European Commission
TEMPUS

Innovation and Knowledge Management towards eStudent Information System – iKnow

Self-evaluation report

Dejan Gjorgjevikj, Ana Madevska Bogdanova
University “Sts. Cyril & Methodius”
Faculty of Computer Science and Engineering

Objective

- The general objective of this report is to
  - evaluate the current state of the iKnow system
  - point out the functionalities that are successfully implemented
  - point out the functionalities that are not fully implemented (yet) – and why
    - What was dropped out
    - What was postponed
    - What will be reimplemented

Overall implementation status

- ✓ Enrolment module
- ✓ Core module (CM)
  - ✓ Administration (University and faculty)
  - ✓ Students online services
  - ✓ Student services department, overall student data processing
  - ✓ Teachers module
- ✗ Existing student’s data migration into iKnow
- ✗ Automatic migration of candidates from the enrolment into the core module

Enrolment module

- Candidates enrolment wizard
- Enrolment committee manual forms and candidates processing
- Ranking modules
- Enrolment results publishing

CM Administration (University and faculty)

- Users management
- Study programs management
- Courses management
- LMS integration (Moodle)

CM Students online services

- Semester enrolment
- Exam application
- Courses selection
- Documents and certificates request
- Grades overview
- Diploma thesis management
- Payments processing

CM Teachers module

- Enrolled students listing
- Grades management

Existing student’ s data migration into iKnow

… delayed because of

☒Users migration issues (CAS into iKnow membership)

☒Incomplete current data structures

☒Logistics of updating outdated study programs

General software quality impression

- Generally accepted as:
  - quite usable,
  - easy to learn and work on
- Agile development
- Procedures and the user interface – remodeled
- Adaptations - teachers and student services employees
- Students – confirmed the high usability

Detailed evaluation of the enrolment module

- **Enrolment module overall status**
  - Some faculties completely relied on the iKnow system
  - Others used it in parallel with legacy systems, achieving identical results
  - Several existing bugs - corrected
- **The second and third cycle** - The university changes the rules each year
  - prone to changes each year
  - adaptation into a universal and flexible solution is active

Enrolment module - Functional requirements

- **Student interfaces**
  - Students are generally satisfied
  - Very few mails sent from students to the support team
  - Most mails about data inconsistences
  - Almost no mails about system features and usability

Enrolment module - Functional requirements

- **Application processing, ranking and reporting**
  - Major issue - second and third term of enrollment had different rules
  - Two interfaces to process applications:
    - wizard
    - a short form
  - The main ranking - extensively tested and no inconsistencies
  - The reporting - generally satisfactory

Enrolment module - Functional requirements

- **Application and management functionality**
  - The management of master data – functional
  - Most of the data – preimported by the developers
  - Adding new records in the master tables
    - Was rather slow and demanded more steps
    - Much better interface is expected to be implemented later

Enrolment module – Non-functional requirements

- **Usability and response times**
  - Response time during testing phase - demanded several optimizations
  - Response time during enrolment process - satisfactory response times
  - The most frequently used forms - were optimized for best usability
- **Documentation and support**
  - Extensive and detailed materials
  - Multiple training sessions
- **System stability and reliability**
  - Some stability problems due concurrent transaction - rarely noticed
  - Updates in the new version to overcome the problem

Enrolment module – Organizational issues

- **Changes in administrative procedures**
  - The enrolment process is regulated by laws and regulations issued by the government and the university
  - Practical implementation & final responsibility - faculties
- **Status of archiving**
  - The database of the enrolment term - backed up
  - Expected: online archiving in the second version
- **Other affected processes in the institution**
  - Candidate’s signatures on official paper enrolment applications
  - Paper proof of payments needed
- **Measure of success**
  - Using social media : FB, twitter,…

E-Student Services System overall status

- Most functionalities are fully implemented
- Few are dropped (as irrelevant and an unnecessary burden to the system)
- Several are pending (mostly reporting modules)

