Information Systems Department
Faculty of Computer and Information Sciences
Mansoura University

Optimizing Service Oriented Architecture to Support e-Learning with Adaptive and Intelligent Features

DISSERTATION
Submitted in Partial Fulfillment of the Requirements for the Degree
Doctor of Philosophy in Information Sciences

TO
Department of Information Systems, Faculty of Computer
and Information Sciences, Mansoura University

BY
Haitham Abdel Monem El-Ghareeb
M.Sc., Information Systems Department
Faculty of Computer and Information Sciences
Mansoura University

SUPERVISED **B**Y

Prof. Dr. **Alaa El-Din Mohamed Riad**
Vice Dean for Education and Student Affairs
Faculty of Computer and Information Sciences
Mansoura University

Dr. **Hazem M. El-Bakry**
Information Systems Department
Faculty of Computers and Information Sciences
Mansoura University

Mansoura – 2011

بِسْمِ ٱللَّهِ ٱلرَّحْمَـٰنِ ٱلرَّحِيمِ

هُوَ ٱللَّهُ ٱلَّذِى لَآ إِلَـٰهَ إِلَّا هُوَ ۖ عَـٰلِمُ ٱلْغَيْبِ
وَٱلشَّهَـٰدَةِ ۖ هُوَ ٱلرَّحْمَـٰنُ ٱلرَّحِيمُ (٢٢) هُوَ ٱللَّهُ
ٱلَّذِى لَآ إِلَـٰهَ إِلَّا هُوَ ٱلْمَلِكُ ٱلْقُدُّوسُ ٱلسَّلَـٰمُ
ٱلْمُؤْمِنُ ٱلْمُهَيْمِنُ ٱلْعَزِيزُ ٱلْجَبَّارُ ٱلْمُتَكَبِّرُ ۚ
سُبْحَـٰنَ ٱللَّهِ عَمَّا يُشْرِكُونَ (٢٣) هُوَ ٱللَّهُ
ٱلْخَـٰلِقُ ٱلْبَارِئُ ٱلْمُصَوِّرُ ۖ لَهُ ٱلْأَسْمَآءُ
ٱلْحُسْنَىٰ ۚ يُسَبِّحُ لَهُۥ مَا فِى ٱلسَّمَـٰوَٰتِ
وَٱلْأَرْضِ ۖ وَهُوَ ٱلْعَزِيزُ ٱلْحَكِيمُ (٢٤)

صدق ٱللهِ العظيم

سُوْرَةُ الحشر

In the name of Allah, the Beneficent, the Merciful

He is Allah, than Whom there is no other God, the Knower of the Invisible and the Visible. He is the Beneficent, Merciful. (22) He is Allah, than Whom there is no other God, the Sovereign Lord, the Holy One, Peace, the Keeper of Faith, the Guardian, the Majestic, the Compeller, the Superb. Glorified be Allah from all that they ascribe as partner (unto Him). (23) He is Allah, the Creator, the Shaper out of naught, the Fashioner. His are the most beautiful names. All that is in the heavens and the earth glorifieth Him, and He is the Mighty, the Wise. (24)

God Almighty has spoken the truth

Surah Al-Hashr

To my Mother, the one who taught me to be frank, bold, simple, honest, pure, and always believe in God

The one with me when I am happy or sad, support me in times good or bad

The fragrance of love, who sought me to make the effort, and do not wait for pay back

The one with the bright smile as sun, can't thank you enough for what you have done

Acknowledgments

First and foremost, I thank **<u>ALLAH</u>** for giving me incentive to go on and finish this work and giving me glimpses of heavenly hope in the darkest times.

I would like to thank my supervisor, **Professor Alaa El-Din Mohamed Riad**, Vice Dean for Education and Student Affairs, for his unlimited support and encouragement, for providing me with valuable insights and encouraging me to always do better, for keeping my thesis on track and always giving me great and valuable ideas to enhance my work.

I would like to thank my supervisor, **Dr.Hazem M. El-Bakry** for the insightful discussions, and ideas, for taking long times to discuss smallest details.

I would like to thank **Dr.Hamy El-Minir**, head of communications department of the Misr High Institute of Engineering for his support and motivation from the early stages of this work.

To **My Father** and **Mother**, **My Beloved Brothers** ***Mohamed and Hesham*** and **The Sweet Sister** ***Mariam,*** you are the people whom without I couldn’t have continued the journey to the day I witness this dissertation alive. Thanks for always being there to support me, every single moment, with everything you can, when I was about to lose hope and trust in me being able to continue through the darkness, trying to reach the goal through long nights of mess, you were there,

through long distances you were pushing me forward by your support, encouragement, guidance, and help. Thanks to ALLAH, I'm really blessed with amazing family like you. I Love You all!

Last but not least, I would like to thank people who are not waiting for my thanks, who are keeping me up and pushing me forward, who are wishing me the best, praying for me, sacrificing everything they have, and giving everything they can just to see me where I am right now, **my family,** and **my friends**.

Haitham A. El-Ghareeb

October 2011

Abstract

E-Learning solutions face challenges that are leading a group of researchers to doubt the efficiency of such an experience. Challenges include lack of adaptation in the e-Learning process. Service Oriented Architecture (SOA) - as a design pattern presenting systems as collection of reusable services that can be exposed and consumed on the Internet with standard interfaces - has many advantages that can be achieved on technical, managerial and implementation aspects. It can be useful in integrating adaptive and intelligent features in e-Learning systems.

However, utilizing SOA in e-Learning faces challenges. Those challenges can be categorized into two categories: pedagogical and technical. Pedagogical challenges are related to the activities and the learning path of the e-Learning process itself. Technical challenges are related to the way system architects design and build e-Learning systems. Optimization is a continuous process that happens iteratively and recursively to achieve better:

- Addressing and understanding the problems in hand
- Presenting different proposed solutions
- Optimizing the selected solution
- Applying the optimized and enhanced solution
- Evaluating the applied solution

Dissertation addresses pedagogical and technical challenges facing the e-Learning process and presents an Adaptive e-Learning Model as a solution to the highlighted problems. Pedagogical solutions include new adaptive features presented to: students based on their learning preferences, and to instructors to empower them through e-Learning. Technical solutions include optimizing newly presented features. Presented model uses SOA in integrating major e-Learning components together, side by side with adaptive and intelligent features. e-Learning Components include:

- Adaptive Learning Management System (LMS): includes
  - Student Learning Subsystem
  - Student Learning Profile Subsystem
- Quality Assurance and Accreditation Project (QAAP) Management System: includes
  - Course Specifications Module
  - Instructor Time Table Module

- Exam Management System: includes
  - Exam Data
  - Exam Application
- Learning Content Management System (LCMS): includes
  - Questions
  - Learning Objects (LOs)

Presented Adaptive e-Learning Model includes different intelligent services that use fuzzy logic as an intelligent technique to empower it. Presented Model proposes nine intelligent services utilized in different e-Learning functionalities. Intelligent services are grouped into two categories based on their aims: Instructor Services and Student Services. Instructor Intelligent Services are:

- Intelligent Learning Objects (LOs) Classifier: combines both supervised and non-supervised learning algorithms with fuzzy logic to classify LOs.
- Intelligent Online Lecture LOs Advisor: enables the Adaptive Online Lecture Model to present different pedagogical aspects via:
  - Recommending LOs based on students’ learning preferences.
  - Involving students in the learning process from the very early beginning of the lecture.
  - Preparing for the next lecture, so students feel the lecture’s adaptivity.
- Intelligent Student Performance Tracker: intelligently keeps track of students’ learning profiles and preferences and verifies completeness.
- Intelligent Cheating Depressor: intelligently identifies students cheated during online assessments based on their history, consumed assessment duration, and scored marks.

Student Intelligent Services are:

- Intelligent Time-to-Learn Topic Calculation: guides the students to the time needed to learn topics. This service helps students come over the variance between the time they expect to learn, and the time they actually take to learn.
- Intelligent Study Plan Advisor: guides students through branching decisions in courses based on previous study plans, learning profiles and preferences of current students.

- Intelligent Agenda Study Time Planner: helps students identify study times and control interruptions to improve performance. It integrates with available activities to update students' calendar automatically.
- Intelligent Meeting Manager for Suspended Students: intelligently schedules a meeting with one of the instructors for suspended students.
- Intelligent LOs Recommender: intelligently recommends LOs to satisfy courses' prerequisites that are not fulfilled by students, and to suggest further readings for students based on their learning preferences.

Optimizing the presented SOA based Adaptive e-Learning Model and intelligent services is achieved on: Model level, Architectural Level, and Service level. Model level optimization included presenting new adaptive and intelligent features, and using them innovatively. Architectural level optimization included the shift from traditional methods of building SOA based e-Learning systems to decrease the granular level of services, and utilizing new Web services technologies. Service level Optimization is experienced through the optimization of Intelligent LOs Recommender, as it was the bottleneck of the presented Adaptive e-Learning Model identified during implementation and deployment.

Optimized Adaptive e-Learning Model and intelligent services are evaluated from three perspectives: User Satisfaction, Performance, and Information Retrieval. User Satisfaction is evaluated from both instructors and students points of view using surveys and reviews of the presented Adaptive e-Learning Models. Performance perspective of the presented model focuses on the effect of identifying the presented model's bottleneck through experimenting the system, and presenting a solution to overcome this bottleneck, optimizing the solution, and applying it. The performance bottleneck is identified at the Intelligent LOs Recommender service. Information Retrieval (IR) of the presented Adaptive e-Learning Model is an important aspect as the presented system has to handle large amounts of LOs. Presented system IR capabilities are measured by the three IR measures: Precision, Recall, and F-Measure. Presented Adaptive e-Learning system shows enhanced IR performance measures. The capability of Intelligent LOs Classifier on classifying different LOs was examined. Intelligent LOs Classifier shows high classification rates on testing set.

Table of Contents

List of Figures

List of Tables

Chapter 1

Chapter One

Overview

1.1 Introduction

Adaptive e-Learning that is supported by intelligent techniques and methods is one of the ways to support personalized e-Learning, and so it is the way to overcome many of the limitations and solve today's e-Learning challenges. Different adaptive and intelligent e-Learning systems proposed over a long e-Learning research will be reviewed, highlighting the importance of adaptive and intelligent features in e-Learning, and the attempt to introduce services based e-Learning systems in order to overcome many of the technical e-Learning challenges, like reusability, scalability, integration, and interoperability. There is a need to present a new learning model that attempts to solve different challenges. Though there's no single unified learning model that can be the only right model, hopefully this work will be a step towards a better learning model supported with the appropriate adaptive and intelligent features, methods, and Service Oriented Architecture (SOA) technologies.

Integrating UMIS and LMS is a need to achieve better e-Learning systems. Different software architecture patterns can be utilized in integrating both systems. Studying and analyzing many of the available software architectures has led to the realization of efficiency and effectiveness of utilizing SOA, however it has led to challenges. Challenges include lack of performance when transmitting large amount of data. Web Service Software Factory design pattern as an example of SOA design patterns that builds everything within systems as Web services will yield to big performance degradation to proposed integrated e-Learning systems.

Most of the AI applications have not yet been expanded to or adopted in widely used e-Learning systems, especially open-source systems such as Moodle and Sakai. Current intelligent LMS systems are still in their early stage, while AI applications need to handle some problems or to be modified before applying them into the LMS systems, and AI technology also needs to be brought to open source communities. Presented Adaptive e-Learning Model integrates different intelligent features within the system to empower the presented model.

1.2 Research Motivations

Optimization begins with addressing pedagogical and technical problems in hand. Pedagogical problems are addressed and optimized through the presented adaptive e-Learning models. Technical problems are addressed and enhanced through combining Business Process Management (BPM) and Service Oriented Architecture (SOA) in a layered model to overcome technical, composition, and integration challenges. With the maturity of presented Adaptive e-Learning Models, it came the time to implement them. However, performance challenges appeared, mainly with the Intelligent LOs Recommender, as it deals with tremendous amount of updatable data and LOs on the internet. Intelligent LOs Recommender lacked some performance challenges in the first place due to the tremendous amount of generated keywords that affect processing and insertion times. Including Course Objectives as an input parameter in the keywords Term Frequency processing optimizes the performance by focusing on the important and needed keywords instead of wasting the processing time and storage spaces for non-important keywords.

Pedagogical and technical problems are the basis of this dissertation. Pedagogical problems are touched by the researcher through a pilot study, and technical problems are identified through evaluation of previous SOA based LMS.

1.2.1 Pedagogical Problems

Later in the year 2008, pilot study for fourth year Information Systems department students at the faculty of Computers and Information Sciences in Mansoura University, Egypt was conducted by the researcher. Although the 57 students who conducted the questionnaire answered that they heard about e-Learning, and they all believe it is effective, almost 25% of the participating students don't use Internet as their main source of information. This is considered weird when compared to the 100% e-Learning efficiency commit.

Table 1.1 provides the following information:

- 83% of students classified themselves as good Internet users, while 72% of them do not even know what “tutorials” are. Web based tutorials is one of the main sources for teaching students.
- Although 83% of participants believe they can learn via e-Learning, and almost the same percentage agree to participate in an e-Learning, only 68% of students appeared to be accepting the idea of using mobile in learning purposes.
- Though Mobile Learning (M-Learning) is an important e-Learning research field, the study shows that only 28% of students access Internet via their mobile phones, and only 46% of students are willing to participate in a free of charge mobile learning experience when asked to.
- 61% of the students said that they will not participate in a mobile assessment experience when they are asked to.
- 58% of the students believe that M-Learning will not become popular.
- Finally, all students participated in the pilot study agreed that the authors’ faculty utilizes different forms of e-Learning.

Though the faculty of the researcher does not provide an official site for e-Learning, any online courses, assessment site or any other form of e-Learning other than the researcher's attempts, students still believe that e-Learning is efficient - even if they have not experienced it at all. Students believe that e-Learning is effective without trying it, and are not excited about M-Learning regardless to the researches that the e-Learning researchers provide. There are lots of contradictions in students' responses that forces the researcher believe that students haven't experienced real e-Learning experience due to backgrounds issues, lack of personalization and flexibility, and lack of Internet access in many scenarios.

Table 1.1: Pilot Study for Fourth Year Students

	Strongly Agree	Agree	Disagree	Strongly Disagree
Internet is main source of Information	38%	38%	24%	
Good Internet User	25%	58%	17%	
Familiar with Tutorials	14%	14%	53%	19%
Students Can Learn via e-Learning	23%	60%	17%	
Participate in e-Learning	18%	68%	14%	
Use Mobile Learning	12%	56%	32%	
Access Internet via Mobile Phones	7%	16%	49%	28%
Participate in M-Learning Experience	9%	37%	54%	
Participate in M-Assessment Experience	7%	32%	61%	
M-Learning will become Popular	5%	37%	58%	

1.2.2 Technical Problems

Mansoura University runs its in house developed and deployed University Management Information System (UMIS) for more than a decade. UMIS has reached a stable and mature state when compared to the newly introduced LMS. To adopt LMS functionalities in the University without making LMS and UMIS isolated, both need to interoperate to enable university achieve managerial and educational tasks. Researcher designed, tested, deployed, and evaluated such a solution and came out with noticeable evaluation results. From technical perspective, quality parameters like performance shall be addressed. SOA based systems rely heavily on messaging and add extra headers to manage requests and responses in standard format. Headers affect directly the amount of data transferred over the network. Headers sizes differ according to the number of records to be handled, and differ from application to another, so network performance differs. However, the comparative study yielded the fact that network delay is highly affected by the header size and such delay cannot be neglected. Network delay differences in range exceed 400% between both implementations. Optimization of SOA utilization in e-Learning systems is needed to address and overcome those challenges.

1.3 Dissertation Objectives

Dissertation objectives focus on optimizing e-Learning through an iterative and recursive optimization process that starts by identifying different types of problems, presenting solutions, evaluating presented solutions, optimizing presented solution, and applying the optimized solution. Objectives include:

- Address Pedagogical Problems facing e-Learning.
- Track e-Learning Systems evolution.
- Evaluate the importance of Services based e-Learning Systems and the role of Service Oriented Architecture (SOA) for such systems.
- Address Technical Problems resulting from adopting SOA in e-Learning
- Address both students and instructors’ needs from e-Learning
- Present Adaptive e-Learning Models for both students and instructors as a solution to current pedagogical problems
- Design and build the presented Adaptive e-Learning Models
- Present innovative ways of utilizing intelligent techniques in e-Learning
- Empower Adaptive e-Learning Models with Intelligent Services
- Address the bottlenecks of the presented built Models

- Optimize presented solutions to overcome identified bottlenecks
- Evaluate optimized solution from different perspectives

1.4 Contribution of Dissertation

Dissertation combines pedagogical and technical aspects of e-Learning to achieve enhanced e-Learning. Focusing either on pedagogical or technical issues alone will not present the full screen of e-Learning. Besides, researcher addressed real world e-Learning problems that affect students and touches instructors, and presented solutions to current challenges. Dissertation presents a new Adaptive e-Learning Model that blends instructor lead education with e-Learning capabilities. Presented Model focuses on the two elements of the e-Learning: Students, and Instructors.

Dissertation presents innovative new intelligent features to improve students' and instructors' performance and enhance presented Adaptive e-Learning Model. Fuzzy Logic utilization in different services as a technique to present intelligent features is highlighted in different aspects. SOA is the architecture used all over the system. Presenting adaptive and intelligent features as services with standard interfaces will allow different e-Learning systems to adopt them, so they will be reusable.

Dissertation presents optimization and evaluation techniques of presented Adaptive e-Learning Model and Intelligent Services, highlighting the effects of optimization decisions taken, and opens the door for future work related to e-Learning.

1.5 Dissertation Structure

Chapter One: Presented an overview of the dissertation, objectives, contribution, and addressed the research problems.

Chapter Two: Introduces a closer look on e-Learning challenges through reviewing current e-Learning status; the shift from traditional e-Learning systems to Services based Adaptive and Intelligent e-Learning systems. This chapter also focuses on highlighting the importance of utilizing SOA in e-Learning Systems and presenting the technical challenges facing this utilization. This part presents results from SOA utilization systems.

Chapter Three: Introduces the Proposed Adaptive e-Learning Model, raising the importance of SOA in integrating different e-Learning systems and components. Desktop applications are used to overcome Internet access challenges. Pedagogical solutions focus on introducing adaptive and intelligent features in e-Learning to reach personalized environment.

Chapter Four: Introduces the Intelligent e-Learning services that empower presented Adaptive e-Learning Models using fuzzy logic as one of intelligent techniques. Nine intelligent services are presented. Intelligent services are grouped into *Instructor Services* and *Student Services*.

Chapter Five: Presents closer look on implementation details of developing, building, and deploying the presented Adaptive e-Learning Models and composing services. Technical details include database tables, class diagrams, packages, and interface design.

Chapter Six: Highlights evaluation of the proposed Adaptive and Intelligent e-Learning Model Components and Services. Presenting adaptive and intelligent features as services with standard interfaces will allow different e-Learning systems to adopt them, so they will be reusable.

Chapter Seven: Presents Conclusion and Future work.

Dissertation ends with References.

Chapter 2

Chapter Two

e-Learning Pedagogical and Technical Problems

2.1 Introduction

Current Learning Model Paradigm faces many pedagogical and technical challenges. Pedagogically, Educational psychologists agree that students differ in the ways they learn and very few teachers can adapt learning to each student in the typical large classes. Computer-based learning systems are criticized by many researchers for their limited ranges and adaptability of teaching actions compared to rich tactics and strategies employed by human expert teachers. Many universities in the developing countries, started to adopt e-Learning by modifying network infrastructure, establishing new labs, providing internet connection, and purchasing different tools for creating e-Learning courses and using different LMSs. However, these modifications and supplement were not enough to ensure successful e-Learning outcomes because other important elements for e-Learning success were missing such as flexibility of the system, adaptability towards students' needs, reusability of learning objects, interoperability between LMSs, effective and official design of e-content. SOA as a design pattern that has achieved technical advantages in integration, interoperability, and many other aspects faces performance challenges that can be enhanced.

This chapter presents Pedagogical problems, and the Technical problems, facing e-Learning opening the door for closely identifying the problems in hands, and beginning the road for Optimizing SOA to Support e-Learning with adaptive and intelligent features.

2.2 Pedagogical Problems

E-Learning can be thought of as the learning process created by interaction with digitally delivered content, services, and support. E-Learning involves intensive usage of Information and Communication Technology to serve, facilitate, and revolutionize the learning process [1-3]. Learning methods include traditional learning “face-to-face”, distance learning “complete asynchronous time and place learning delivery; mainly online”, and blended learning “learning that combines instruction lead learning with online learning activities leading to reduced classroom contact hours”. Blended learning has the potential to increase student learning while lowering attend rates compared to equivalent fully online courses [4]. Blended learning is the learning paradigm that attempts to optimize both traditional learning and distance learning advantages, potentials, and benefits while eliminating both learning paradigms shortages and challenges. When compared to traditional learning paradigm, blended learning is found to be consistent with the values of traditional learning paradigm adopted in almost all higher education learning institutions for decades, and has the proven potential to enhance both the effectiveness and efficiency of meaningful learning experiences [5].

Learning Management System (LMS) is the software that automates the administration of education. LMS registers students, tracks courses in a catalogue, records data from learners, and provides reports to management. LMS is typically designed to handle courses by multiple publishers and providers. It usually doesn’t include its own authoring capabilities; instead it focuses on managing courses created by a variety of other learning resources. Prototypical LMS is presented in [6]. Technology and the great advancement in recent Web 2.0 and informal learning methods allowed the existence of complete education programs and courses to be presented online.

This part reviews the different adaptive and intelligent e-Learning systems proposed over a long e-Learning research era, highlighting the maturity of adaptive and intelligent features and the e-Learning researchers attempt to introduce services based e-Learning systems in order to overcome many of the e-Learning challenges, like reusability, scalability, integration, and interoperability. This chapter concludes with the need to present a new learning model that attempts to use the best of what was presented before, and avoid all the challenges and mistakes.

Current Blended Learning Model Paradigm faces many pedagogical challenges. Today's technologies advance by decades technologies used in teaching and education less than decade ago. However, a number of recent articles have commented that science education is no better today than it was fifty years ago. The National Assessment of Education Progress (NAEP) shows that in most areas today's students are achieving at about the same levels as students tested in 1971 [7]. This pedagogical issue is the result of:

- The attempt to use whatever technology currently available or becomes available in the near future without pedagogically considering student or the learning process.
- Allowing technology to stand against the learning process, because no matter what advancement we have achieved, technology is still limiting our ambitions.
- The poor evaluation that is available for many of the innovations. Most of the required evaluations are either inadequate or doesn't exist at all.

