courses;
* The Management of programs.

The purpose of this project is to provide a tool which will support admission officers in their daily tasks.

## 1.2 – Modular decomposition of the project

From the previous section, we can infer that our software will consist of 6 main modules.

### 1.2.1 Students management

This module will allow keeping track of information about students from the time of their registration to when they leave the school. We should be able at any time to get detailed and personal information, as well as a history of progression, and the registration form of each student.

### 1.2.2. Staff management

This module will be used to manage the staff of PKFokam Institute of excellence. The staff includes the administration members (the President, the Rector, the Deans, the Head of departments, the Academic Affairs Officer, the Administrative and Financial Offier), lecturers (part time and permanents), and other supports like cleaners, etc.

### 1.2.3 Grades management, results consolidation and transcripts generation

This module will be used to register grades, produce transcripts, and generate summaries of grades anytime needed. This module will also allow getting relevant statistics concerning grades. In short, this module can be split in four sub modules:

* Grade management: includes the assignment of a grade a student got to a test set for a given course, update, delete, or see details about the grade;
* Transcript generation: generating student transcripts anytime;
* Reporting: reports of all kind will be generated from this module;
* Grade summary automatic generation for end of semesters jury.

### 1.2.4 Attendance management

Discipline is a key factor of success in a University like ours. So managing students and lecturer’s attendance is a relevant matter. This module will allow the admission officer to keep track of attendances and generate statistics and graphs of presence and participation. Attendances should be registered and summarized in the student space in order to have in real-time information on attendances to alerts parents in case of alarming students disciplinary situation. Here, the admission officer will create schedules, generate attendance lists, and mark the presences, based on the hard copied returned by lecturers.

### 1.2.5 Courses Management

This module is one of the most important one in a dynamic schooling system as the one we are to develop. It is through this module that new courses will be added, existing courses updated or deleted. A course opened a semester for a given program is not guarantee to figure in that same program subsequent semesters. This is the reason why at the start of a semester, courses have to be opened for different programs. This is mandatory in order to keep a maximum level of flexibility. Through this module, we will also be able to have amongst others, the list of students registered to a given course a given semester.

### 1.2.6 Program Management

This module is designed to handle the different programs offered by the university. Each student enrolled by the university belongs to a given department and follows a specific program. Still due to the dynamic aspect of a university, this module should provide a flexible management of different programs. For example, courses can change one semester to the other, can be assigned to a different level, or the passing grade can change from one semester to the other.

### 1.2.7 Settings Module

This module is dedicated to configuration of the software. The configuration concerns the configuration of departments and semesters:

* a new semester should be explicitly created at the start of every semester;
* If there is a new department, it has to be created and configured in the system;
* The maximum number of credits a student can take per semester should be configured;
* Closing a semester to delete information which will not be useful the following semester.

For each of these modules, relevant reports have to be provided whenever needed.

# 2. Current status of the project

* The software has been developed and installed. The first version was installed in august 2018, the admission officer has been trained to use the software and a user manual has been provided;
* The exploitation is ongoing and bugs fixing and enhancements are done gradually.

# 3. Improvements and Future work

## 3.1 Improvements

Several improvements have been performed.

* The generation of periodical and customized transcript. This was necessary because some students use to change their program and sometime request for transcripts including courses they took when they were following their former program;
* The generation of grade summaries for the jury was customized to enable the generation per faculty, per department or even per level, depending on the needs;
* The implementation of a small data source configuration module which will enable the current version of the software to be fully 2-tier;

## 3.2 Future work

### 3.2.1 On the software

One of the most accepted solutions today are multi tiers and web based applications. The actual version was developed without using any framework. The main advantage is to have a total control of the code, as well as any optimization which might need to be implemented. The drawback is the flexibility of the code, which makes the evolution of the application more difficult, because it is one person centered.

Some operations are still to be implemented to improve the current version. These operations include:

* The implementation of a caching system. The outcome will be a proxy which reduces the number of access to the database, enhancing the execution time of the application;
* The implementation of a user management module, to enable adding new user profiles, editing user profiles and deleting some existing user profiles.

We intend to move our solution to another level by restructuring it using advanced frameworks, some which have been developed at our research center to propose a more generic solution. Through this, we would like to move from the monolithic current version to a distributed core. This will open doors to the implementation of many utility tools for the University Information System.

### 3.2.2 Parallel applications/modules that could be developed around the application (exploiting data from the database)

1. Developing a view (a mobile application or a small web based application) for students to get their semester results when available;
2. Improving the reporting module to generate for example disciplinary reports for students.