ID	Functionality	Description	Implementation
[1]	Administration of students data	Enable administration or only preview of the students personal data. Data needed for reports, statistics, contacts and guidance during studies.	Implemented
[2]	Administration of students enrolment details	Enable administration of the parameters of a student at the time of enrolment (initial program, tuition size etc.)	Implemented
[3]	Administration of the student semesters	Sign in new semesters for the student in question, change programs, tuition group – amount. Close completed semesters	Implemented
[4]	Administration of students courses	Enter the student's choices for elective and mandatory courses for the signed semester in question. Calculate credits for the semester based on the courses chosen.	Implemented

ID	Functionality	Description	Implementation
[5]	Administration of student exam applications	Enter student exam applications or manage student exam applications entered online.	Implemented
[6]	Administration of student’s exams	Insert or update grades achieved by the student on a particular exam. Review list of exams passed.	Implemented
[7]	Administration of seminar thesis	Insert or update details of the seminar thesis undertaken by the student	Not Implemented. Decided that it is irrelevant and an unnecessary burden to the system. Mandatory seminar thesis are solved as any other course.
[8]	Administration of diploma thesis	Update details for the diploma thesis that the student works on	Implemented

ID	Functionality	Description	Implementation
[9]	Administration of exam terms	Add new exam terms when exams are scheduled	Implemented. Responsibility divided among the university and faculty administrator
[10]	Administration of semesters	Create new semesters with appropriate list of activated courses eligible for taking. Join courses with professors for the semester in question	Implemented
[11]	Administration of quotas and tuition prices	Inserting and updating tuition details for various quotas that students can enrol in, costs for different programs and quotas.	Implemented
[12]	Administration of courses in programs	Make relations between programs and courses that can be taken in the program in question	Implemented
[13]	Administration of courses	Insert or update details for each course, ECTS credits, fall/spring semester, number of classes	Implemented

ID	Functionality	Description	Implementation
[14]	Administration of programs	Insert/update details for the study program that a student can enrol in (start year, bachelor/master/PhD level etc).	Implemented
[15]	Administration of program revisions	Insert revisions of program	Implemented
[16]	Administration of members of the faculty	Insert new employees, update details for the current employees / faculty members	Implemented
[17]	Diploma thesis report	Present a list of completed on-going diploma thesis work	To be Implemented
[18]	Exams report	Present results from exams with various filtering options	To be Implemented
[19]	Graduated report	Present list of graduated students	To be Implemented
[20]	Elective courses report	Present a list of students per course per semester	Implemented

ID	Functionality	Description	Implementation
[21]	Master book	Prepare the report for printing the master book requested by the ministry of education	To be Implemented
[22]	List students per program	Present list of students with various filtering options	Implemented
[23]	Seminar thesis report	Present list of seminar thesis	Not implemented
[24]	Administration of student mobility	Insert / update details for student exchange programs the student has participated in.	To be Implemented
[25]	Administration of Master thesis	Update details on the Master thesis the student is working on	Implemented
[26]	Issuing documents	Issuing various documents for the student (list of courses and grades, proof of enrolment, etc)	Implemented
[27]	Issuing news and announcements	Post news and announcement or mail bulk mails to students	Implemented

ID	Functionality	Description	Implementation
[28]	Administration of financial services	Create and manage various services with financial implications	Implemented
[29]	Administration of lookup data	Administration of cities, countries, municipalities and other general data used in various modules in the system	Implemented
[30]	Administration of class schedule	Management of classes, rooms and groups of students	To be Implemented
[31]	Administration of faculties	Administration of the faculties in the university	Implemented
[32]	Administration of student sign outs	Administration of the students leaving the university	Implemented
[33]	Retrieval of enrolment data	Retrieve data from the enrolment module when accepting a new student at the university	To be Implemented