Alfred Bork et al. argue that some of the reasons why technology has not led to improvements in learning globally are [7]:

- Grabbing Onto Each New Technology: The belief that each new technology will enhance learning needs more arguments about efficiency than just belief.

- Failure to Continue Successful Development: funders often prefer to look for something new rather than follow up on successful approaches because they want to make a mark by being in the forefront. Funders want to make a statement, and following up on someone else’s work doesn’t provide them with the credit or “name” they desire.

Pedagogically, most training methods and technologies produce, at best, “trained novices”. That is, they introduce facts and concepts to students, present them with relatively simple questions to test this new knowledge, and provide them with a few opportunities to practice using this knowledge in exercise or scenarios. However, becoming proficient requires extensive proactive solving realistically complex problems in wide range situations, combined with coaching and feedback from managers, more experienced peers, or other types of experts [8].

Most evaluations of today’s presented technologies in the learning model focus on technology aspects of the solution while ignoring the pedagogical aspect; almost at all. The result of using technology, particularly computers, in learning has so far not been impressive. A variety of studies and opinions have questioned the use of technology to improve learning. Although it has been many years since computers have begun to be used in learning environments, there is little improvement in learning, with or without technology. Although the use of technology in learning shows no significant difference, that is, computer learning is no better than traditional instruction, learners have been provided with the convenience of any time, any place learning. Students’ understand and retention improves when students learn by experience. Technologies such as collaboration, interactivity, modeling, simulation, virtual reality interfaces, and gaming will help students experience the skill being taught, but they have not helped students that far yet [7]. Besides, students lack of awareness of different e-Learning technologies stand up against the presentation of effective e-Learning model.

Adaptive learning for students with many different backgrounds, learning styles, and interest is almost a must. Educational psychologists by and large agree that students differ greatly in the ways they learn and very few teachers or professors can adapt learning to each student in the typical large classes, the costs associated with delivering different instruction for varied learning styles is prohibitive [7]. Benjamin Bloom (1984) showed twenty-seven years ago, as reported in his 2 sigma paper, that almost all students can learn to the mastery level, given the right learning environment [9]. In Bloom’s experiments, the most successful learning strategy was tutoring. Adaptive e-Learning that is supported with intelligent techniques and methods is one of the ways to support tutoring in e-Learning, and so it is believed that it will be the way to solve many of the limitations and today’s e-Learning challenges.

2.3 Adaptive and Intelligent e-Learning Systems

Computer-based learning systems are criticized by many researchers for their limited ranges and adaptability of teaching actions compared to rich tactics and strategies employed by human expert teachers [10]. Many universities in the developing countries, started to adopt e-Learning by modifying their network infrastructure, establishing new labs, providing internet connection, and purchasing different tools for creating e-Learning courses and using different LMSs. However, these modifications and supplement were not enough to ensure successful e-Learning outcomes because other important elements for e-Learning success were missing such as flexibility of the system, adaptability towards students' needs, reusability of learning objects, interoperability between different LMSs, effective and official design of e-Content [11].

Adaptive e-Learning systems would be a good solution for better e-Learning. The absolute majority of Web-enhanced courses rely on LMS because they are powerful integrated systems that support a number of teachers and students' needs. Though LMSs look surprising, indeed for every function that a typical LMS perform there is an Adaptive Web Based Educational System (AWBES) that can do it much better [12]. Adaptivity is the ability to modify e-Learning lessons using different parameters and a set of predefined rules. Researchers differentiate slightly between adaptivity and adaptability by thinking about adaptability as the possibility for learners to personalize an e-Learning lesson by themselves. These two approaches go from machine centered (adaptivity) to learner centered (adaptability). In practice, it is quite difficult to isolate one from the other due to their close relationship [13, 14]. Adaptive e-Learning is often meant to be new or in an early development stage [10]. Adaptive e-Learning system is the environment of software modules, which comprises a set of features for adaptivity and adaptability [15].

Important factors for adapting to student needs and desires include [7]:

- **Each student should move at a unique pace:** Given all the variations between students' backgrounds, interests, and abilities, it is highly desirable to allow each student to move at a unique pace in the learning units.
- **Adaptation should be very frequent:** Changes based on occasional exams are inadequate. Learning activities should adapt to each student on a moment-by-moment basis. Students should feel that the adaptive program is responding to them as individuals.
- **Each student should be successful in learning:** a major advantage of adaptive variable placing is that the students can continue to learn in a given area until they have learned the material. Almost all learners can succeed and achieve mastery, but some learners need more time and more practice than others.

- **When something is successfully learned, the learner should move on:** Often in classroom learning, after a student has learned something, the class continues working on the topic, boring the student. This will not occur in a fully adaptive learning environment.
- **No one should be taught something he already knows:** By assuring learner competencies, avoiding unneeded instruction, and moving each student forward when ready, students are expected to achieve a major reduction of learning time, but this cannot be verified empirically until there are full range of computer-based adaptive learning units.

The provision of static learning material will not meet the requirements of students. Adaptive e-Learning enables personalizing learning process to individual learners via adapting some parameters; like identifying, analyzing and monitoring relevant aspects of instructions, such as different velocities, paths, or strategies of learning. Performance improvements within the learning process can be gained via adaptive e-Learning systems [15]. Adaptation and personalization will improve the learning process; therefore, a paradigm shift from the consumption of static learning contents to well tailor and highly personalized learning sessions is needed.

2.3.1 Adaptive e-Learning Approaches

Four main approaches which are used to give a historical overview of adaptive e-Learning can be identified [10]:

- **Macro-Adaptive Approach:** Addresses adaptation of instructions on a macro-level by allowing different alternatives in selecting a few main components such as learning objectives, levels of detail, and delivery system. In this approach, instructional alternatives are selected mostly on basis of the student’s learning goals, general abilities, and achievement levels in the curriculum structure.

- **Aptitude-Treatment Interaction (ATI) Approach:** This approach treats adaptation of instructional strategies to specific student's characteristics. This strategy proposes different types of instructions or even different media types for different students. The most important classes of learner characteristics can be summarized with the following ones: intellectual abilities, cognitive styles, learning styles, prior knowledge, anxiety, achievement motivation, and self-efficiency. One aspect of the ATI approach is the user's control over the learning process according to the abilities of the students by giving them full or partial control over the style of the instruction or the way through the course. Level of control can be one of three levels: complete independence, partial control within a given task, and fixed tasks with control of pace.
- **Micro-Adaptive Approach:** Addresses adaptation of instructions by diagnosing the student's specific learning needs during instruction and providing instructional prescriptions for these needs. Researchers have attempted to establish micro-adaptive instructional models using on-task rather than pre-task measures. Monitoring the user's behavior and performance, such as response errors, and response latencies can be used for optimizing instructional treatments and sequences on very refined scale [15].
- **Constructivistic-Collaborative Approach:** Follows the constructivist pedagogical approach. An important element of this approach is the usage of collaborative technologies which are considered often an essential component of e-Learning. Adaptive system enables learning by focusing on how knowledge is learned and should consider the context, learning activities, cognitive structures of the content, and the time extension. Some new adaptive e-Learning systems take account of students' motivational factors combining the instructional plan with a "motivational" plan.

Over the last decades, various types of adaptation systems and possible areas for their applicability have been identified, thus leading to the emergence of specialized research fields, like Adaptive Hypermedia Systems (AEHS), Computer Aided Instruction (CAI), Computer Managed Instruction (CMI), Recommender Systems, Intelligent Tutoring Systems (ITS), Personalized Systems of Instruction (PSI), and many others. Adaptive multimedia systems as an improved learning environment is well documented in the research work of Christian Gutl et al. [15].

2.3.2 Intelligent e-Learning Systems

Artificial Intelligence (AI) utilizes programming algorithms to simulate thought processes and reasoning that produce behavior similar to humans. A successful implementation of AI could be tested using a Turing Test approach, in which a human interacts with an interface that could have either a human or computer on the other end. The test is considered successful if the human is unable to determine whether there is a computer or a human on the other end. The applications of AI within e-Learning can produce the potential of creating realistic environments with which students can interact. The student essentially would interact with the intelligent agents which in turn perceive changes in the simulated environment. The intelligent agents would then communicate perceived changes in the environment back to the student who then makes decisions based upon their own perceptions of the environment [16].

Current learning technologies can help create trained novices more efficiently, but they are really not up to the job of creating true experts. For example, multimedia Computer Based Training (CBT) systems are good at presenting information and then testing factual recall using multiple choice or fill-in-the blank questions. However, traditional CBT systems are incapable of providing intelligent, individualized coaching, performance assessment, and feedback students need to acquire deep expertise [8].

Employing the state-of-the-art AI technology in current e-Learning systems can bring personalized, adaptive, and intelligent services to both students and educators. Most of the AI applications have not yet been expanded to or adopted in widely used e-Learning systems, especially open-source systems such as Moodle and Sakai. Current intelligent LMS systems are still in their early stage, while AI applications need to handle some problems or to be modified before applying them into the LMS systems, and AI technology needs to be brought to open source communities [17].

2.4 Utilizing Service Oriented Architecture in e-Learning Systems

Service Oriented Architecture (SOA) is a design pattern that presents IT infrastructure and information systems architecture as loosely coupled, fine granular services that can address system requirements once they are presented either by adding new services or modifying existing ones. SOA also addresses enterprises information systems' inefficiency by enhancing reusability, thus theoretically, shortening information systems development time and effort required. Besides reusability, interoperability and integration are other main driving forces for adopting SOA in e-Learning systems. W3C defines Service as 'A Component capable of performing a task'. Service is 'A vehicle by which a consumer's need or want is satisfied according to a negotiated contract (implicit or explicit) which includes Service Agreement, Function Offered and so on'. SOA is the design pattern that utilizes services concept to achieve architectural advantages. W3C defines SOA as 'A set of components which can be invoked, and whose interface descriptions can be published and discovered'. This definition can be expanded to include the science, art and practice of building applications, so SOA can be defined as 'The policies, practices, frameworks that enable application functionality to be provided and consumed as sets of services published at a granularity relevant to the service consumer. Services can be invoked, published and discovered, and are abstracted away from the implementation using a single, standards based form of interface' [18].

2.4.1 Web services as main SOA enabler

Web services are applications that use standard transports, encodings, and protocols to exchange information [19]. A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. W3C defines Web service as "A software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a format that machines can process (specifically WSDL), Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with XML serialization in conjunction with other Web-related standards" [20]. Web service can also be defined as 'A programmatic interface to a capability that is in conformance with Wsnn protocols'. Wsnn protocols are present efforts in the W3C and more recently in OASIS to reach a Web service maturity model. Wsnn protocols include WSDL, SOAP, and XML [21]. SOAP is a lightweight protocol intended for exchanging structured information in a decentralized, distributed environment [22]. XML solves a key technology requirement that appears in many places. By offering a standard, flexible and inherently extensible data format, XML significantly reduces the burden of deploying the many technologies needed to ensure the success of Web services [23]. Web services is a general framework that expedites the sharing of heterogeneous data and software resources dispersed on the internet. The standard-based resource sharing and platform-neutral characteristics of Web services have motivated many organizations to apply the technology in diverse areas, such as supply chain management, virtual enterprise, homeland defense, and e-business [24].

2.4.2 SOA and e-Learning Systems

Integration, interoperability, scalability, and reusability are the main axis that SOA based e-Learning researchers attempt to address, solve, and enhance in e-Learning systems. Though some researchers ignored the pedagogical features of e-Learning systems, others considered those features as the main motivator of adopting SOA in e-Learning. e-Learning is moving from passive to active; because the learner is getting more involved with the learning experience [25]. Besides, technology emerging from the adaptive hypermedia, semantic web, mobility and distributed computing communities are being widely employed in online learning, and steps toward Service-Oriented e-Learning platforms need to be taken. LMSs can be classified into three generations: First Generation “Monolithic”, Second Generation “Modular”, Third Generation “Service-Oriented”. First generation focused on presenting learning content without tracking learner activities. Second generation enhanced the learning process but cannot stand handling new technologies, interoperability and integration between evolving systems like third generation shall do. Interoperability needs to be on control basis not just on data basis. Service-Oriented LMS include adaptive hypermedia and semantic web.

2.4.2.1 Moodle and Web services

Research work that focuses on “assignment” modules based on Moodle and the steps to enrich it with Web services is presented in [26]. It is argued that Web services is important and required in integrating different Moodle(s) resources; especially assignments, and to provide the capability for more than one instructor to be working on the same course. Instructor can search for the best assignment within different Moodle(s) instances and retrieve it. PHP based SOAP Web services to integrate best with Moodle is presented. There is no evaluation presented at his research, however it presents a utilization of UML in describing Web services specifications. Though the research work title includes SOA, the common misusage of Web services as SOA is clear in this research. Web services are main SOA enabler, however SOA is not just Web services.

2.4.2.2 Services based e-Learning Systems and Reusability

One of the good utilization of SOA in e-Learning is presented by [27] to address the reusability capabilities provided by SOA. It is argued that instead of building an e-Learning system from scratch, it can be assembled by choosing the required functionalities from a set of Web services related to e-Learning. Study results in a set of Web services for the e-Learning domain. In the abovementioned study, there are different points of strength and weaknesses. The main functionalities of typical e-Learning systems are built using Web services. The strength of this approach is reusability and interoperability. Developing a new e-Learning system will involve assembling the required Web services. The e-Learning system can be developed and run on different hardware and software platforms. However, the disadvantage of this approach is if the server that hosts the Web services is off-line, the e-Learning system which depends on these Web services will not be operational. Another important feature of this study is the use of rubrics in guiding the instructor to evaluate subjective assessment. Rubrics improve consistency in the evaluation of students' work especially when there is a large number of student and multiple instructors are involved in the evaluation. Another good point of the proposed research is the commitment to E-Learning Framework (ELF) and e-Learning standards as depicted in the architecture presented at the study.

However and to repeat from almost all services based LMSs, SOA is not Web services. There is no indication of implementation at all, and so evaluation was not mentioned. It is really rare to find evaluation for SOA based e-Learning systems. SOA adoption in e-Learning systems is new to some extent.

2.4.2.3 Services based e-Learning Systems and Scalability

Another SOA based framework for e-Learning Systems is presented by [28]. Scalability issues of educational institutions are discussed and SOA based e-Learning framework to address scalability capabilities of SOA is presented. This research handles the educational institutions incapability of maintaining large scalable LMSs by grouping LMSs services (based on Advanced Distributed Learning “ADL” Seven services) into local and global groups. Local services group which constitutes the “Local LMS” are provided and maintained internally while other services can be spread over the internet. ADL modularize LMS into seven services: Tracking, Delivery, Learner Profile, Course, Sequencing, Content Management, and Testing/Assessment. It is argued that tracking and delivery services shall compose local LMS while rest of the services can be spread globally. Authors argue that maintaining learning material might exhaust educational institution resources, so Local LMS will not store learning material, however it will acquire it from different places.

Though proposed framework combined content with course services to be almost one service, and though they did not mention Assessment/Test service at all, they still presented a SOA based e-Learning framework that included modifications to the Original ADL LMS model to fit with the research objective.

2.4.2.4 Services based e-Learning Systems and Integration

Web services oriented framework for e-Learning systems aimed at providing a flexible integration model in which all the learning components and applications are well defined, effectively discovered, and loosely connected through using Web services as the main system component to achieve that objective is presented in [29]. However, it argues that there is a need for a standard mechanism for supporting complete automation through all aspects of end-to-end learning process that includes finding suitable learning components or learning management services, getting information about their services and invoking their services. Though automation of complete end-to-end learning process might seem to be a goal to achieve, doubts about the effectiveness of such a learning approach is under questions. Arguments that prove the efficiency and effectiveness of pure e-Learning programs are not completely confirmed. There is a need of human tutor to support blended learning (at least at some level of the learning process). This research did not include any details about evaluation of the proposed work. Experience proved that SOA is not just Web services.

Another SOA based e-Learning research project is the one presented in [30] that presents a SOA based Course Management System (CMS) to address different integration challenges. One of the critical limitations of a newly established educational institution is the lack of available well prepared courses. It is more applicable to use widely available courses that might be higher in quality than preparing new courses. Current CMSs do not exploit courses shareability. To address this shortage, a CMS is presented to highlight automated discovering and importing of courses maintained and managed by external CMSs. Proposed CMS architecture utilizes SOA as a design pattern to integrate different CMSs on service level. Proposed CMS consists of two layers: Presentation layer and Service Layer. Presentation layer is responsible for interacting with user either via displaying information or receiving user inputs. Service layer contains core system services. Service layer is divided into three sub layers: orchestration layer, application services layer, and agents layer. Orchestration layer holds business logic required by system processes. Application services layer contains set of stateless services that are capable of performing certain tasks. Agents layer presents the suggested required software agents to serve the overall system. Suggested agents are: Discoverer, Ranker, Tracker, and Analyzer software agents. Integration between software agents and Web services is achieved by utilizing SOA. Proposed CMS facilitates integration between different CMS in order to share resources of educational institutions.

Other SOA based e-Learning research projects that concern integration is the one presented in [31]. It is argued that Web services have drawn the attention of learning technology researchers and practitioners, e.g., for decentralized, integrated support of Web-service-based agents, for contract-based provision and discovery of distributed Reusable Learning Object (RLO) repositories, or for enhancing the functionality and interoperability of existing learning technology applications, to mention a few. All of these approaches employ Web services to increase extensibility and flexibility of existing solutions and to foster standards-driven development, dissemination, and usage of desired functionality. They present an open, standards-based learning technology architecture that uses distributed Web services to support a broad variety of blended learning scenarios. This project addresses blended learning where; according to the author, not much projects address efficiently.

2.4.2.5 Services based e-Learning Systems and Interoperability

“Scalable Adapter” design pattern constitutes a software architecture that can be used to create interoperability between differently targeted educational tools. The key idea behind the pattern is to add a small “data adapter” to each learning environment. The adapters can then access arbitrary (scalable) parts of the data of “their” learning environment and exchange this data with other adapters. These changes are not costly and usually easy to create since the existing systems do not have to be changed but merely need to be extended. This design pattern is used mainly to provide interoperability features between different e-Learning tools used in educational institutions. Advantages of this design pattern are many and three different case studies presented in this research.

However, integrating different applications on data level has never been the appropriate solution for all system integration problems.

Interoperability approach based on data is presented in [32]. There are driving forces for applying that approach, include: easier exchange of data, reduced development time, and reduced maintenance costs. Architects and programmers do not have to change current already running systems; instead they need to add new adapters that connect systems together. However, connecting systems together via adapters is not “always” the optimum solution. Data based interoperability lacks application logic interoperability and might stand as an obstacle against Business Processes adoption within educational institutions. However, data based interoperability cannot be ignored.

Another SOA based interoperability research project is the one presented by [33]. This research project presents a SOA based Assessment Management System (AMS) to address Mobile Assessment as one of the e-Learning activities. Mobile Learning (M-Learning) is an approach to e-Learning that utilizes mobile devices and is strongly recommended to be enabled by LMS. Assessment is one of the learning activities that can be achieved electronically and via mobile devices. Mobile assessment refers to the capability of conducting assessments via mobile devices. Mobile assessment relies on external services that are not part of the LMS. Providing interoperability between different external systems and services to be virtually part of the educational institution LMS is one of integration and interoperability challenges. Authors presented an extension to the SOA based LMS developed in the faculty of computers and information systems at Mansoura University to address mobile assessment. Proposed architecture consists of two layers: Interface layer, and Service layer. Interface layer interacts with instructors, learners, and business managers via human interface (portals), and with external organization services via machine interface (Web services). Service layer contains core LMS services and has three sub layers: Orchestration, Application Services, and Agents layer. Orchestration layer holds business logic presented by system processes as executable services. Application Services layer contains set of stateless Web services that are capable of performing certain tasks related to system entities. Agents' layer presents the suggested required software agents to serve the overall system. Agents' layer presents Tracker software agent; which is responsible of tracking students' non-conducted assessments and taking appropriate actions to inform them.

Other interoperability approach that acts on "service basis" is presented by [34]. Importance of standards and its role in interoperability between different applications is highlighted. Also, they argue that layering is one of the patterns that are important while considering e-Learning solutions. Also, LMSs' future tendency is to be service-oriented. In this scenario, LMSs are based on modular components and they can support different services that do not stick to a specific platform. They envision that the ideal scenario is one in which all the different educational services can be interoperable among different LMSs; and in which the entire design of different LMS courses can be done off-line (outside of LMSs) in an easy way for teachers without high technological knowledge using proper authoring tools' and next these courses can be imported within the different LMSs.

Infrastructure layer represents the final resources of an institution, such as file systems or databases. The Common Services are services that are used by several educational applications, such as authorization or authentication. The Educational Services are specific educational modules like assessment or Course Management. Finally, the Educational Applications are the applications a user directly interacts with and these educational applications can use the implemented educational and common services. The IMS Abstract Framework architecture is very similar to the defined by Open Knowledge Initiative (OKI), and a perfect relationship among layers of both architectures can be established. Both architectures capture the strong importance of LMS services. Besides interoperability, Reusability of courses, services, and all materials is a main motivator of researchers to highlight the importance of layering and standards to achieve interoperability and reusability between different educational systems.

2.4.2.6 Services based e-Learning Systems and XML

e-Learning framework based on the design and implementation of a middleware is presented in [35]. Authors ought to adopt technologies that are standardized and widely deployed in both e-Learning systems and network infrastructure layers. The general architecture of the e-Learning management scheme is based on the configurable component-based middleware architecture for deploying e-Learning services. Each component has one or more agents, which maintain a local XML-based Management Information Base (MIB), and communicate with manager residing at the service or session management component. The agent communicates with the middleware via a light protocol such as SOAP. Another research project that utilizes XML in e-Learning is the one presented by [36]. E-Learning platform with the required functions to provide information to everyone in anytime and at anywhere is proposed.

2.5 Technical Problems with Services based e-Learning Systems

SOA utilization from different researchers’ perspectives leads to different SOA based e-Learning systems architectures. From full SOA functionalities architectures to just Web services enabled architectures are available by different researchers. SOA based e-Learning systems frameworks are available; without much evaluation and further analysis of points of strength and weaknesses. Architectures that tend to support fully automated learning process and architectures that tend to support blended learning are available. SOA based systems that support point-to-point integration and interoperability, and architectures that support Middleware based integration and interoperability also exist. Each approach has its own points of strength and points of weakness. Point-to-point integration enhances performance. On the other hand, middleware based architectures are more flexible, scalable, and fault tolerant. Evaluation of utilizing SOA in e-Learning systems need to be more studied to highlight advantages and shortages of utilizing SOA in e-Learning, and still SOA based architectures vary a lot from a research project to another.

