

iKnow Tempus Project

SW REQUIREMENTS AND SYSTEM DESIGN
Enrolment Module

Associate Professor Toni Stojanovski, PhD

Overview

- ▶ **Functionality**
- ▶ **Work done**
- ▶ **To be done**

Overview

- ▶ ***Functionality***
- ▶ Work done
- ▶ To be done

Functionality

- ▶ **(ENR) Module for enrolment**
 - ▶ Management of enrolment of students (for every level of studies)
 - ▶ Management of candidates, ranking and completing the enrolment of new students
 - ▶ Exchange of information with the Ministry of education
- ▶ **(MGR) Migration of existing data**
 - ▶ Preparing web services, forms and specification of formats for migration of data from existing systems
 - ▶ Correction and fine tuning of migrated data

Overview

- ▶ **Functionality**
- ▶ ***Work done***
- ▶ **To be done**

Prepared document

- ▶ List of functions
- ▶ Introduction, Reference Documents
- ▶ Objective, General Overview, Scope
- ▶ Existing System(s), Benefits, Goals
- ▶ Users
- ▶ Functionalities
- ▶ UML Diagrams: Use case, Activity, Sequence
- ▶ User interface
- ▶ Data model
- ▶ Interface for web service with MoES
- ▶ Appendix A: Selection Tables

Reference documents

- ▶ Extensive list of documents which define the ranking rules
- ▶ Demonstrate the document

Benefits

- ▶ Reach to geographically scattered students. One of the important objectives of the admission system is to be able to communicate with all the students irrespective of their geographical location.
- ▶ Reduce time spent on enrolment. Reduce the time taken to process the applications of candidates, admit a candidate, conduct the online examination, verify student marks, and send confirmation letters to admitted candidates.
- ▶ Centralized data handling. Transfer the data smoothly to all the departments involved and handle the data in a centralized way.
- ▶ Paperless admission with reduced manpower. Reduce the manpower needed to perform admission and administration tasks by reducing the paper works needed.
- ▶ Cost cutting. Reduce the cost involved in the admission process.
- ▶ Operational efficiency. Improve the operational efficiency by improving the quality of the process.

Goals

- ▶ The main goal of the system is to automate the admission process at universities with improved performance and realize the vision of paperless admission. Some of the goals of the system are listed below:
- ▶ Manage large number of candidates with much data.
- ▶ Create candidate accounts and maintain candidate's data in an effective way
- ▶ View all the details of the candidates.
- ▶ Create extensive reports.
- ▶ Enable the candidates to take proactive role in the process by entering their data in the system
- ▶ Highest level of transparency: Provide the candidates with updates of the current status of the enrolment process
- ▶ Effective data management;
- ▶ Lowest possibility of human factor failure;
- ▶ Applying from distance

Users

- ▶ Demonstrate the document
- ▶ Integrate the roles with the other modules

Data model

- ▶ Applicant
- ▶ Applicants to second cycle of studies
- ▶ Paid Fee
- ▶ Provided Document
- ▶ Signed Contract
- ▶ Study programmes
- ▶ Study programmes group / faculty
- ▶ Student admission period
- ▶ Ranking list
- ▶ Master data

Data Entity - Applicant

- ▶ **Demonstrate the document!**

Master data

- ▶ List of subjects that applicants sit as part of the high school final year exam
 - ▶ List of faculties
 - ▶ List of study programmes
 - ▶ List of contract types
 - ▶ List of document types
 - ▶ List of fees payable by applicants
 - ▶ List of High schools
-
- ▶ Appendix A contains the data for some of the above tables
 - ▶ Integrate the master data from all modules!

Overview

- ▶ **Functionality**
- ▶ **Work done**
- ▶ ***To be done***

To be done

- ▶ Clean up the document
- ▶ Check the consistency between sections
- ▶ Completing the enrolment of new students
 - ▶ How is it done
 - ▶ Integration with CMF

Some notes and thoughts

- ▶ Rules and procedures at private universities are subset of rules and procedures at public universities
- ▶ Have we taken into account specifics of other faculties????
- ▶ Non-functional requirements should be moved into a separate document (demonstrate document)
- ▶ User Interface should be defined through data model and guidelines on usability, design etc.
 - ▶ The software shall offer user-friendly interfaces, present default options, present windowed screens with menu bars, option and command buttons, toggle commands, dialog boxes, scrolling windows and drop-down list boxes for option and instance displays. Mouse-assisted and keyboard shortcut key commands shall be featured. The technical evaluation of the software will take into account qualities such as robustness, availability, flexibility, maintainability and general adequacy for distributed processing and Web operability.
- ▶ Configurability is a huge issue due to the differences between universities!

Thank You!