ID	Functionality	Description	Implementation
[34]	Establish AAI infrastructure	Establish the necessary infrastructure for supporting the AAI	Implemented, with MS .NET membership
[35]	Maintain AAI infrastructure	Maintain the central point of the infrastructure, LDAP schemas, and registry of member institutions, software repository.	Implemented, with MS .NET membership
[36]	Support the member institutions	Provide support to FCC to establish the identity providers and resource providers	Implemented, with MS .NET membership
[37]	Manage users identities	Open accounts for students and faculty staff, manage levels of authorization, revoke accounts	Implemented, with MS .NET membership
[38]	Manage resource access	Define access rights for own resources with respect to user roles and responsibilities	Implemented, with MS .NET membership
[39]	Synchronize AA data with Core module data	Maintain synchronized student records between LDAP directories and core modules	Not implemented

ID	Functionality	Description	Implementation
[40]	Access student resources	Using single credential, access all the needed resources.	Implemented
[41]	Access faculty resources	Using single credential, access all the needed resources.	Implemented
[42]	Staff and students mobility	Access resources in visiting institutions using home identities	Not Implemented
[43]	Presence identification	Using RFID or other means, identify physical presence of students	To be Implemented
[44]	Issuing of personal ID cards – student ID	Take pictures, compose layout and print student e-ID cards	To be Implemented
[45]	Maintain student records	Manage the students' records.	Implemented.
[46]	Track classes attendance	Using the schedule and the presence records, track the attendance to lectures	To be Implemented
[47]	Establish mailing lists	Using the data about the course enrolment, establish appropriate mailing lists	Implemented

ID	Functionality	Description	Implementation
[48]	Get financial info for students	Check the payments from students for their invoices for services	Implemented
[49]	Generate invoice for service	Generate invoices based on the fees for each type of service (semester, exam, penalties, …)	Implemented
[50]	Exchange data with LMS	Populate LMS courses, extract grades	Implemented
[51]	Information about books recommended by a course	Borrow a book recommended by a course or other books from the same author that ULS possess	Not Implemented
[52]	Manage ULS resources	Define rules for returning the borrowed books in different administrative situations.	Not Implemented
[53]	Acquire HR information	Get information for the needs of the HR systems, on the engagements of the staff	To be Implemented

ID	Functionality	Description	Implementation
[54]	Customizing the user types, layout of the reports and the module accessibility	The administrator should have opportunity to customize the layout of the reports. Based on the university preferences, every university should be able to fully customize which reports will be accessible from the module	To be Implemented.
[55]	Manage look-up tables	The administrator should have opportunity to manage the look-up tables used in the report process (exam report form, exam application form, …)	To be Implemented
[56]	Create/edit/delete exam reports and notifications	Faculty staff should be able to create/edit/delete exam reports that he/she has created. Reports should be signed with digital signature.	To be Implemented
[57]	Fill exam applications	Faculty staff should fill in the exam applications of the students who applied for his/her course exam with the appropriate details	Implemented

ID	Functionality	Description	Implementation
[58]	View student's data/ reports on student attendance	Faculty staff should be able to view student's personal information, additional skills, certificates, course enrolment	Implemented
[59]	Student registration	Students register on the system by entering their university e-mail. After the account activation the student can use the system	Implemented
[60]	Viewing completed activities	Students should be able to have overview of his/her activities status.	Implemented
[61]	Filling/updatin g personal data	Students can fill their personal data. The data involve basic information and additional skills, activities and obtained certificates	Implemented
[62]	Registering/ Viewing course information	Students should be able to register to a course with password given by the specified professor.	Implemented

ID	Functionality	Description	Implementation
[63]	Filling complaint/ question	Students should be able to rise complain about an error in his/her data in the reports. Students should be able to address administration staff about unclear questions.	Implemented
[64]	Request diploma/ certificate	Students should be able to request issuing of diploma or certificate by the Administration Office.	Implemented
[65]	Ranking the students	The system should be able to automatically rank all the students for every study program based on their average and cumulative GPA	Implemented
[66]	Produce various statistical reports for existing students	The system should produce various statistical reports for all existing students based on their average and cumulative GPA	To be Implemented