2.5.1 Evaluation of Utilizing SOA in integrating e-Learning Systems

To help universities achieve their goals, it is important to align managerial functionalities side by side with educational aspects. Universities consume University Management Information Systems (UMIS) to handle managerial aspects as they do with Learning Management Systems (LMS) to achieve learning objectives. UMIS advances LMS by decades and has reached stable and mature consistency level. LMS is the newly acquired solution in Universities; compared to UMIS, and so adopting LMSs in universities can be achieved via three different deployment approaches.

This part presents the current situation at Mansoura University; Egypt, presents integration as the most suitable solution, and evaluates three different implementation techniques: Dynamic Query, Stored Procedure, and Web services. Evaluation concludes that though SOA enhanced many different aspects of both UMIS and LMS; and consequently university overall. It is not recommended to adopt SOA via Web services as the building unit of the system, but as the interdisciplinary interface between systems. Mansoura University runs its in house developed UMIS for more than a decade right now. UMIS has reached a stable and well mature state when compared to the newly introduced LMS in the university. To adopt LMS functionalities in the University; without making LMS and UMIS isolated islands, there are three deployment approaches to choose from:

- Approach One (LMS replaces UMIS): University will replace its UMIS with LMS that will perform both educational and managerial functions. Challenges are: UMIS has been customized to fit University rules and regulations and it is not easy to let it go simply, importing current data into LMS might be a challenge, and there is a risk of system inconvenience especially if LMS failed to provide managerial functionalities as UMIS.
- Approach Two (UMIS takes over LMS): University will add learning functionalities to current UMIS. Though this approach overcomes shortages of previous one but still has some challenges to manage, like time to develop and add the new features while university can make use of advanced features available right now via LMS providers, and dealing with emerging standards.
- Approach Three (Integrate UMIS and LMS): Neither LMS will replace UMIS nor UMIS will take over LMS. Both UMIS and LMS will exist and interoperate to enable university to achieve its managerial and educational tasks in efficient and effective manner. This alternative avoids all challenges presented in alternatives one and two. It avoids replacing UMIS risks, and provides flexibility to change LMS without affecting UMIS, and provides an immediate solution to make use of current available LMS functionalities.

Approach three presents the most suitable solution to the current situation with technical challenges to integrate different information systems side by side while keeping in mind that different information systems require different information presentation for the same entity. Student is an example of the entities that require different information presentations. Student data required by UMIS differ than student learning profile needed by LMS. UMIS student record includes data like ID, Social Security Number (SSN), name, age, gender, address (street, city, country), email, username, password, Date of Birth (DOB), faculty, year, department while LMS student profile include data like a detailed records of what students have already learned at the level of learning objects, a learning preferences profile, and a development portfolio of transferable skills, a history of student interactions with tutors. Figure 2.1 presents the current scenario in Mansoura University; where users can be classified into UMIS users; to handle non-educational activities, and LMS users; to handle educational activities. To generate a detailed report of the courses and the learning contents, the user has to go through UMIS to generate the courses IDs and LMS to acquire the learning content. UMIS and LMS are isolated islands connected only via users. The assessment experience by Faculty of Computers and Information Sciences in Mansoura University; http://www.m-assessment.info highlighted many of the challenges exist in the University. Assessment Management System (AMS) team asked students to register explicitly for the AMS; and that is not accepted. A single-student registration is a must to satisfy all learning interactions with the University. Figure 2.2 presents the proposed solution where a Service Layer shall be added in the middle area between UMIS, LMS, and users. Middle layer facilitates integration between different systems via Web services. Web services are relatively a new technology that have received wide acceptance as an important implementation of SOA. Middle layer can provide portal(s) to unify users' interaction with different systems.

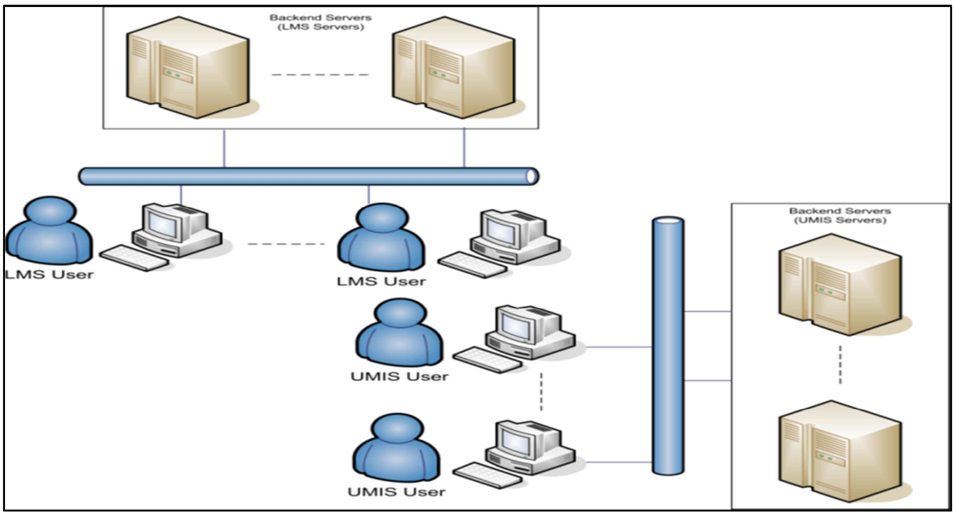


Figure 2.1: Current Scenario (Isolated UMIS and LMS integrated via users)

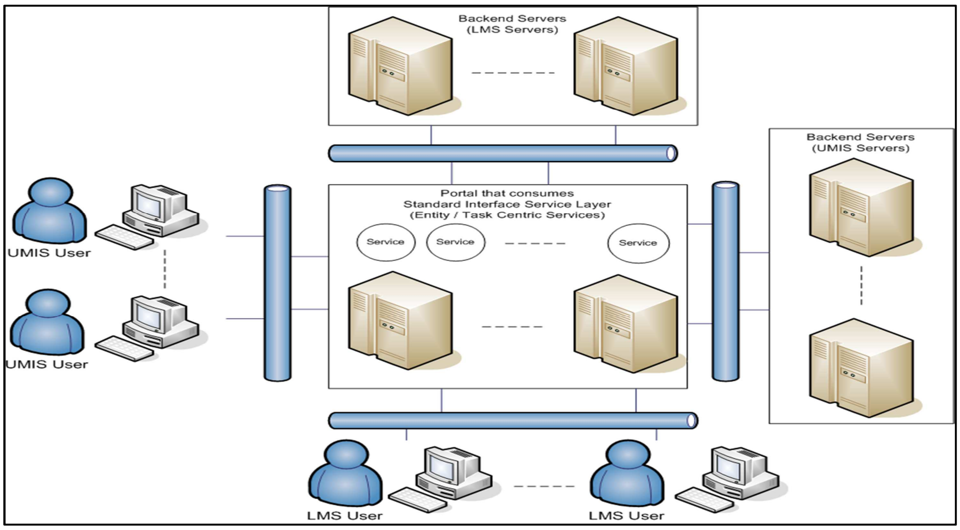


Figure 2.2: Proposed Solution Presenting a Service Layer as Intermediary between UMIS, LMS, and Users

It is evaluator's responsibility to determine the most valuable architectural aspects to be considered in the evaluation process. Information systems quality parameters evaluated in this chapter are: Network Performance, User Perceived Performance, Integration and Interoperability, and Reusability. SOA enhances system overall security, replace ability, testability, and both hardware and software scalability.

2.5.2 Technical Evaluation Quality Parameters

Quality parameters like performance, integration and interoperability, compliance, security, maintainability, analyzability, decomposability and modularity, testability, portability via replaceability and scalability, simplicity, modifiability, and reusability shall be addressed. A Comparative performance analysis study is presented to test SOA based systems user-perceived performance against non-SOA based systems.

2.5.2.1 Network Performance

SOA based systems relies heavily on messaging. It is clear that SOA based applications need to add extra headers to manage requests and responses in standard format. Header can be classified into two Static Header and Dynamic header. Static header is added once for every time the service is invoked while Dynamic header is added for every record contained within request or response message. By analyzing data in the request and response messages, it is noticeable that there are three data categories:

- Static Header: This header occurs once for each service invocation no matter how many records in the request. There are 463 characters for one of the test headers.
- Dynamic Header (XML Tags): Those tags are the overload of requests and responses. Those tags are named by developer, so they are not static every time, but in the same test message there is 179 characters.
- Actual Data: Those are the record details to be inserted after invoking the Web service.

Added extra headers differs according to the no. of records to be handled, and differs from an application to another (because the header used to represent an author might be different from the one used to represent a book) so network performance differs from an application to another. It is the system architect responsibility to decrease the transferred data over the network to the maximum extent (so decrease network delay) because it is noticeable that headers needed by SOA cannot be neglected easily.

2.5.2.2 User Perceived Performance

Web services are the main SOA enabler. It is expected that utilizing Web services within an application will affect User Perceived Performance. In order to understand the extent to which Web services affect User Perceived Performance, three different Library Management Systems (LIS) were implemented tested against the same data samples.

The three different LISs are Parameterized Query based LIS, Stored Procedures based LIS, and Services based LIS. While Parameterized Query based LIS SQL statements exist within the web pages and accesses database directly, Stored Procedures LIS highlights the separation between data layer and application layer by the presence of Stored Procedures as a middle layer in-between the portal and the databases. The Portal consumes stored procedures to access the databases. The services based LIS presents the services layer in between the portal and database layer to present a standard based interface layer that consumes stored procedures and available for portals. Services based LIS presented the highest arithmetic mean and mode values while Stored Procedure based LIS were the best for the insert operation. From the presented performance analysis and after evaluation of the three LISs, it is clear that the time consumed to perform the same operations using the services based LIS exceeds the time consumed to perform the same operation using either the stored procedure LIS, or the parameterized query one.

2.5.2.3 Integration and Interoperability

Systems can share their effects within a single operation via service level integration. Assessment Management System (AMS) did not have to access Student Affairs Information System database tables to retrieve and update student table data; instead, it just invoked the Update_Student service exposed. AMS includes a Take Assessment Process that needs interoperability between AMS and external systems as presented in [33]. Without this interoperability, Mobile assessment would not have taken place at all. SOA utilization in the system gave the system capability to expose standard interfaces that act like sockets to be plugged in to connect systems.

2.5.2.4 Reusability

Reusability is achieved in the proposed architecture on two levels: Internal and External. Internal reusability distinguished the application capability to use the implemented service more than once without modification. This happened with the Update functionality, where it consumed Delete and Insert functions. Services were not written every time. External Reusability refers to the external systems that consumed the exposed internal services to achieve functionalities. CMS shared services with AMS and UMIS, and that reusability distinguished the advantages of SOA.

2.6 Summary

Mansoura University runs its in house developed University Management Information System (UMIS) for more than a decade. UMIS has reached mature state compared to the newly introduced LMS. To adopt LMS functionalities in the University without making LMS and UMIS isolated, both need to interoperate to enable university achieve managerial and educational tasks. Comparative performance study was conducted to test SOA based systems performance against non-SOA based systems. Technical Challenges include:

- SOA based systems rely heavily on messaging and add extra headers to manage requests and responses in standard format. Headers sizes differ according to the number of records to be handled, and differ from application to another, so network performance differs. However, the comparative study yielded the fact that network delay is highly affected by the header size.
- Three different Library Management Systems were implemented and tested against the same data samples. The three different systems are Parameterized Query based, Stored Procedures based, and Services based. Services based System faced serious challenges in performance when compared to other two systems.

Chapter 3

Chapter Three

Proposed Adaptive e-Learning Models

3.1 Introduction

This chapter presents: Adaptive e-Learning Model as a solution to pedagogical e-Learning challenges facing students, and Adaptive Online Lecture as an enabler to instructors to address adaptivity features in e-Learning in new and innovative form.

3.2 Adaptive e-Learning Model as a Solution

There is a need to present a learning model that utilizes SOA in providing integration and interoperability between different e-Learning system components to provide students with adaptive features. This chapter presents an Adaptive e-Learning Model that E-Learning system components include: Adaptive LMS, Quality Assurance and Assurance Project Management System, Exam Management System, and Learning Content Management System (LCMS) as depicted in figure 3.1.

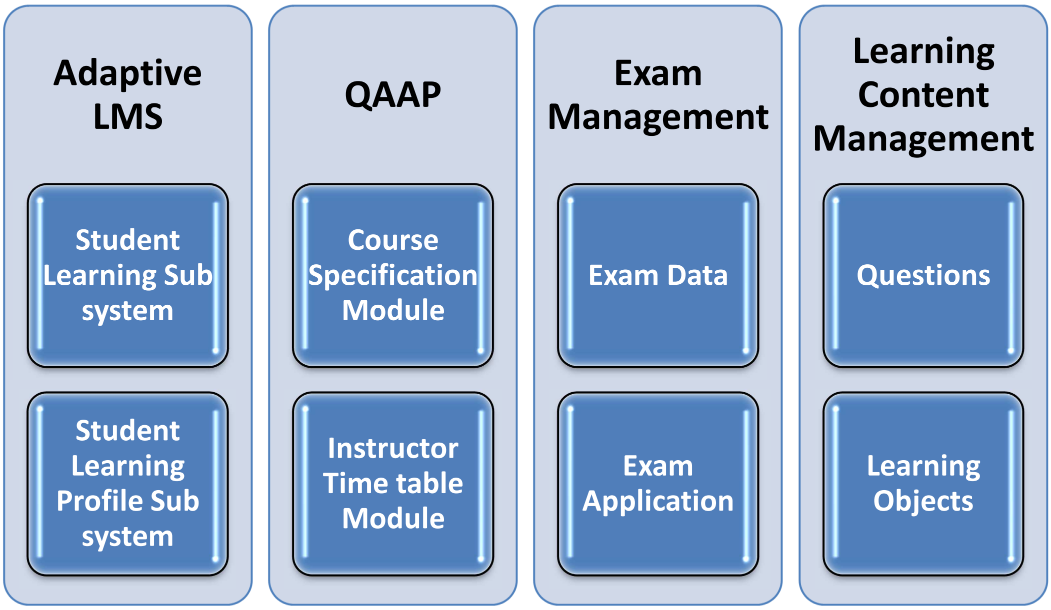


Figure 3.1: Adaptive e-Learning Model Components

3.2.1 Adaptive e-Learning Model Components

Proposed adaptive e-Learning model requires the integration of different systems to achieve the required goal. There are four sub systems:

- **Adaptive LMS:** Responsible for providing the adaptation features to each student via determining the learning road, topics and time required for each student based on performance, learning profile, and learning preferences. Adaptive LMS provides the basic functionalities provided by different LMSs in an adaptive manner. Adaptive LMS contains two subsystems: Student Learning, and Student Learning Profile subsystem.
- **Quality Assurance and Accreditation Project (QAAP) Management System:** An Egyptian National Initiative and Project that is maintained by the Egyptian Ministry of Higher Education QAAP include: *Course Specification Module*, and *Instructor Module*. *Course Specification Module* focuses on defining and determining courses contents, learning objectives and other course resources. *Instructor Module* contains the instructor time table that will be used to define suitable times for meetings between students and instructors.
- **Exam Management System:** Blended model of online questions repository and desktop application delivery exam is used to overcome Web based exam systems vulnerabilities to cheating. Students will run the desktop application on exams times. The application will retrieve questions from online repositories. Those repositories are maintained by Learning Content Management System (LCMS).
- **Learning Content Management System (LCMS):** It is critical and vital to the success of the proposed model implementation. LCMS holds questions items, and Learning Objects (LOs). Proposed Adaptive e-Learning Model addresses extra needed meta-data for questions and LOs to support needed adaptivity features. LCMS focus on providing a standalone LOs management

that can be utilized by different LMSs. Though LCMS is thought to be part of LMS, it is a best practice to provide it as a standalone system for two reasons: support different LMSs, and isolate LOs meta-data management from LMS.

3.2.2 Adaptive e-Learning Model Learning Scenarios

To make the proposed model clearer, it will be illustrated in words describing what takes place with students in four different scenarios, and present each scenario in a separate figure, followed by a complete figure describing the while adaptive e-Learning model presented in this chapter. Figures present the proposed adaptive e-Learning model using Business Process Modelling Notation (BPMN). BPMN 2.0 is the modelling language provided by Object Management Group (OMG). It is specifically designed to illustrate complex processes and models. Scenario 1 presents Student (A) who uses the system for the first time and has not built the learning profile yet. Scenario 2 continues with Student (A) in the learning phase. Scenario 3 presents Student (B) who is currently doing well through learning, and now has a due exam. Scenario 4 presents Student (C) failed twice before in the exam, and is attending the exam for the third time.

3.2.2.1 Scenario 1: New Student

Student (A) attempts to log in but as it is the first time, student finds himself forced to register. During registration, the student completes the forms needed to identify student’s learning profile and preferences. The second time the student logs in; the Student Learning Sub system tends to retrieve the student learning preferences from student learning profile subsystem. If it is not complete; the system forces the student to complete it before starting to learn. Otherwise, the Student Learning System extracts the student information and registered courses, then checks if this student has an exam. Student (A) does not have an exam, so again, the Student Learning System checks if student has a meeting with an instructor. Student (A) does not have a meeting; otherwise the system would have displayed the calendar. Figure 3.2 presents the details.

3.2.2.2 Scenario 2: Studying Student

Student Learning System calls QAAP Course Specification Module, and acquires the course specifications and list of topics. The Student Learning System displays to the student the list of topics; available and unavailable as a result of the requirements pre-requisites, so the student can identify their position on the roadmap. Student selects the topics to learn within the rules. Student Learning Profile is then updated with the selected and not selected topics. Student Learning System displays the suitable learning material and a list of recommended learning materials and what others have seen learning material list. The Student Learning System qualifies the student to make sure he understands the learning objectives of the topic. If the student is not qualified, then he goes back to the study plan. The Student can quit learning at any time to continue later. Then the Student Learning System checks if the topics that the student has learned form an exam, if yes, student becomes eligible for an exam. Student can go through Learn via Questions (LVQ) experiment. LVQ is a learning method that simulates the exam environment by presenting questions with feedback, so students can measure their readiness for an exam. The main objective of LVQ is helping students define their readiness level, not testing them. Then student receives an exam date, and the learning process moves to Student (B). Figure 3.3 presents this scenario details.

3.2.2.3 Scenario 3: Due Exam Student

Here comes Student (B), who logs into the system and has a due exam, so the Student Learning System attempts to initialize the desktop application responsible for examination process. The exam application retrieves questions from LCMS questions repository based on the exam ID submitted by the

Student Learning System, The exam application ranks the exam, and updates the student profile with this rank. The next time the student logs in, student continues learning new topics. Figure 3.4 presents the process details of this learning scenario.

3.2.2.4 Scenario 4: Suspended Student

Student (C) faces troubles with some topics. Student attended the exam twice but did not pass. So, the third time student logs in, attends the exam and does not pass, an automatic initiation of the Intelligent Meeting Manager for Suspended Students happens to arrange a meeting to this student with one of the instructors to help the student. The meeting details with the student's detailed profile are mailed to the instructors. The next time the student logs in, student finds the system paused and the calendar is displayed directing the student to instructor's meeting. Figure 3.5 presents the details of this scenario.

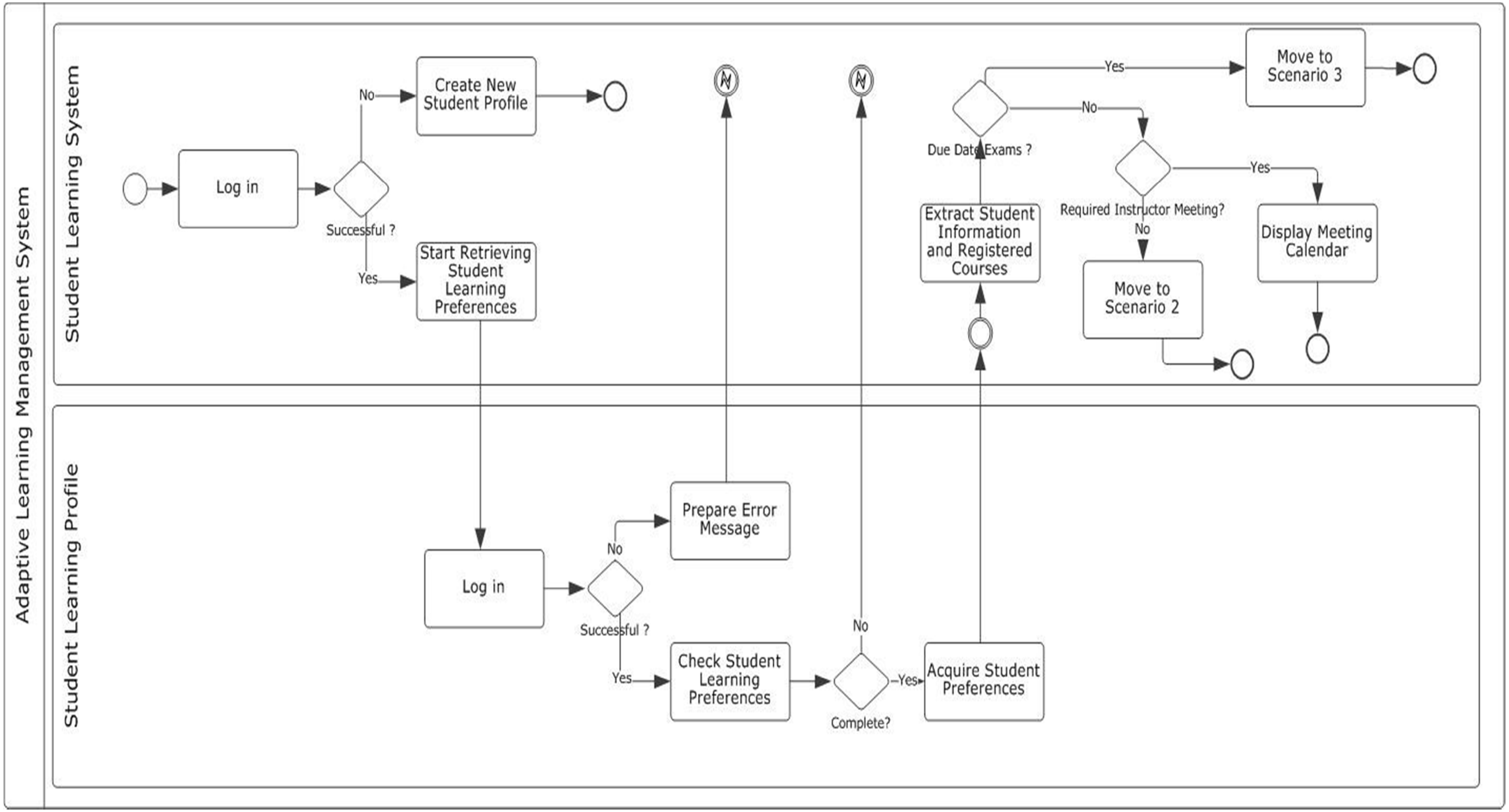


Figure 3.2: Scenario 1 of Adaptive e-Learning Model "New Student"

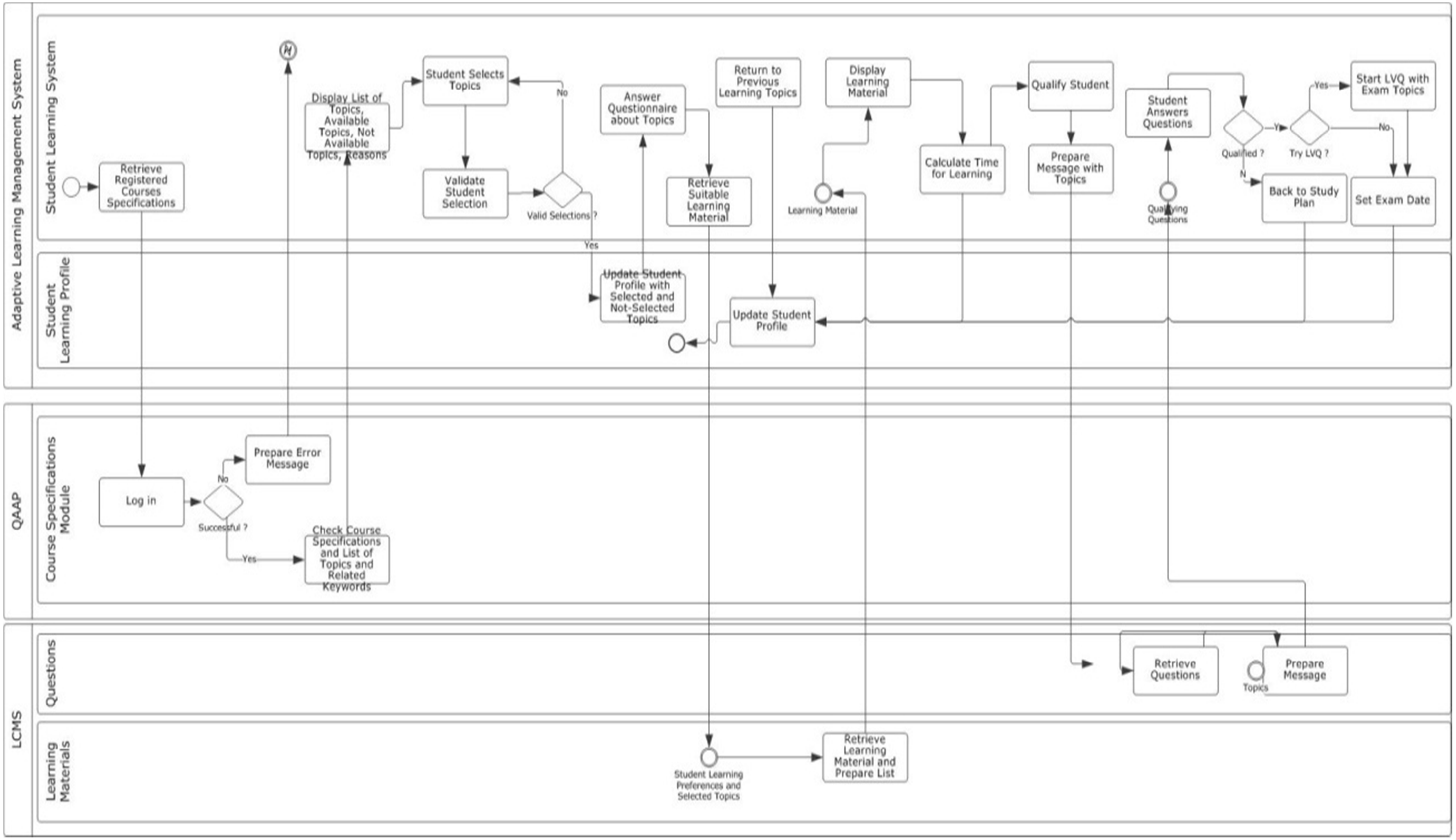


Figure 3.3: Scenario 2 of Adaptive e-Learning Model “Studying Student”

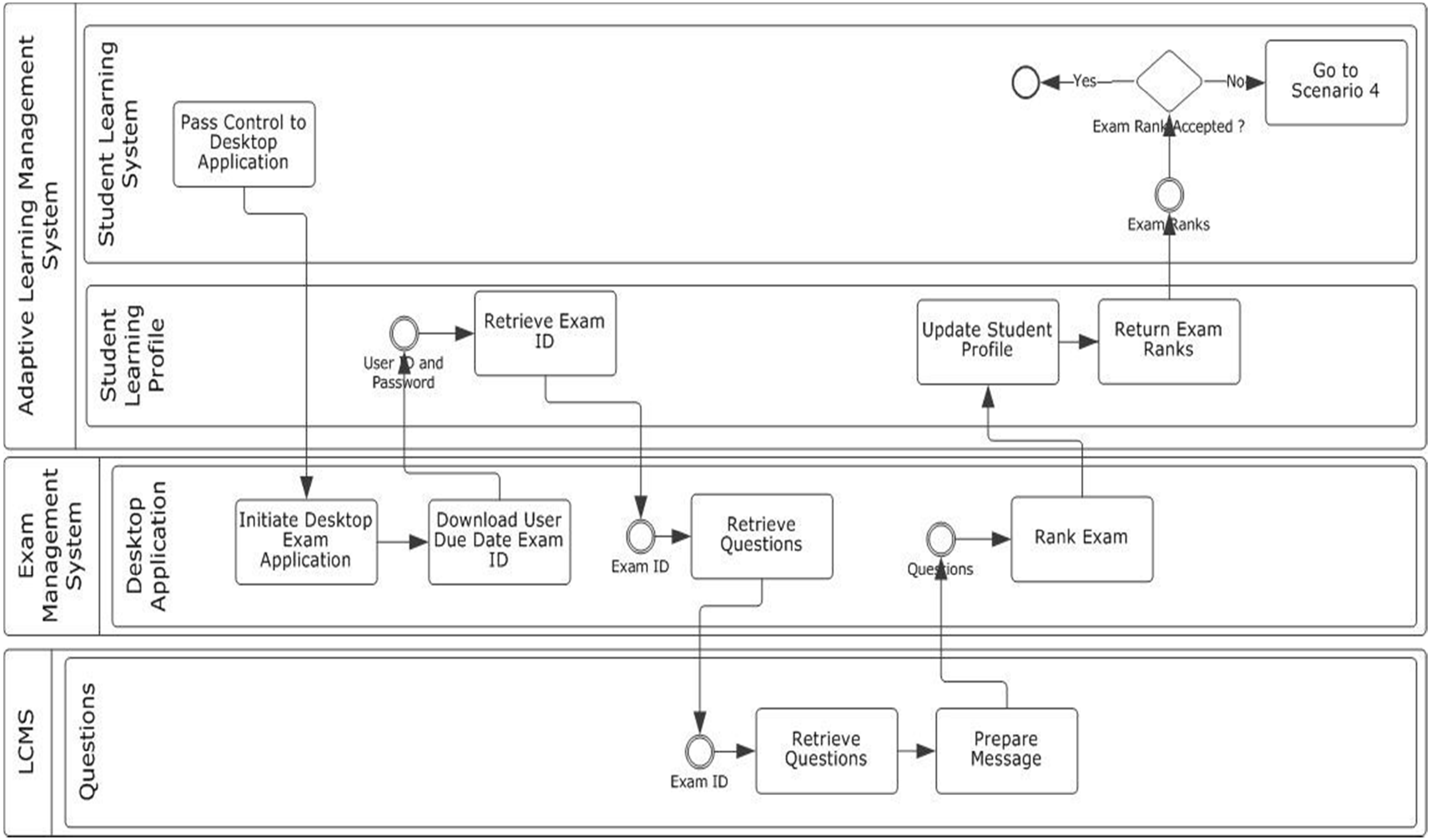


Figure 3.4: Scenario 3 of Adaptive e-Learning Model "Due Exam Student"

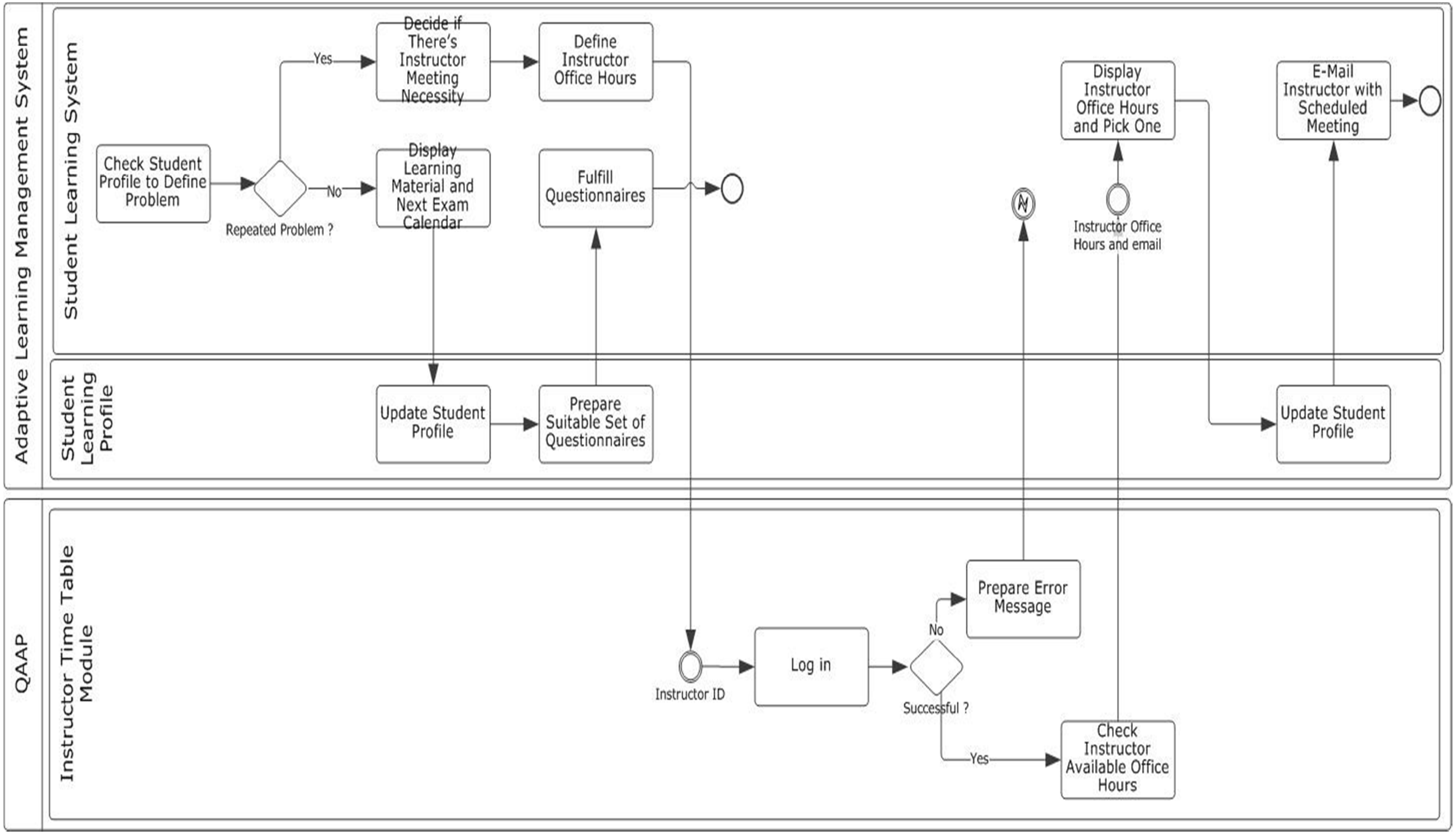


Figure 3.5: Scenario 4 of Adaptive e-Learning Model "Suspended Student"

Only the instructor can solve the situation after meeting the student by reactivating the student account after the proper action has been taken and recorded in the study profile. Instructor can illustrate the topic more than once to the student, examines student orally, written or whatever method the instructor finds appropriate. Figure 3.5 presents this scenario's activities.

3.2.3 Adaptive Features in Adaptive e-Learning Model

Figure 3.6 presents the complete adaptive e-Learning model. Adaptive features in proposed adaptive e-Learning model are many and include:

- Building learning profile and identifying learning preferences for each student using different methodologies.
- Checking student profile and learning preferences before recommending learning objects.
- Allowing the students to choose among the topics to learn within the constraints of the pre-requisites (partial control).
- Capability to arrange meetings for suspended students. Students are given the chance to self-study the subjects and attend the exams 3 times. If the student fails to pass the exam 3 times, a meeting must be arranged between the instructor and the student to submit a report by the instructor to the student profile, so the student can continue the learning process again in the adaptive way. This sort of blended learning gives strength to model.
- Providing the capability to calculate the required time to study a topic.
- Tracking students' behaviour in the exams and attempting to identify cheating incidences.
- Integration with different online forum, wiki and blog services is available to enhance collaboration between students and encourage them to help each other. Facilities to enable online study groups - like chatting applications - are available.

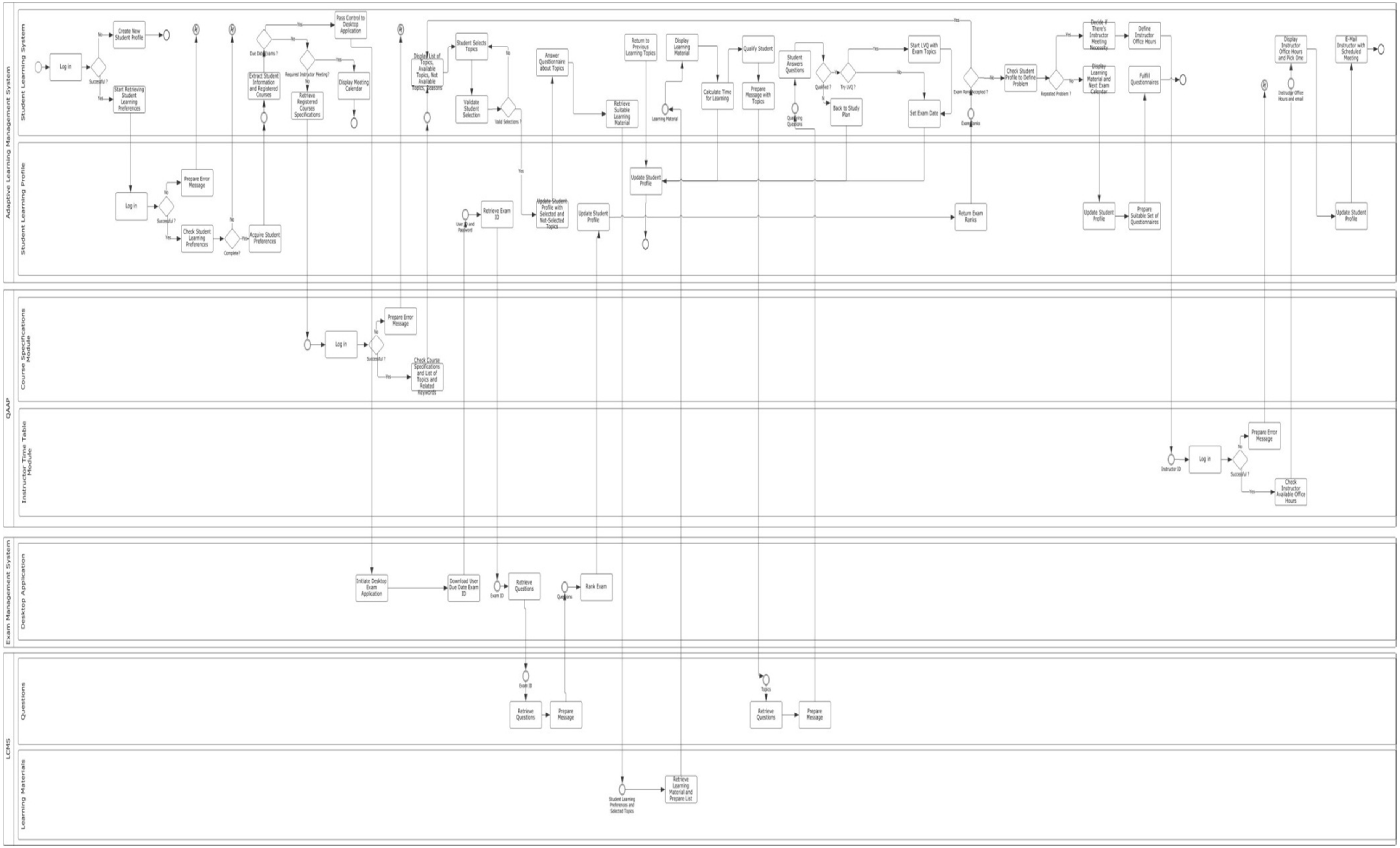


Figure 3.6: Complete Adaptive e-Learning Model

Table 3.1 presents a summary of the mapping between proposed adaptive features in the proposed learning model and the four adaptive learning approaches presented in chapter two – part one. In addition, the proposed model combines both Adaptable and Adaptive capabilities. Adaptability made available by giving students the capability to define and edit their personal preferences, which will affect directly their learning experience.

Table 3.1: Summary of Adaptive e-Learning Model Features and Adaptive Learning Approaches

Adaptive Feature of the Proposed Model	Adaptive Learning Approach
Adaptation of Instructions on a Macro Level	Macro Adaptive Approach
Allowing Students to select among the topics to learn taking in consideration pre-requisites	Partial Control, Aptitude Treatment Interaction (ATI) Approach
Suspend Account and Arrange Meetings with Instructors based on Learning Situations	Aptitude Treatment Interaction (ATI) Approach
Online Forum, Wiki, Blog, Chatting, Grouping Services	Constructivistic – Collaborative Approach

3.3 Adaptive Online Lecture Model

The need for conducting and attending lectures in learning is clear for both students and instructors. Online meetings present the required audio and video communications with the capability to share presentations, desktop activities and transfer files. Different Online Meetings software and applications in both Web and desktop forms are available. However, their design and implementation were not aimed to be used for online lectures in the first place. Despite the tremendous advancement in technology that is witnessed by those applications, they still lack certain level of feedback from students to instructors that pushes students into more engagement within the learning process.

Engaging students in the lecture activities will enhance the students' learning experience. Technically, this is available via extensive utilization of technologies that exist nowadays. During the online lecture, students are encouraged to give continuous informal feedback about different lecture activities via the same Web 2.0 technologies they are using. This feedback can be studied and analyzed later by the instructor. It can be used as an indicator on how the lecture was progressing, and know how to enhance the upcoming lectures. Formal feedback request can be initiated by the instructor periodically to test certain points about which the instructor needs to ensure as a "check point" before moving on to the next point. Involving students in different assignments and activities during the lecture is welcomed and needs to be recorded in the students learning profiles. Finally, preparing for the next lecture is not only the instructor's responsibility. Pedagogically, the instructor is supposed to define the topics for the next lecture and the pre-requisites to learn these topics. Technically, LMS is supposed to check the student's learning profile and preferences to define to what extent the student is familiar with those topics, and then providing the student with the learning materials. It is the responsibility of the student to study and examine these learning materials before the next lecture. Adaptive online lecture goals can be achieved in three phases.

3.3.1 Phase One: Preparing Online Lecture

The students' learning models are not the same, and that will be considered while selecting the contents to be displayed during the Online Lecture. In order not to lose the student's attention during the lecture, types of contents shall be mapped with both their direct feedback and learning profiles. Figure 3.7 presents the different activities required to fulfil this phase.

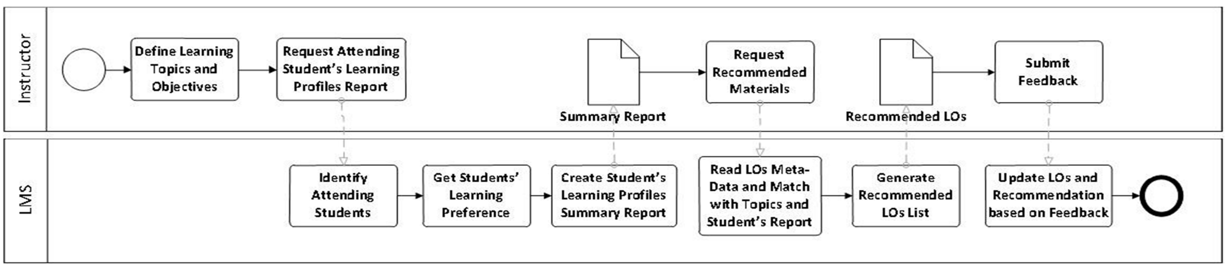


Figure 3.7: Recommend LOs for Instructor based on Analysis of Learning Profiles Process

3.3.2 Phase Two: During Online Lecture

The Proposed Adaptive Online Lecture Model attempts to address: Order of Contents, Assessments, and Assignments and Collaborative as aspects of lecture activities.

- **Order of Contents:** Displaying the video file before / after discussing it or even twice in the lecture is one of the decisions that the instructor might not pay enough attention to, while it is important in keeping the students focused on the lecture activities. If the students are given some capability to re-order the contents of the lecture and discussions, they would feel the personalization of the Online Lecture, and would get deeply involved in the lecture.
- **Assessments:** Instructors might need to conduct one of the on the fly assessments to ensure that students have reached a basic level of knowledge regarding one of the topics they were discussing before moving on to the next topic.
- **Assignments and Collaboration:** Students attending the online lectures are already connected to the Internet via their laptops, they have accounts on multiple Web 2.0 collaboration tools providers, like Microsoft and Google, so they can easily transform to those tools based on the instructor’s directions. Their collaborative work can be marked and discussed online as if they are in a traditional lecture.

Figure 3.8 presents the proposed Activity Diagram of the During Lecture activities that allows students to submit informal feedback during lecture.