ID	Functionality	Description	Implementation
[67]	Generate various statistical reports for applied / admitted candidates	The system should generate various statistical reports for all candidates that have applied, for the accepted and rejected ones	To be Implemented
[68]	Generate various statistical reports for students	The system should generate various statistical reports for all students	To be Implemented
[69]	Generate various statistical reports for faculty staff	The system should generate various statistical reports for faculty staff	To be Implemented
[70]	Generate various statistical reports for university administration (Office members)	The system should generate various statistical reports for office member of university administration	To be Implemented

ID	Functionality	Description	Implementation
[71]	Generate various statistical reports for supporting personnel.	The system should generate various statistical reports for supporting personnel	To be Implemented
[72]	Audit log	The system should keep audit log with time stamp and user ID of all changes taking place in the Reporting module	Implemented, interface for the log pending
[73]	Reports for the Ministry of Education	Generate reports that are requested by the Ministry of Education	To be Implemented
[74]	Reports for the Statistical Office	Generate reports that are requested by the Statistical Office	To be Implemented
[75]	XML Web services for the data of points 24 and 25	The system must enable web services in order to be able to access the data in points 24 and 25	To be Implemented
[76]	Alumni Site	Alumni site with CMS capabilities to follow the career of the alumni students	To be Implemented

E-Student Services System – Overall status

- **76** functional requirements
  - Implemented – **46** functional requirements
  - To be implemented – **24** functional requirements
  - Not implemented – **4** functional requirements
  - Not implemented and irrelevant – **2** functional requirement

European Commission
TEMPUS

E-Student Services System – Open issues

- Open issues in the implementation process
  - Reporting modules are not completed due to other modules higher priority and changes
  - Seminar thesis management – left to LMS (Moodle)
  - Student mobility module – to be specified more precisely

Practical experiences so far

- Various faculties and universities have different regulations for the functionalities
  - yielding in conflicting requirements that need to be balanced and solved
- Necessary fine tuning of the interface and functionalities
  - feedback from the users (all categories) using the software

E-Student Services System – Implemented functional requirements

- **Module for study programs and schedules**
  - Defining study programs, courses, prerequisites and rules for studies
    - Focus on making this interface easy to use
    - Excel import with simple predefined templates was provided
    - Multiple different forms with the same functionality are provided
  - Student activities module
  - Mapping of faculty staff to courses
  - Equivalence of courses, modules and programs

E-Student Services System – Implemented functional requirements

- Student activities module
  - Enrolment in a semester and selection of courses
    - Both students can do it by themselves or the student services can do that for them
    - Forming course groups
- Module for the administration
  - Administration of faculties and accredited study programs
  - Administration of classrooms, rooms and laboratories – fine tuning needed
- Module for administration of academic results
  - Administration of earned ECTS credits and grades from exams passed

E-Student Services System – Implemented functional requirements

- Module for personal identification and access control
  - Currently, MS .NET membership access control is implemented, integration with CAS is planned
- Module for personal records of students
- Module for presence monitoring and student activities
  - Currently, teachers can enter the attendance of each student in the system
  - Automated monitoring using student RFID cards is planned

E-Student Services System – Implemented functional requirements

- Module for electronic payment and use of resources
  - Administration of payments by the students
    - Automatic retrieval of SMS payments for administrative tax for students is implemented
    - Other payments are entered manually by student services department
  - Administration of the use of learning systems (LMS)

Conclusion

- Most of the planned modules / activities are successfully implemented / carried out
- The system is generally accepted as quite usable and easy to learn and work on
- Some of the activities were postponed
- The enrolment module is undergoing a redesign and reimplementation
- Some of the functionalities are currently under development

Thank you for your attention.