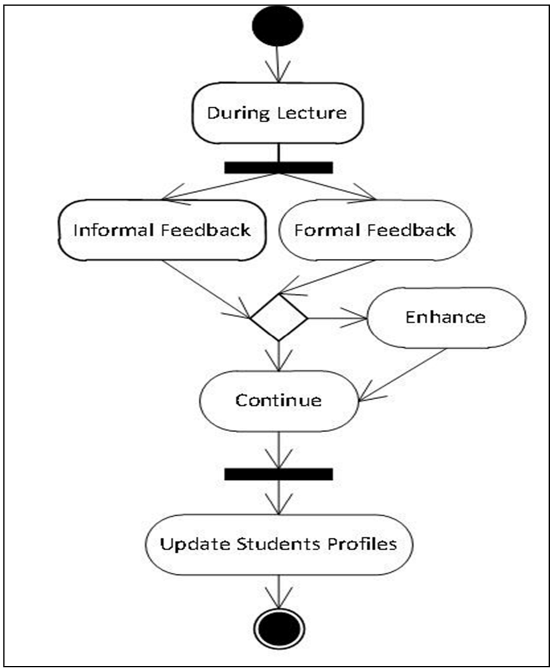


Figure 3.8: Formal and Informal Feedback Processes Synchronization during Lecture

3.3.3 Phase Three: Upcoming Lecture

Before the students leave the current lecture, the instructor shall ensure that they are familiar with the prerequisites of the upcoming lecture. Proposed Adaptive Online Lecture can facilitate this by conducting assessments for the students and ask them clearly about the prerequisites, or by checking their learning profiles. Proposed Adaptive Online Lecture Model can access the Student Profile and Online Preferences for data about their previous attended sessions, courses, specifications and other details. In case one of the students does not satisfy the requirements defined by instructor, a personalized content can be generated for that student via Intelligent LOs Recommender, and then the student's interaction with those materials is tracked. Figure 3.9 presents the different system activities to support the adaptive LOs recommendation.

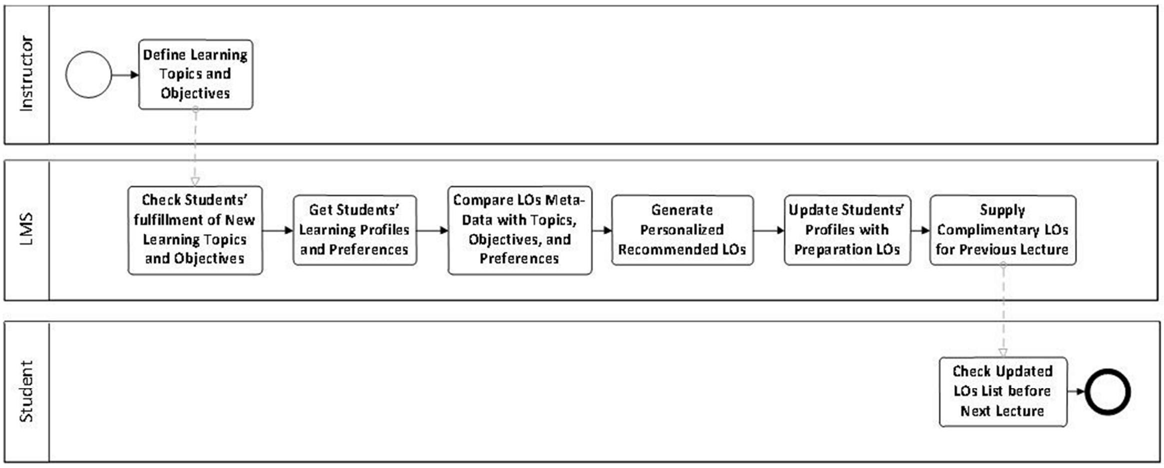


Figure 3.9: Recommend LOs for Students based on Upcoming Lecture Topics and their Preferences

3.4 IT Architecture to Enable Presented Adaptive Models

IT architecture needed to support both the adaptive e-Learning model and the adaptive online lecture model is complex and includes the utilization of different technologies. SOA utilization has many advantages in solving integration challenges. IT architecture includes the following servers as presented in figure 3.10. Servers are grouped into four groups based on their functionality:

1. Utility, Middleware, and Load Balancing
2. Adaptive LMS
3. Learning Content Management System and Content Servers
4. Adaptive Online Lecture

3.4.1 Utility, Middleware, and Load Balancing Servers

This category includes servers responsible for security, authentication and authorization, middleware, and load balancing. Servers include:

- **Proxy and Firewall Server** acts as the gate between online and external systems, and the rest of Servers.
- **Identity and Log in Server** shall be added to enable single sign-on authentication and authorization capabilities for the whole solution. It helps avoid the repeated Log-in process between different applications and servers.

- **Middleware:** Responsible for managing Quality of Service (QoS) and directing messages among different components of the systems.
- **Web services Application Server**: shall be added to hold Web services responsible mainly for:
  - **Data Adapter**: Accessing data stored at QAAP, and external systems.
  - **Integration and Interoperability:** Between different servers based on different required functionalities.

3.4.2 Learning Content Management System and Content Servers

This category addresses the Learning Objects characteristics, meta-data, recommendation process, and the Learning Content Administration System (LCAS). Servers in this category include:

- **LCMS and Content Servers:** Logically, the following servers are required to manage the presented functionalities:
- **Learning Content Server:** To hold the learning materials physical files. Different implementation can include FTP Server.
- **Learning Content Data and Meta-Data Repository:** To maintain data and meta-data about learning materials. The data will be used to determine the appropriate learning materials for the students.
- **Learning Content Administration System (LCAS):** Holds the application that enables administrators to manage learning content files, learning content data and learning content meta-data.
- **Exam Management Server:** Manages exam process and accesses LCMS questions based on exam IDs.
- **Learning Content Recommender:** Works on the learning contents data, meta-data, students learning profiles, students learning history, and instructors' recommendations in order to recommend learning materials to students.

3.4.3 Adaptive LMS Servers

Adaptive LMS is responsible for providing adaptive functionalities presented in the adaptive e-Learning model and providing the main interface for students to access the Web application. This category include:

- **Adaptive LMS Application Server:** Holds the portal that can be accessed by students.
- **Adaptive LMS Database Server:** Holds the student profiles, learning preferences and learning history.

3.4.4 Adaptive Online Lecture Servers

This category includes the required servers to support the Adaptive Online Lecture model, which are:

- **Collaboration, Assessments and Assignments:** Main components of the Learning Process that are maintained separately to provide greater flexibility and the ability to utilize different technologies.
- **Real-Time Communication Server:** Responsible for providing communications functionalities between instructors and students, and the students and each other. It manages Online Lecture file, desktop, text sharing, other activities and Web 2.0 technologies that will be used in the Informal Feedback.
- **Analyzer and Report Generator:** Responsible for analyzing gathered data and generating appropriate reports to help instructors take the appropriate decisions.

3.5 Integration of Proposed Models Services Via SOA

A combination of both Business Process Management and SOA is proven to achieve numerous advantageous features for systems. Proposed adaptive e-Learning model presents an adaptive process that changes based on students' performance. SOA is the utilized software architecture in building the system and in integrating different components required to support the adaptive models. Services are the building blocks of SOA, and proposed model services can be categorized in the layers depicted in figure 3.11. The four layers are:

- **Orchestration Layer:** holds services responsible for maintaining learning process logic and activities. It includes services that utilize both composite services' layer services, and data services' layer services.
- **Composite Services:** are services that hold other services and don't complete functioning unless all composing services execute successfully; however it is not controlling them.
- **Data Services Layer (Information as a Service "IaaS"):** is the layer that holds services responsible for transforming Meta-data into meaningful information to other utilizing information systems, instructors, and students.
- **Model Layer (Database):** it is the database layer that holds data tables.

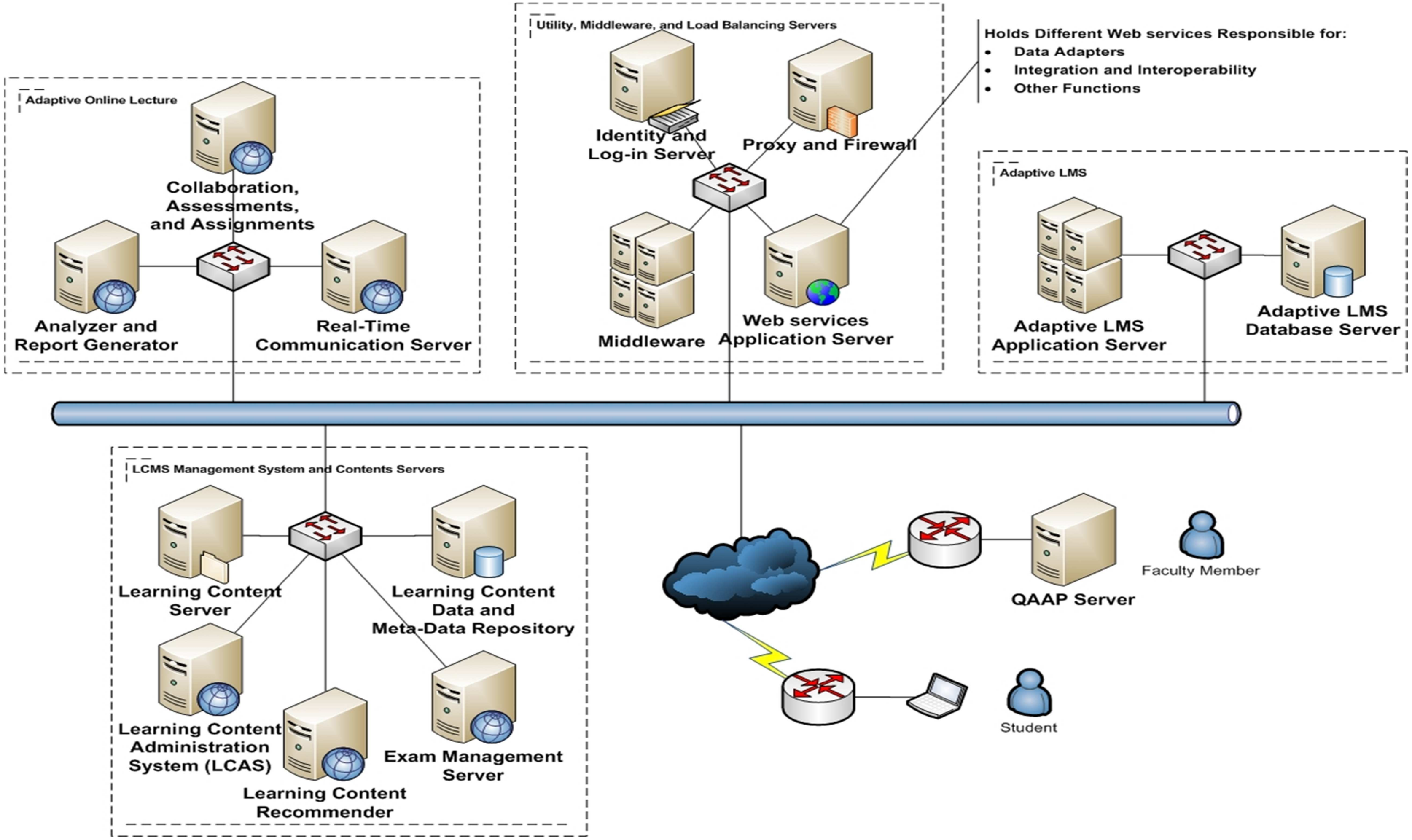


Figure 3.10: IT Architecture Specifications to Enable Presented Adaptive Models

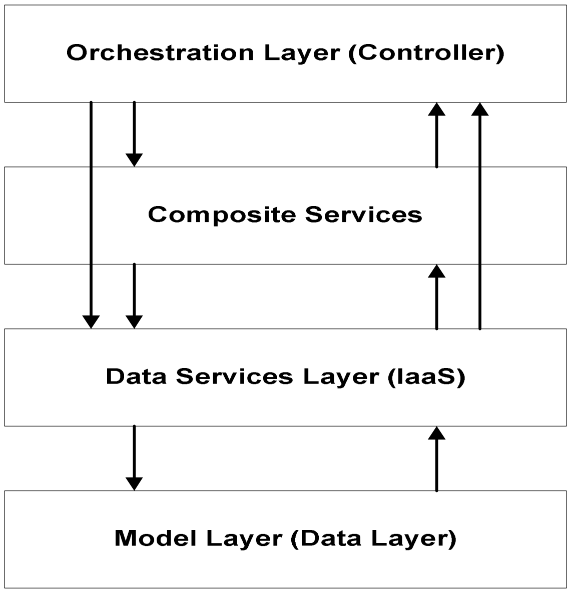


Figure 3.11: Proposed Model Layered Architecture

Table 3.2 presents a summary of presented services, applied to IT architecture to highlight the implementation of the services, and identify how services utilize each other. Table 3 layers are:

- **Orchestration Layer:** holds services responsible for maintaining learning process logic and activities. It includes services that utilize both composite services' layer services, and data services' layer services.
- **Composite Services:** are services that holds other services and don't complete functioning unless all composing services execute successfully; however it is not controlling them
- **Data Services Layer (Information as a Service "IaaS"):** holds services responsible for transforming Meta-data into meaningful information to other information systems, instructors, and students. Data adapters for accessing external systems are presented in this layer. QAAP data adapter for example is presented for accessing QAAP.
- **Model Layer (Database):** it is the database layer that holds data tables.

Table 3.2: Model Services, Categorized by Architecture Layers, and Mapped to IT Infrastructure

Design Layer	Service Name	Parent System	NW Diagram Server	Utilized Services
Orchestration Layer (Controller)	**O1:** Search for LO	Adaptive LMS	Adaptive LMS	C1,C2
	O2: Intelligent LOs Recommender	Adaptive LMS	Adaptive LMS	D2, D3, D4, D5
	O3: LVQ	LVQ	Adaptive LMS	
	O4: Intelligent Meeting Manager for Suspended Students	Adaptive LMS	Adaptive LMS	D5,D10
	O5: Intelligent Study Plan Advisor	Adaptive LMS	Adaptive LMS	D2, D4, D5
	O6: Intelligent Time-to-Learn Topic	Adaptive LMS	Adaptive LMS	D3, D4, D5
	O7: Intelligent Online Lecture LO Classifier	Adaptive LMS	Adaptive LMS	D1, D4, D5, D10
	O8: Intelligent Student Tracker	Adaptive LMS	Adaptive LMS	D5
	O9: Intelligent Cheat Depressor	Adaptive LMS	Adaptive LMS	D5, D6, D7
	O10: Intelligent Agenda Time Planner	Adaptive LMS	Adaptive LMS	D1, D2, D3, D4, D5
	O11: Recommend LOs for Lecture	Adaptive Online Lecture	Learning Content Recommender	D1, D5
	O12: Prepare Next Lecture	Adaptive Online Lecture	Learning Content Recommender	D1, D5
	O13: Formal Feedback	Adaptive Online Lecture	Collaboration, and Assessment Server	
	O14: Informal Feedback	Adaptive Online Lecture	Real-Time Communication	C2
	O15: Generate Lecture Report	Adaptive Online Lecture	Analyzer and Report Generator	
	O16: Satisfy Student Missing Pre-requisites	Adaptive Online Lecture	Learning Content Recommender	D1, D5

Composite Services	**C1:** Intelligent LOs Classifier	LCMS	Learning Content Recommender	D1,D5
	C2: Gather Informal Feedback	Adaptive Online Lecture	Analyzer and Report Generator	D11
Data Services Layer (IaaS)	**D1:** Learning Objects Manager	LCMS	LCAS	DB1
	D2: Lessons Manager	Adaptive LMS	Adaptive LMS	DB2
	D3: Topics Manager	Adaptive LMS	Adaptive LMS	DB2
	D4: Course Manager	Adaptive LMS	Adaptive LMS	DB2
	D5: Student Profile Manager	Adaptive LMS	Adaptive LMS DB Server	DB3
	D6: Questions Manager	LCMS	Exams Management	DB4
	D7: Exams Manager	LCMS	Exams Management	DB4
	D8: Questionnaires Manager	Adaptive LMS	Adaptive LMS	DB3
	D9: Course Module Adapter	QAAP	Web Services Application Server	--
	D10: Instructor Data Adapter	QAAP		--
	D11: Social Networks Data Adapter	Social Network Sites		--
	D12: Assignments Manager	Adaptive LMS	Adaptive LMS	DB5
	D13: Assessments Manager	Adaptive LMS	Adaptive LMS	DB5
Model Layer (Databases)	**DB1:** Learning Objects	LCMS	Learning Content Repository	--
	DB2: Lessons, Topics and Courses	Adaptive LMS	Adaptive LMS DB Server	--
	DB3: Student Profiles and Questionnaires	Adaptive LMS	Adaptive LMS DB Server	--
	DB4: Questions and	LCMS	Exams	--

	Exams		Management	
	DB5: Assignments and Assessments	Adaptive Online Lecture	Collaboration, Assessments, and Assignments	--

3.6 Learning Objects in Presented Adaptive Models

Learning Object (LO) is the basic building block of a learning resource; it is the electronic representation of media, such as text, images, sounds, assessment objects or any other piece of data that can be rendered by a Web client and presented to a learner. LOs play an important role in the proposed system to present different adaptivity features, hence the need to specify a standalone system to manage it, which is Learning Content Management System (LCMS). LO's meta-data needed to support the adaptivity features are presented in details in table 3.3 highlighting the focus area of each category and presenting the needed attributes to be stored for each category. LOs meta-data grouped into:

- **General:** Groups general information that describe LOs as a whole.
- **Lifecycle:** Groups the features related to the history and current state of LO and those who have affected the component during its evolution.
- **Technical:** Describes technical characteristics and requirements of LO.
- **Educational:** Describes educational or pedagogic characteristics of LO.
- **Rights:** Describes intellectual property rights for LO.
- **Annotation:** Provides comments on the educational use of LO.
- **Classification:** Describes where LO falls within a classification system.

Table 3.3: LO's Meta-Data Categories, Attributes, and Description

Category	Attribute	Description
General	Groups the general information that describes LO as a whole.	
	Identifier	Represents a mechanism for assigning a globally unique label that identifies LO.

	Catalog	Represents the name or designator of the identification or cataloging scheme for LO. There are a variety of cataloging systems available. Some types of cataloging systems: ▪ Universal Resource Identifier (URI) ▪ Universal Resource Name (URN) ▪ Digital Object Identifier (DOI) ▪ International Standard Book Numbers (ISBN) ▪ International Standard Serial Numbers (ISSN)
	Title	Name given to the LO.
	Language	Primary language or languages used in LO .
	Description	Textual description of LO.
	Keyword	Define common keywords that describe LO.
	Coverage	Describe the time, culture, geography or region to which the LO applies.
	Structure	Describe the underlying organizational structure of LO. Values are: ▪ Atomic: Object that is indivisible. ▪ Collection: Set of objects with no specified relationship between them. ▪ Networked: Set of objects with relationships that are unspecified. ▪ Hierarchical: Set of objects whose relationships can be represented by a tree structure. ▪ Linear: Set of objects that are fully ordered.
	Aggregation Level	Describe the functional granularity of LO. Values are: 1: The smallest level of aggregation, e.g., raw

		media data or fragments. 2: A collection of level 1 LOs, e.g., a lesson. 3: A collection of level 2 LOs, e.g., a course. 4: The largest level of granularity, e.g., a set of courses that lead to a certificate.
Life Cycle	Groups the features related to the history and current state of LO and those who have affected the component during its evolution.	
	Version	Describes the edition of LO.
	Status	Describe the completion status or condition of LO. Values are: ▪ Draft: The component is in a draft state (as determined by the developer). ▪ Final: The component is in a final state (as determined by the developer). ▪ Revised: The component has been revised since the last version. ▪ Unavailable: The status information is unavailable.
	Contribute	Describe those entities (i.e., people, organizations) that have contributed to the state of LO during its lifecycle.
	Role	Defines the kind or type of contribution made by the contributor (identified by the Entity element). Values are: ▪ author ▪ publisher ▪ unknown ▪ initiator ▪ terminator ▪ validator ▪ editor ▪ graphical designer

		▪ technical implementer ▪ content provider ▪ technical validator ▪ educational validator ▪ script writer ▪ instructional designer ▪ subject matter expert
	Date	Identifies the date of the contribution.
Technical	Describes all of the technical characteristics and requirements of LO.	
	Format	Represents the technical datatype(s) of all of the components used in the makeup of the LO.
	Size	Represents the size of LO in bytes.
	Location	Specifies the location of LO. .
	Requirement:	Expresses the technical capabilities necessary for using LO.
	Type	Represents the technology required to use LO (e.g., hardware, software, network, etc.). Vocabulary token include: ▪ Operating system ▪ Browser
	Name	Represents the required technology to use LO.
	Minimum Version	Represents the lowest possible version of the required technology to use LO.
	Maximum Version	Represents the highest possible version of the required technology to LO.
	Installation Remarks	Used to represent any specific instructions on how to install LO.
	Other Platform Requirements	Used to represent information about other software and hardware requirements.
	Duration	Represents the time a continuous LO takes

		when played at intended speed. This element is useful for sounds, movies, simulations and the like.
Educational	Describes the key educational or pedagogic characteristics of LO. This category is typically used by teachers, managers, authors and learners.	
	Interactivity Type	Represents the dominant mode of learning supported by LO. Vocabulary tokens: ▪ **Active**: Active learning (e.g., learning by doing) is supported by content that directly induces productive action by the learner. ▪ **Expositive**: Expositive learning (e.g., passive learning) occurs when the learner's job mainly consists of absorbing the content exposed to them. ▪ **Mixed**: A blend of active and expositive interactivity types.
	Learning Resource Type	Represents the specific kind of LO. Vocabulary tokens: ▪ Exercise ▪ Simulation ▪ Questionnaire ▪ Diagram ▪ Figure ▪ Graph ▪ Index ▪ Slide ▪ Table ▪ Narrative text ▪ Exam ▪ Experiment ▪ Problem statement ▪ Self-assessment ▪ Lecture

	Interactivity Level	Represents the degree of interactivity characterizing LO. Interactivity refers to the degree to which the learner can influence the aspect or behavior of LO. Vocabulary tokens: ▪ Very low ▪ Low ▪ Medium ▪ High ▪ Very high
	Intended End User Role	Represents the principal user(s) for which LO was designed. Vocabulary tokens: ▪ Teacher ▪ Author ▪ Learner ▪ Manager
	Context	Represents the principal environment within which the learning and use of LO is intended to take place. Vocabulary tokens: ▪ School ▪ Higher Education ▪ Training ▪ Other
	Typical Age Range	Represents the age of the typical end user. Value should be formatted as *minimum age – maximum age*.
	Difficulty	Represents how hard it is to work with or through LO for the typical intended target audience. Vocabulary element: ▪ Very easy ▪ Easy ▪ Medium ▪ Difficult

		▪ Very difficult
	Typical Learning Time	Represents the approximate of typical time it takes to work with or through LO.
	Description	Used to comment on how the LO.
	Language	Represents the human language used by the typical intended user of LO.
Rights	Describes the intellectual property rights and conditions of use for LO.	
	Cost	Represents whether the LO requires some sort of payment. Vocabulary tokens: ▪ Yes ▪ No
	Description	Allows comments on conditions of use of LO.
Annotation	Provides comments on the educational use of LO and information on when and by whom the comments were created. This category enables educators to share their assessments of LO.	
	Entity	Identifies the entity created the annotation.
	Date	Identifies the date the annotation was created.
	Description	Used to represent contents of the annotation.
Classification	Describes where LO falls within a particular classification system. Multiple Classification categories may be used.	
	Purpose	Defines the purpose for classifying LO. Vocabulary Tokens: ▪ Discipline ▪ Idea ▪ Prerequisite ▪ Educational Objective ▪ Accessibility Restrictions ▪ Educational Level ▪ Skill Level ▪ Security Level ▪ Competency

	Description	Represent content of classification.

3.7 Exam Management System and Cheating Challenges

One of the utilized e-Learning activities is "Online Assessments". Although Online Assessments are not the only criteria to qualify students, it is still an important feature to be activated. One of the problems that prevent e-Learning from gaining advantages of Online Assessments is "Leak of Assessments". Students search the internet for assessments' questions and answers, and unfortunately they can easily find them. Of course it is the students' choice to either follow those answers or not. No matter how close the instructor to the students, they will not confess "cheating". Online Assessments are not conducted in a secure and supervised environment most of the time. Here, analysis results of online assessment experiment are presented to address online assessment challenges and present solutions. During analysis of the assessments' results, some facts become clear. One of the results that forced the analysis of assessments' data was the noticeable number of students who finished the assessment in less than 10 minutes and acquired more than 30 out of 50 as a mark. The assessment consists of 50 True/False questions. Those questions are very well prepared; some of them are accessible via the resources available from the book author(s), and the rest are prepared internally. It was shocking to find that the number of students that got high grades in an almost "not enough time to read the questions" is high. Luckily, students do not know that the system records start-time and end-time, and it can easily calculate duration, or they would have spent longer times just pretending to be solving the assessment.

3.7.1 Problem Domain Analysis

This section holds the analysis results of the online assessment conducted in the academic year 2010 in Information Systems Department, Faculty of Computers and Information Sciences, Mansoura University, Egypt. Figure 3.12 presents the percentage of students with variant assessment completion times. There are 223 students enrolled in this course with 209 online active users. The number of students who attended the first assessment was 182. Students are classified into 7 groups based on the assessment time as presented in table 3.4. The strange notice was that almost two third of the students conducted the assessment in less than 20 minutes.

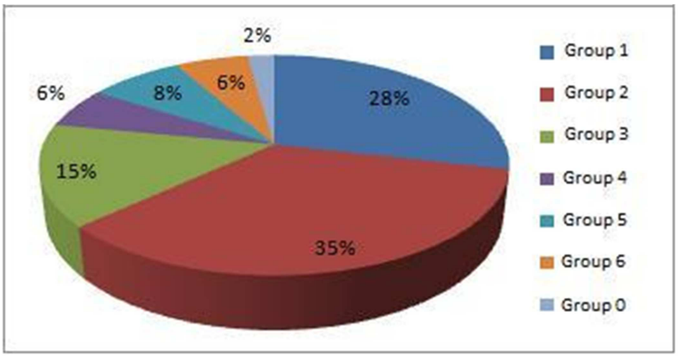


Figure 3.12: Percentage of Students per Assessment Time

Table 3.4: Different Students Groups in this Section

Group 0	Students started but did not complete the assessment and will not be mentioned anymore in this study
Group 1	Students conducted the assessment in duration less than 10 minutes.
Group 2	Students conducted the assessment in duration between 10 and 20 minutes.
Group 3	Students conducted the assessment in duration between 20 and 30 minutes.
Group 4	Students conducted the assessment in duration between 30 and 40 minutes.
Group 5	Students conducted the assessment in duration between 40 and 50 minutes.
Group 6	Students conducted the assessment in duration between 50 and 60 minutes.

To verify the situation, the marks average of each group was calculated and again the results clearly indicate something that is not as "planned to be" situation. Figure 3.13 depicts the average of the six different groups with the notice that averages are almost the same. That means there are students who solved the assessment in less than 10 minutes with marks close to - and may exceed sometimes - those who solved it in almost an hour. To be sure about the grading issue, further analysis to the results was applied with the result that: Number of students from all groups who scored between 0 and 10 out of 50 is (zero). The number of students from all groups who scored between 10 and 20 is only (one). Figure 3.14 shows the different counts of different groups for marks between 20 and 30. Figure 3.15 and figure 3.16 shows the different number of students with marks between 30 and 40, and 40 and 50 respectively.

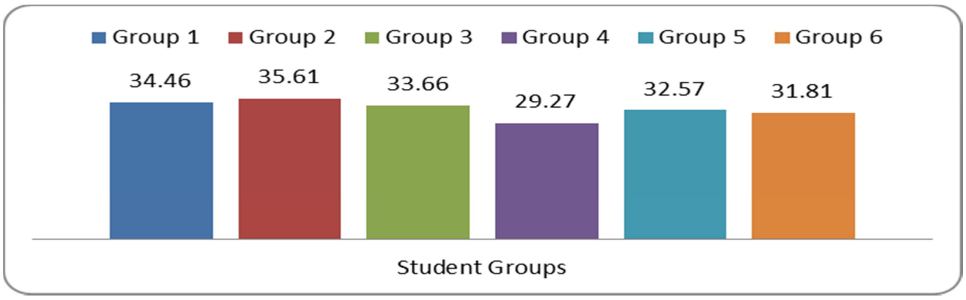


Figure 3.13: Bar Graph of Marks Average per Different Student Groups

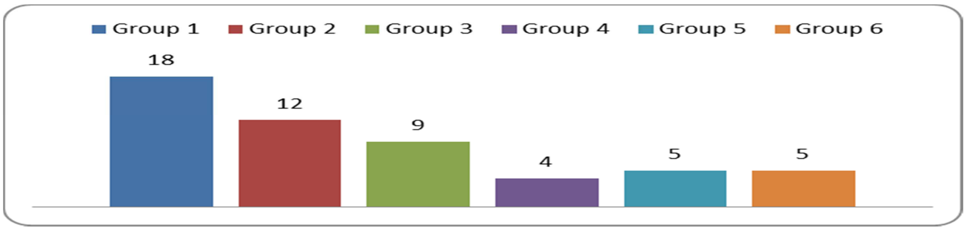


Figure 3.14: No. of Students Achieving Grade Range from 20 to 30 Categorized by Group

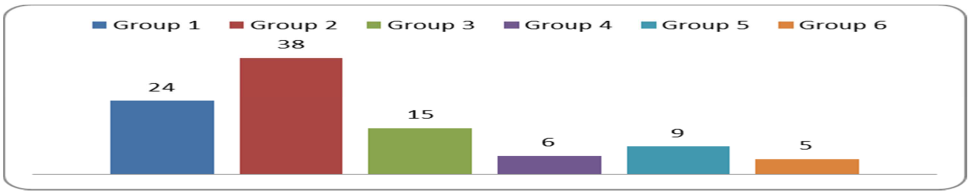


Figure 3.15: No. of Students Achieving Grade Range from 30 to 40 Categorized by Group

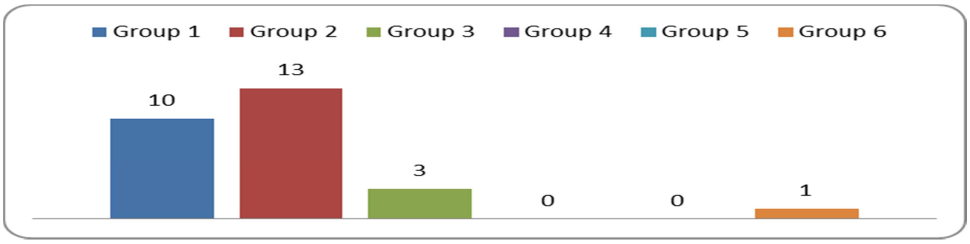


Figure 3.16: No. of Students Achieving Grade Range from 40 to 50 Categorized by Group

3.7.2 Comparative Study between Supervised and Non-Supervised Quizzes

To take a further look on the problem, six different quizzes was conducted in the academic year 2010 to compare between students' performance in both supervised and non-supervised quizzes. Each quiz was conducted twice, once in a supervised environment, and another in non-supervised environment. Students behavior and scored marks are recorded. Table 3.5 presents further details on the types of those quizzes. Table 3.6 presents groups distribution for each quiz.

Table 3.5: Statistics about the Six conducted quizzes in 2009-10

	1st Quiz	2nd Quiz	3rd Quiz	4th Quiz	5th Quiz	6th Quiz
Quiz Title	Quiz 1	Quiz 2	Quiz 3	Quiz 4	Quiz 5	Quiz 6
Total Marks	50	50	25	50	50	50
Total No. of Questions	50	50	20	50	50	50
Types of Questions	T/F	MCQ	Match	Mix	Mix	Mix
Overall Difficulty Level	Med.	Med.	Med.	Med.	Med.	Med.
T/F Questions	50	0	0	20	20	20
Easy T/F Questions	5	0	0	2	2	3
Medium T/F Questions	35	0	0	14	17	17
Hard T/F Questions	10	0	0	4	1	0
Multi Choice Questions	0	50	0	10	10	10
Easy MCQs	0	3	0	0	0	1
Medium MCQs	0	39	0	8	9	8
Hard MCQs	0	8	0	2	1	1
Match Questions	0	0	20	20	20	20

Table 3.6: Detailed Quizzes' Statistics

	1st Quiz		2nd Quiz		3rd Quiz		4th Quiz		5th Quiz		6th Quiz	
	On-line	Lab	On-line	Lab	On-line	Lab	On-line	Lab	On-line	Lab	On-line	Lab
Total	212	212	212	212	212	212	212	212	212	212	212	212
Enrolled	167	46	170	45	166	50	160	69	155	66	153	65
Time Avg.	24.8	16.8	26.4	21.4	7.8	3.6	22	15	17	15	15.4	12.7
Marks Avg.	39.5	29.9	39	26	22	15.7	35	26	43.6	30	42.8	21.8

Group 0	29	0	21	0	5	2	14	0	14	0	13	0
Group 1	23	10	22	2	130	38	36	21	37	20	44	29
Group 2	40	27	33	20	23	10	41	27	63	32	66	24
Group 3	34	5	37	18	7	0	37	17	25	10	17	10
Group 4	13	1	26	4	1	0	13	4	7	3	8	1
Group 5	14	2	20	1	0	0	7	0	5	1	3	1
Group 6	14	1	11	0	0	0	12	0	4	0	2	0

Figure 3.17, 3.20, 3.23, 3.26, 3.29, 3.31 presents groups' distribution for the six quizzes respectively. Figures 3.18, 3.21, 3.24, 3.27, 3.30, 3.32 present time comparison between supervised and non-supervised students for the six quizzes respectively. Figures 3.19, 3.22, 3.25, 3.28, 3.31, 3.33 present the scored marks comparison between supervised and non-supervised in the six quizzes respectively. Analyzing students' consumed time and scored marks for the six quizzes, and comparing both the supervised and non-supervised environments results in two noticeable things: students spend longer times in online (non-supervised environments), and score higher marks in online quizzes. That might be used as an indication of cheating. To take a closer look on the issue, a closer study of the intersection students between both supervised and non-supervised students is presented. 24 students fall in the intersection.

Figures 3.35, 3.36, 3.37, 3.38, 3.39, 3.40 show comparison for intersecting students for each quiz respectively comparing: Online Time, Online Marks, Lab Time, Lab Marks. Students spend longer times in online quizzes and score higher marks when compared to lab times. Tracking students' behavior intelligently can be useful in identifying cheating incidences and notifying instructors' of their occurrences.

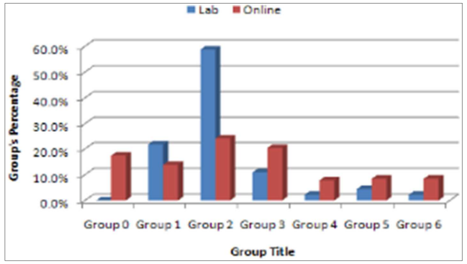


Figure 3.17:

Quiz 1 Groups' Percentage Comparison

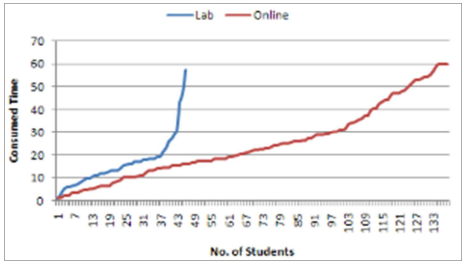


Figure 3.18:

Quiz 1 Time Consumption Comparison

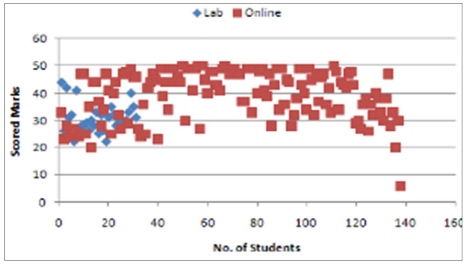


Figure 3.19:

Quiz 1 Scored Marks Comparison

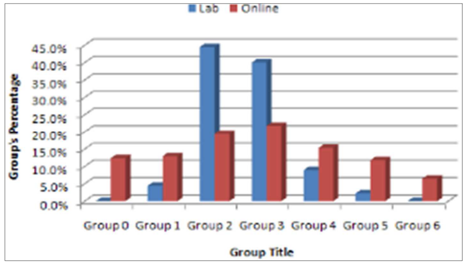


Figure 3.20:

Quiz 2 Group's Percentage Comparison

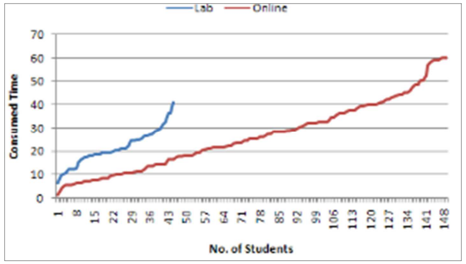


Figure 3.21:

Quiz 2 Time Consumption Comparison

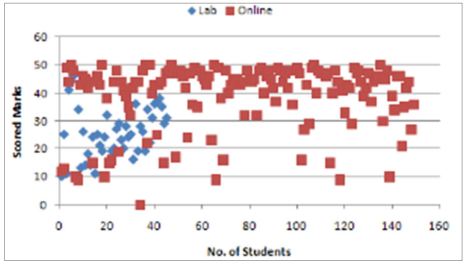


Figure 3.22:

Quiz 2 Scored Marks Comparison

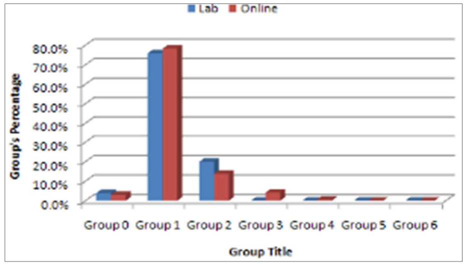


Figure 3.23:

Quiz 3 Group's Percentage Comparison

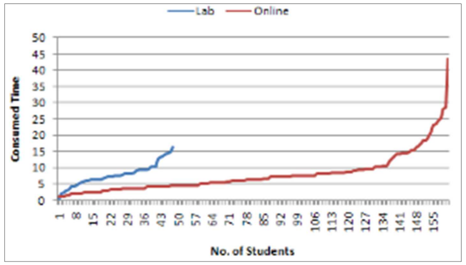


Figure 3.24:

Quiz 3 Time Consumption Comparison

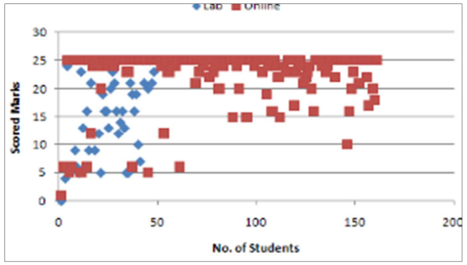


Figure 3.25:

Quiz 3 Scored Marks Comparison

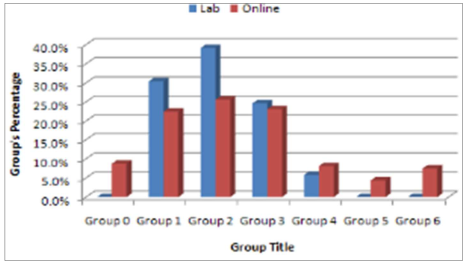


Figure 3.26:

Quiz 4 Group's Percentage Comparison

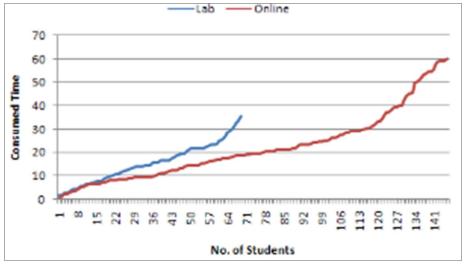


Figure 3.27:

Quiz 4 Time Consumption Comparison

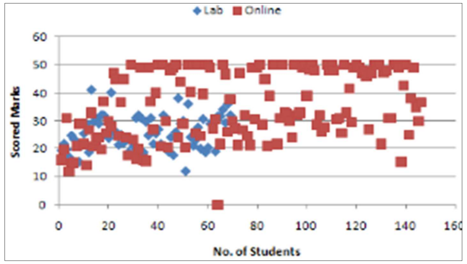


Figure 3.28:

Quiz 4 Scored Marks Comparison

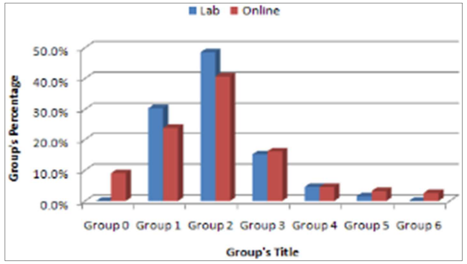


Figure 3.29:

Quiz 5 Group's Percentage Comparison

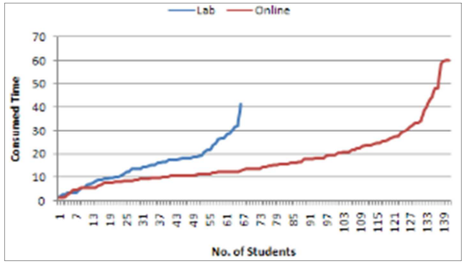


Figure 3.30:

Quiz 5 Time Consumption Comparison

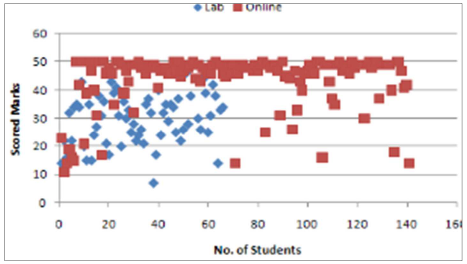


Figure 3.31:

Quiz 5 Scored Marks Comparison

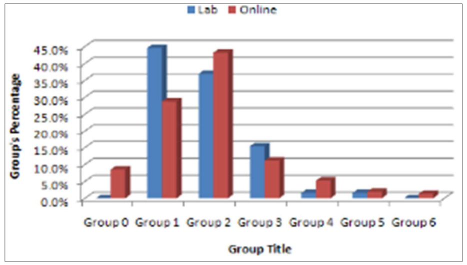


Figure 3.32:

Quiz 6 Group's Percentage Comparison

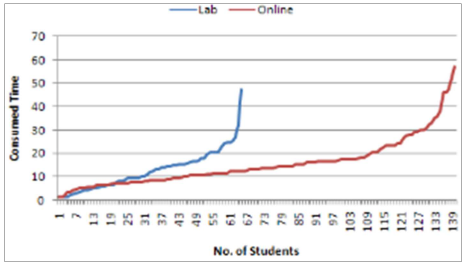


Figure 3.33:

Quiz 6 Time Consumption Comparison

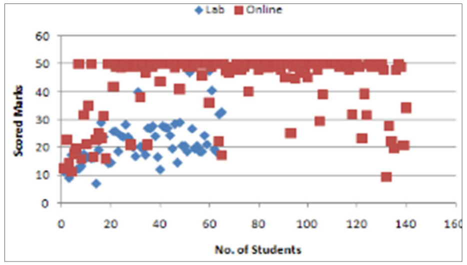


Figure 3.34:

Quiz 6 Scored Marks Comparison

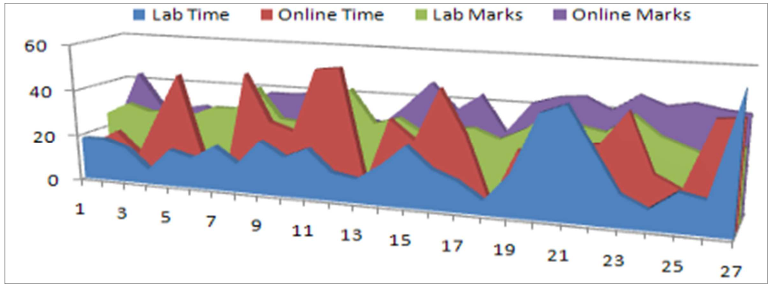


Figure 3.35: Quiz 1 Defined Intersection Students

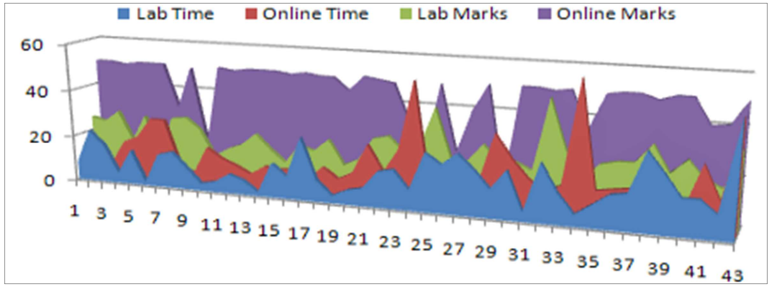


Figure 3.36: Quiz 2 Defined Intersection Students

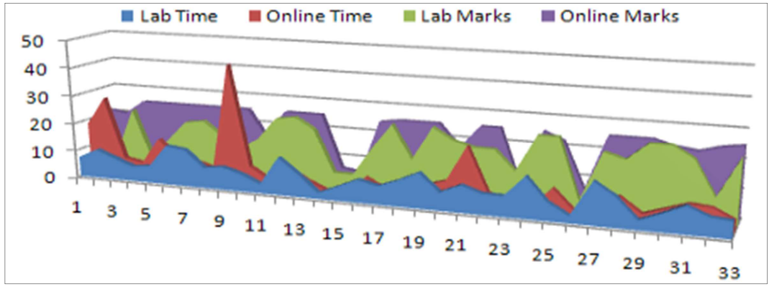


Figure 3.37: Quiz 3 Defined Intersection Students

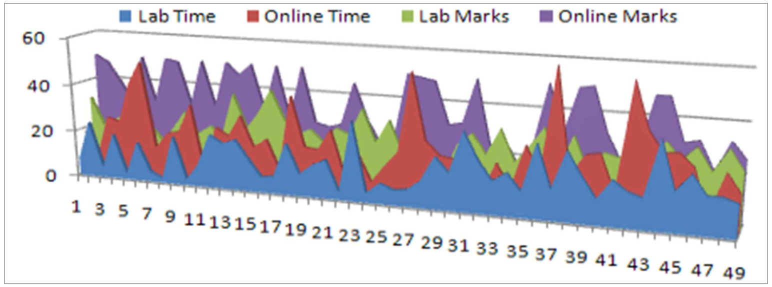


Figure 3.38: Quiz 4 Defined Intersection Students

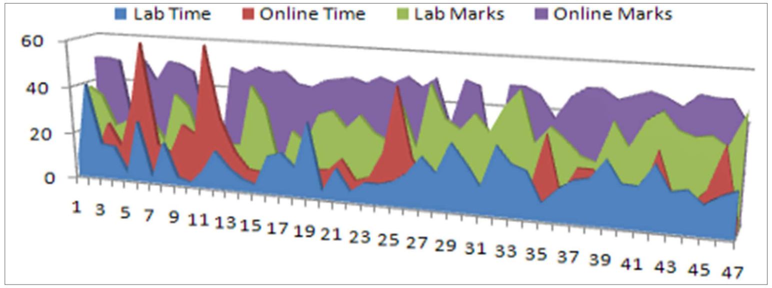


Figure 3.39: Quiz 5 Defined Intersection Students

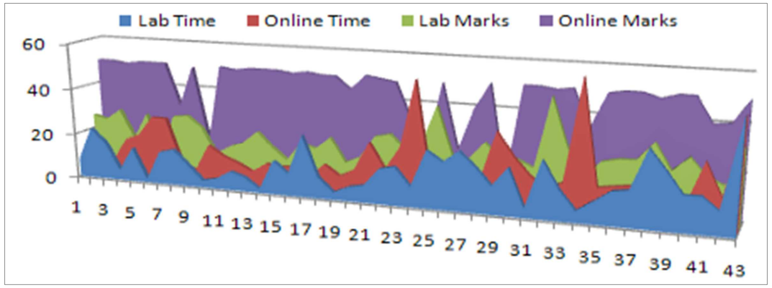


Figure 3.40: Quiz 6 Defined Intersection Students

3.7.3 Comments on Results

Here are some cheating tips that have been witnessed during students' monitoring and feedback:

- **Access to Answers' Files:** Open the assessment PDF or document file, search for keywords, and immediately apply answers. Most students have high memorable capabilities regarding mapping questions and answers.
- **Collaborative Solution:** Though collaboration is really important in the learning process, this way of collaboration to cheat was really new. More than one student conducts the assessment. One holds the laptop, others hold different pages of the assessment answers, so they optimize the search time, and the one holding the laptop says the question loudly and of course students can find answers in no time. Of course in Non-Supervised e-Learning environment, there is no way to guarantee that students themselves attended the assessment.

Two categories of students must not be neglected to assure certain learning quality level:

- **Careless Students:** They do not really need to read the assessment questions. They only pick an answer and they don't care about the results. There are students who answered 50 questions in less than 3 minutes, which gives them an average reading of 3.6 seconds for each question. Another form of carelessness was presented in the 4 students (out of 182 that is almost 2%) who did not finalize the assessment.
- **Not Interested Students:** Almost 23% of the course enrolled students (41 students out of total 223 students) did not attend any of the e-Learning activities. This percentage is huge, and in our course, it is not acceptable at all. However, motivating students to attend Learning activities is always a challenge.

3.7.4 Proposed Solution to Cheating Problems

Based on results presented in the aforesaid section, it is clear that there are issues that shall be considered before providing students with online exams. There must be a stronger way of controlling the Exam process in order to make marks more trustable. Proposed Solution tips to this issue are many. More studies about efficiency and effectiveness of each one need to be conducted and further analyzed and studied. Those actions can be categorized into two categories: Educational and Technical solutions. The solution includes Pedagogical and Technical aspects.

3.7.4.1 Pedagogical Proposed Solution Aspects

Pedagogical Solutions include the attempt to present an unlimited Assessment Items Repository, and to track the students' progress during the learning process, so peaks can be determined and be a mark for inappropriate activity during the learning process. Also, a timed question is almost a must in the exam process. Timer shall not only start after the student sees the question, we are thinking about calculating time for both displaying and solving the question. Therefore, theoretically, students shall never find time to cheat. This study proposes some tips that can be used as solutions that focus on four aspects of the online assessment process and can be thought of as the integration of the all:

- **Questions Based Solution:** Assessments banks should consist of larger number of questions with the chance to have 1/4 or 1/3 the assessment different from one student than the other. Also, instructors should work on updating assessments banks and keeping them out of reach for students.
- **Environment Based Solution:** This solution is complimentary to the previous suggested one. Supervised e-Learning environments are important, and simply are the only way to guarantee certain accepted level of learning quality. Students can find the time to search the answer files because they can

access them easily. Hopefully when students do not have access to such files, they might learn better.

- **Assessment Based Solution:** Timer that forces students to read questions before viewing the answers might be a good idea. Maybe by forcing students to wait for answers before selecting one of them will be a catalyst for the student to read the questions and all the answers. Though this is not a guarantee, but it might be a good way to do so.
- **Student Based Solution:** It is important to discuss with students the importance of e-Learning activities in the learning process, and the gains they can easily acquire and make use of via utilizing such activities. The attempt to qualify students' culture with e-Learning is important to start gaining e-Learning advantages. Yet, not all students believe in e-Learning; only 182 out of 223 cared about attending the online course activities. The rest needs to be talked to instead of neglecting them.

Most students do their best to play it smart, even if they will not follow the rules. Solutions to guarantee learning efficiency and effectiveness for current situations must be thought of about regularly. Unfortunately, students usually advance instructors in utilizing technology for their purposes, which might be "cheating". We - as instructors - need to evaluate the situation regularly and rely more on student performance analysis tools to find out the unclear facts.

3.7.4.2 Technical Aspects of the Solution

Technical Solutions are a real challenge. There is no Web based assessment system that presented a clear solution to such a problem. The solution lies in a well-controlled desktop application that must be used in the Exam process. Desktop applications provide techniques that are not available via Web based systems. Those techniques include:

- **Keyboard Hooking:** The desktop application can control keyboard strikes on system basis and not on application basis. So, we can control which keys are available for students to click and which are not. However, such solution is applicable for Microsoft Windows based Desktop Applications only, as Java Virtual Machine (JVM) does not provide such control over the Operating System, and that will stop authors from developing a platform independent Exam Desktop Application.
- **Operating System Log File:** The desktop application can check the Operating System Log file, and when it finds that the student executed any of the non-authored applications during the exam, it exits the exam. However, students can be smart enough to use two computers during the exam: one for taking the exam, and another for cheating. Besides, checking the Log file will be a time based process that is not guaranteed to take place anytime.
- **Check Running Processes:** The desktop application will check the running processes on the system before and during the exam, and will exit any non-exam required process that is running during the exam. This technique seems to be the most appropriate technique; however, building this list of processes will take time and effort.

By combining the above mentioned techniques - both pedagogical and technical - better circumstances during exams will be achieved.

3.8 Summary

Adaptive Online Lecture Model is presented to help instructors present an enhanced e-Learning experience to students. Adaptive Online Lecture tends to engage students in lecture activities from the very beginning of the lecture, help instructors prepare lecture by recommending the most suitable set of LOs to be used in the lecture. Adaptive Online Lecture also encourages students to submit informal feedback on the lecture activities, and provides an infrastructure to collect, analyse, and generate reports to instructors to be used to fine tune the lecture. IT architecture needed to enable presented adaptive models is discussed in details in this chapter addressing the needed server components, and how to integrate all of them using SOA. SOA is the design pattern that fits the integration among different systems task, and it is addressed in this chapter through the layered architecture. Detailed mapping between the different layers, servers, and presented systems and databases is presented highlighting services dependencies.

The chapter addresses the online assessment challenges that e-Learning faces. Challenges include cheating as the main threat to online assessments efficiency. By studying the behaviour of students on an online assessment, chapter ends with both pedagogical and technical recommendations to overcome the online assessments challenges.

Chapter 4

Chapter Four

Proposed Intelligent Features

4.1 Introduction

Presented adaptive e-Learning models shed lights on supporting e-Learning with intelligent features. Intelligence can be addressed in different aspects of the proposed models. This chapter presents detailed design of the intelligent features to improve students' performance and help instructors through the e-Learning process. Those goals can be achieved by applying different technologies available to educational institutions, instructors and students in an innovative way. Different intelligent services are presented to enable the intelligent features. Generally, an intelligent service is presented for each intelligent feature. Presented intelligent services can be grouped into two categories based on their users. Figure 4.1 presents the two proposed intelligent services' categories: Instructor Intelligent, and Student Intelligent Services.

The Instructor Intelligent Services are:

- Intelligent Learning Object (LO) Classifier
- Intelligent Online Lecture LOs Advisor
- Intelligent Student Tracker
- Intelligent Cheat Depressor

The Student Intelligent Services are:

- Intelligent Study Plan Advisor
- Intelligent Time-to-Learn Topic Calculation
- Intelligent LOs Recommender
- Intelligent Agenda Study Time Planner
- Intelligent Meeting Manager for Suspended Students

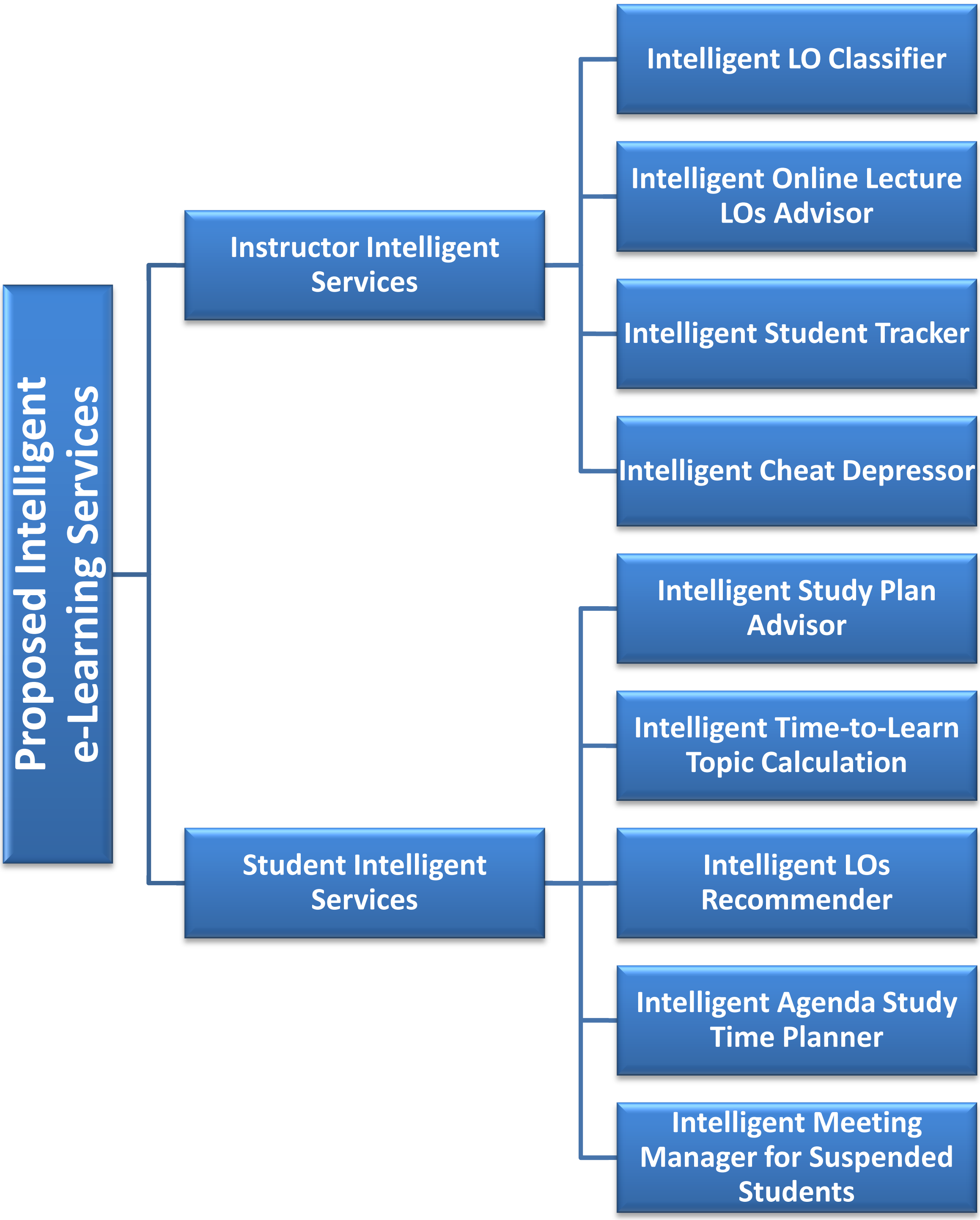


Figure 4.1: Proposed Intelligent e-Learning Features

From Adaptive e-Learning Model, intelligent features are:

- Students are grouped. Each group is delimited by the same start date. Students that do not catch this start date are delayed to the next group, which is 15 days later.
- Crawlers keep searching the internet for newly and updated Learning Objects, besides instructors add different resources to the LOs repository. **Intelligent Learning Objects Classifier** is used to classify found LOs.
- Learning goals are identified by instructors. Based on these learning goals, instructors define learning paths. Instructors can make some branching at certain points to give students the flexibility to customize their learning paths. **Intelligent Study Plan Advisor** helps students at those points.
- **Intelligent Time-to-Learn Topic Calculation** is the service that is used to advice students about the time needed to study a certain topic. Based on students study time of previous topics and the available LOs for this topic, this service can intelligently advise students about the study time issue.
- Students attend one or more adaptive online lecture within the same learning goal. Adaptive online lecture makes use of the **Intelligent Online Lecture LOs Advisor** to recommend LOs for the instructor to use during the lecture, based on the students’ learning profiles.
- **Intelligent LOs Recommender** is the intelligent service that will recommend LOs for students based on their learning profiles. Recommended LOs list is approved by the instructor and reordered based on the students’ preferences.
- **Intelligent Agenda Study Time Planner** is used to organize the students’ time tables and organize their different activities by connecting them to different activities available in the university based on their preferences through announcements.

- **Intelligent Student Tracker** service is used to track the students' performances during the online learning journey, and to verify the completeness of students' learning profiles. Peaks and performance degradation in students' performances need to be recorded and studied.
- Learning path is marked by different learning checkpoints. At each checkpoint, students attend an online exam. Those who pass will continue the learning path. Those who fail will have to re-attend the exam within 4 days. If they fail again, they will have to re-attend the exam within 2 days. If they do not pass the third time, they are suspended.
- **Intelligent Meeting Manager for Suspended Students** service is responsible for managing a meeting between an instructor and the suspended student to handle the learning issues that prevent student from coping with student's group. Suspended students drop behind their group.
- **Intelligent Cheat Depressor** service focuses on utilizing intelligent techniques in prohibiting students from cheating. When combining both Intelligent Student Tracking service, and Intelligent Cheat Depressor service, cheating instances might be identified.

4.2 Instructor Intelligent Services

The intelligent services that aim to help instructor through teaching are: Intelligent LOs Classifier, Intelligent Online Lecture LOs Advisor, Intelligent Student Tracker, and Intelligent Cheating Depressor.

4.2.1 Intelligent LOs Classifier

Different intelligent techniques can be utilized in classifying LO based on LO type. Classifying multimedia-based LOs can be via meta-data, tags, and annotations, while classifying text-based LOs can be done through accessing and analysing content. Text classification or categorization is the process of organizing information logically. It can be used in many fields such as document retrieval, routing and clustering. Document classification tasks can be divided into two sorts: supervised document classification where some external mechanism - such as human feedback - provides information on the correct classification for documents. The second sort is unsupervised document classification, where the classification must be done entirely without reference to external information. Presented Intelligent LOs Classifier utilizes two of the supervised document classification algorithms: Naïve Bayes Classifier, and Term Frequency – Inverse Document Frequency (TF-IDF) algorithms. Both belong to probabilistic classifiers.

4.2.2 Intelligent Online Lecture LOs Advisor

Intelligent Online Lecture LOs Advisor accesses students' profiles and learning preferences side by side with data from previous online lectures and course specification data. This service provides the instructor with a recommended list of LOs based on the attending students. This list can be used during the lecture. Table 4.1 presents Intelligent Online Lecture LOs Advisor specifications, and figure 4.2 presents its detailed flow diagram.

Table 4.1: Intelligent Adaptive Online Lecture LOs Advisor Specifications

<table>
<tr><th colspan="2">Input</th></tr>
<tr><td>Student Preferences</td><td>Different learning preferences that identify the student learning behavior are stored. Those preferences are considered for identifying different study plans.</td></tr>
<tr><td>Related LOs Specifications</td><td>LOs satisfy students’ classes by percentage. The higher available LOs that match students’ preferences, the higher this topic is recommended for teaching.</td></tr>
<tr><td>Learning Topics Data</td><td>Data about courses and topics to be pedagogically used in learning scenarios.</td></tr>
<tr><td>Previous Online Lectures Data</td><td>LOs that were used by previous instructors during online lectures for the same topic and students’ feedback for those LOs are important data for this recommendation process.</td></tr>
<tr><th colspan="2">Processing</th></tr>
<tr><td colspan="2">By assigning different Weights to the different inputs, fuzzy logic is used to generate a weighted list summary report. The intelligent advisor does the following:
▪ Identify LOs presented at previous lectures
▪ Fuzzily classify attending students to one of the learning styles
▪ Check the LOs specifications and meta-data
▪ Identify the most suitable LOs to use with attending students</td></tr>
<tr><th colspan="2">Output</th></tr>
<tr><td>Recommendation Report</td><td>Instructor can use this report to identify Los suitability.</td></tr>
</table>

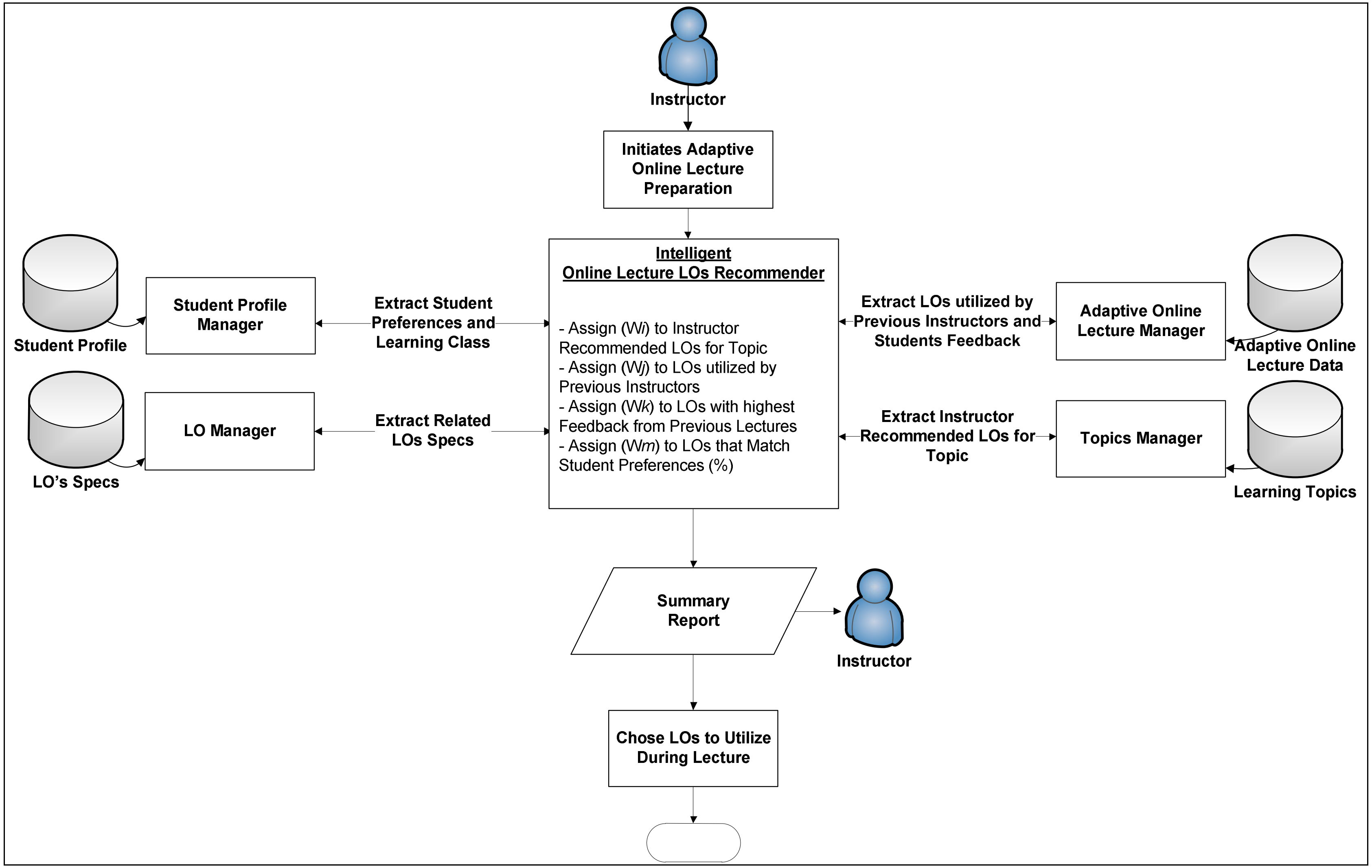


Figure 4.2: Intelligent Online Lecture LOs Advisor Flow Diagram

4.2.3 Intelligent Student Tracker Service

This service tracks student behaviour and assures that there is a complete learning profile that helps the system identifies the student's learning preferences and styles all-over the system. Learning style is the individual's characteristic ways of processing information and behaving in learning situations. Knowledge of learning styles can help instructors better understand learners and have important implications for program planning, teaching, and learning. For each student, there will be four different learning styles available to form the student learning preferences. Student learning styles are:

1. General
2. Felder
3. ATLAS
4. Brain Works

4.2.3.1 General Learning Style

During registration, students are asked to complete general learning style preferences. In case the student selects more than one style, student is asked to rank choices, so later recommendations can define to what extent it is fulfilling the student's requirements. General Learning Profile Preferences are:

- **Visual:** Individuals that learn best when ideas or subjects are presented in a visual format, whether that is written language, pictures, diagrams or videos are visual learners.
- **Auditory:** Individuals who learn best by participating in class discussion, by listening to teacher lecture, listening to audio tapes or by listening to other language formats are Auditory Learners.

- **Tactile:** Tactile Learners are hand-on learners. They learn best when they are able to physically participate directly in what they are required to learn or understand.
- **Logical:** Logical learners like using brain for logical and mathematical reasoning. Logical learners can recognize patterns easily and are good at making logical connections between what would appear to most people to be meaningless content.
- **Social:** Social learners communicate well with others. They are good listeners and are able to understand other's views.
- **Solitary:** Solitary learners tend to be private, introspective and independent. They are able to concentrate and focus on a specific subject, topic or concept without outside help.

4.2.3.2 Felder Learning Style

Felder Learning Model can be identified by promoting the student to answer questions that help identifying the student’s learning preferences. Though Felder identifies that the student is middle between different models, Felder model can help the system to identify the student learning features, and prepare the most appropriate learning environment. Felder Learning Model categories are presented in table 4.2, and they include:

- Active and Reflective
- Sensing and Intuitive
- Visual and Verbal
- Sequential and Global

Table 4.2: Felder Learning Style Categories Description

	Category	Description
Category	Active	Tend to retain and understand information best by doing something active with it, like discussing or applying it or explaining it to others. Tend to work in group.
	Reflective	Prefer to think about things first. Prefer working alone.
Category	Sensing	Like learning facts, solving problems by well-established methods, dislike complications and surprises. Tend to be patient with details and good at memorizing facts and doing hands-on (laboratory) work.
	Intuitive	Prefer discovering possibilities and relationships. Like innovation and dislike repetition. Better at grasping new concepts and are often more comfortable with abstractions and mathematical formulations.
Category	Visual	Remember best what they see: pictures, diagrams, flow charts, time lines, films, and demonstrations. Most people are visual learners.
	Verbal	Get more out of words--written and spoken explanations.
Category	Sequential	Tend to gain understanding in linear steps, with each step following logically from the previous one. Tend to follow logical stepwise paths in finding solutions.
	Global	Global learners tend to learn in large jumps, absorbing material almost randomly without seeing connections, and then suddenly "getting it." Able to solve complex problems quickly or put things together in novel ways, but they have difficulty explaining how did they got it.

4.2.3.3 ATLAS Learning Style

ATLAS Learning Model can be identified by promoting the student to answer questions that help identifying the student's learning preferences. ATLAS learning model categories are:

- **Navigator:** Focussed learners who chart a course for learning and follow it. Initiate a learning activity by looking externally at the utilization of resources that will help them accomplish the learning task and by immediately beginning to narrow and focus these resources. Rely heavily on planning their learning.
- **Problem Solver:** Rely on critical thinking skills. Like Navigators, Problem Solvers initiate a learning activity by looking externally at available resources; however, instead of narrowing the options available, they immediately begin to generate alternatives. They do not do well on multiple-choice tests because these limit divergent thinking.
- **Engager:** Passionate learners who love to learn, learn with feeling, and learn best when they are actively engaged in a meaningful manner with the learning task, the key to learning is engagement.

4.2.3.4 Brain Works Learning Style

Brain Works learning style can be identified by promoting the student to answer questions. Brain Works tries to determine which hemisphere of brain is dominant. It also determines whether the learner react in a more auditory or visual manner. Each of the hemispheres of brain has prescribed functions or specialities. In this manner the brain avoids duplication of function. Hemispheres always work together so that a combination of right and left hemisphere in everything is achieved. There is, however, a tendency for one hemisphere to be dominant. Brain Works learning style categories are:

- Visual vs. Auditory

- Left vs. Right Brain Hemisphere

4.2.4 Intelligent Cheat Depressor Service

Intelligent Cheat Depressor Service tracks students' behaviour in Exams and records both: Students' Marks, and Exam Times trying to identify peaks in marks. Though this service doesn't detect cheat incidents for certain, however it is used as an indicator to the instructor to track certain students. Table 4.3 presents the details of this intelligent process.

Table 4.3: Intelligent Cheat Depressor Service Specifications

Input	
Student's Previous Exam Data	Data include time consumed by student at each exam, type of exam, and mark scored at this exam for previous exams.
Latest Student's Exam Data	Data include time consumed by student at each exam, type of exam, and mark scored at this exam for the latest exams.
Processing	
Utilizing Fuzzy Logic to calculate the cheating susceptibility. Threshold range is allowed incase student's performance is getting better; however peak changes are definitely identified.	
Output	
Instructor Notification	Informing instructor can take a closer look.

4.3 Student Intelligent Services

Student Intelligent Services are: Intelligent Study Plan Advisor, Intelligent Time-to-Learn Topic Calculation, Intelligent LOs Recommender, Intelligent Agenda Study Time Planner, and Intelligent Meeting Manager for Suspended Students.

4.3.1 Intelligent Study Plan Advisor

Intelligent Study Plan Advisor is an intelligent advisory service used to help students identify the appropriate study plans by:

- Identify older study plans
- Identify study plans of colleges

Students differ in their learning behaviour and learning preferences. The Intelligent Study Plan Advisor service considers different students as classes based on their learning preferences. Table 4.4 presents the Intelligent Study Plan Advisor specifications, and figure 4.3 presents the detailed flow diagram. In this service, instructors identify branching capabilities in the learning path where students can have the opportunity to study a learning topic.

Table 4.4: Intelligent Study Plan Advisor Service Specifications

<table>
<tr><th colspan="2">Input</th></tr>
<tr><td>Student Preferences</td><td>Proposed Model stores different learning preferences that identify student learning behavior. Those preferences are considered for identifying different study plans.</td></tr>
<tr><td>Learning Class</td><td>Students are grouped into Classes to ease educational tasks. Classes include: Auditory, Visual, and other classes that are discussed in detail in Learning Profile section.</td></tr>
<tr><td>Study Plans for Previous Students</td><td>Student need to take a closer look on previous instructor plans, grades that students scored by following certain plans, and other data.</td></tr>
<tr><td>Study Plans for Colleges</td><td>What are students in the same groups are studying now</td></tr>
<tr><th colspan="2">Processing</th></tr>
<tr><td colspan="2">To generate the Recommended Study Plan, the system utilizes Fuzzy Logic and does the following:
▪ Identify the class to which the students belong
▪ Check the branching decision that is assigned by instructor for that class, and doubles the weight of this decision
▪ Check the average of branching decision taken by students in the same class
▪ Ranks recommendations from Top-Down based on the generated weights</td></tr>
<tr><th colspan="2">Output</th></tr>
<tr><td>Recommended Study Plan</td><td>Recommended choice to take in the study plan</td></tr>
<tr><td>Study Plan for Colleges as an Information</td><td>Display hints on what colleges are studying, so student is free to follow their path</td></tr>
</table>

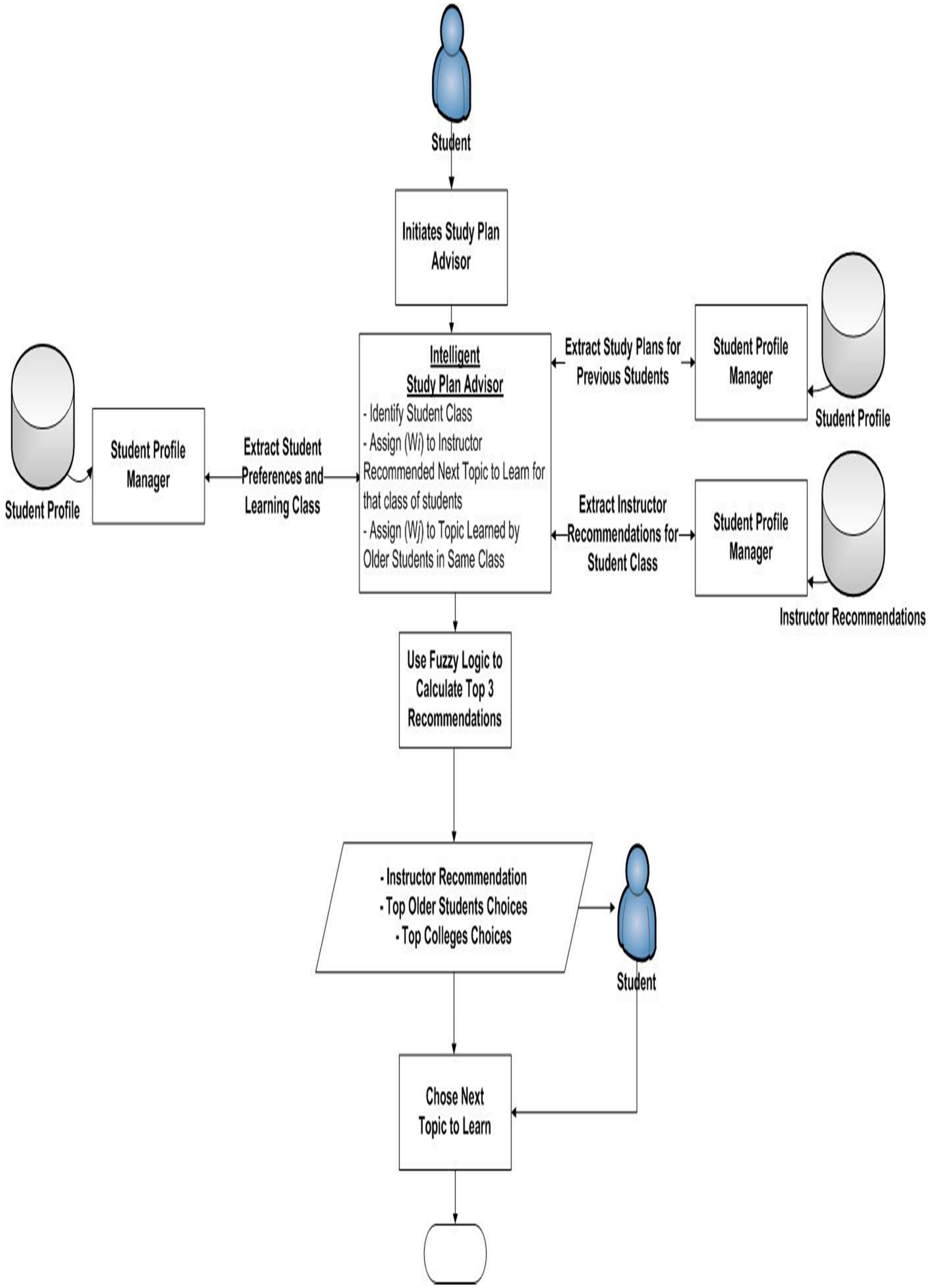


Figure 4.3: Intelligent Study Plan Advisor Flow Diagram

4.3.2 Intelligent Time-to-Learn Topic Calculation

Intelligent Time to Learn Topic Calculation is an intelligent service that helps students identify time needed to learn a certain topic. From study time point of view, the time needed to study a topic is the summation of the time needed to study LOs composing this topic. The instructors define the learning time for each LO as one of the LOs educational meta-data attributes. The system can identify learning time variances between instructors' identified learning time and the students actual consumed learning time through tracking students. This time can help student estimate the time needed to finish studying. Table 4.5 presents the Intelligent Time-to-Learn Topic Calculation specifications, and figure 4.4 highlights the service flow diagram.

Table 4.5: Intelligent Time-to-Learn Topic Calculation Specifications

<table>
<tr><th colspan="2">Input</th></tr>
<tr><td>Instructor Defined Learning Time</td><td>LO's author defines learning time for each LO. Later, different instructors can identify learning times for the same LO to match students' skills.</td></tr>
<tr><td>Student Learning Time Shift</td><td>Tracking the student's learning progress helped the system to calculate the time-to-learn shift between the defined time and the student actual time to learn. Average time to learn calculation will be presented.</td></tr>
<tr><th colspan="2">Processing</th></tr>
<tr><td colspan="2">To calculate Total Time-to-Learn Topic for student, system utilizes Fuzzy Logic in the following process:
▪ System identifies the LOs list the student must learn to finish the topic based on instructor's directions
▪ System identifies the time shift between instructor's identified learning time and the actual time taken by student. Such an entry is identified over time through tracking student.
▪ System estimates time needed for learning each LO, and for all LOs forming Topic, system calculates timed needed to learn that topic.</td></tr>
<tr><th colspan="2">Output</th></tr>
<tr><td>Time To Learn Topic</td><td>Time estimation for student to learn certain topic.</td></tr>
</table>

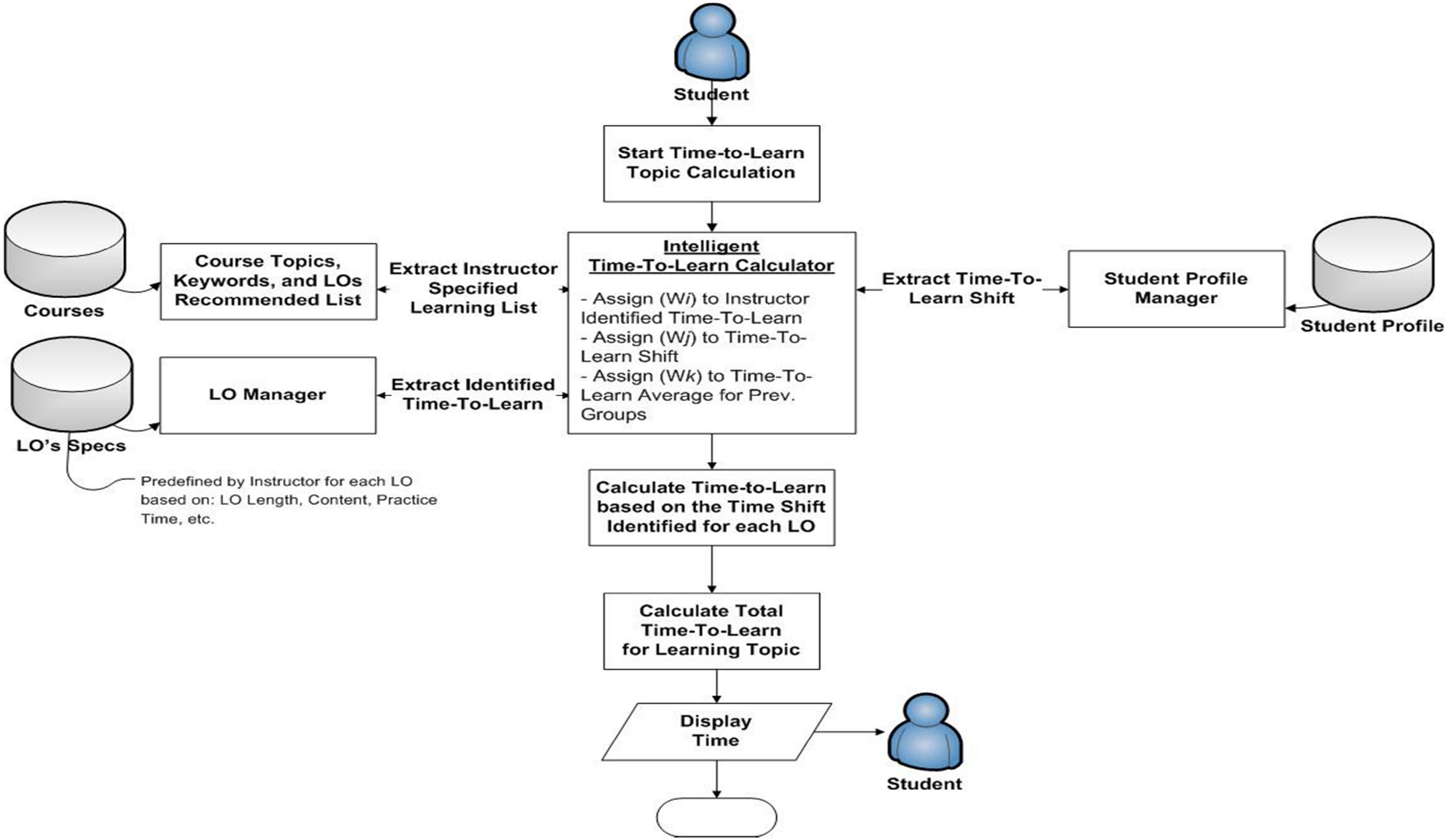


Figure 4.4: Intelligent Time-to-Learn Calculation Flow Diagram

4.3.3 Intelligent LOs Recommender

Intelligent LOs Recommender is the service aims to find the most pedagogically suitable LO for helping student learning a topic, then personalizing the recommended list based on student's preferences. Thus, Intelligent LOs Recommender must efficiently analyze newly introduced LOs, then store information about them for further processing and ordering to each student. From high level view, Intelligent LOs Recommender executes through the two phases presented in Figure 4.5:

- **LOs Finding, Gathering, and Analyzing Phase:** in this phase, system completes different data input resources. Mainly: crawler for supporting open learning environment, digital library data, and students' learning preferences. Web content can be of different types. Audio and video types are identified and handled via annotations that are managed by instructors and learning specialists. Whereas, Information extraction techniques are employed in this process to provide further processing of textual Los.
- **Intelligent Personalized Supervised LOs Recommendation Phase:** in this phase, intelligent fuzzy LOs classifier utilizes fuzzy logic to intelligently discover the degree of relevancy between LOs and specific course specifications. Learning objects that satisfy a specific item of the course with a certain threshold are then recommended. Those recommendations are not considered valid unless they were approved by instructor. After improvement, learning objects are then ranked based on a final score which is a combination of relevancy degrees and user's preferences. Thus, our system guarantees minimum level of pedagogical and learning quality, with a personalization spirit.

Intelligent LOs Recommender phases are interdisciplinary, overloading and it is not easy to set boundaries between them. They must be because they build over each other; they prepare data for processing between them. The two phases are over-viewed in Figure 4.5. Further details about each phase are presented highlighting some of the data and processes required for each phase, and the services that enable the system to achieve those phases.

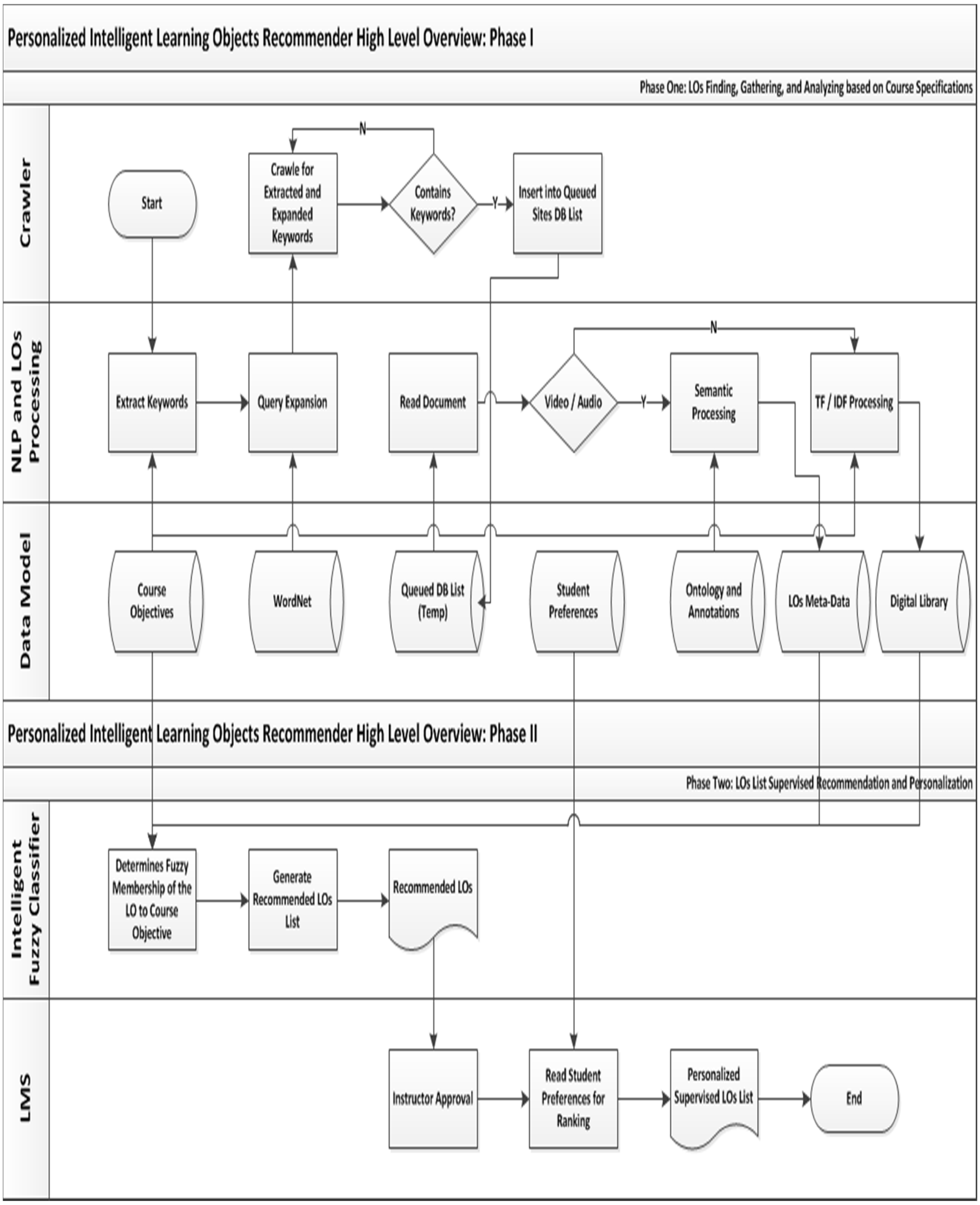


Figure 4.5: Personalized Intelligent Recommender e-Learning Overview

4.3.3.1 LOs Finding, Gathering, and Analyzing Phase “Phase One”:

Extensive amount of data is needed to enable model processing. Different input sources means responsibility in following different input resources and extra effort in completing all input forms. Figure 4.6 presents the steps taken in this phase to ensure as optimal data collection as possible. Four external main data sources for our proposed system must be handled carefully:

- **Internet:** that provides an open learning environment for LOs via Crawler.
- **Digital Library:** that presents different representations of different types of LOs. Digital Libraries provides exclusive information resources that higher in quality and effectiveness when compared to LOs coming from internet.
- **Student Preferences:** in order to be able to personalize the system for each student based on her/his preferences, there shall be a mechanism to inform students with the importance to fill their profiles.
- **Courses' Specifications:** Quality Assurance and Accreditation Project (QAAP) server holds course specifications to be utilized to classify LOs.

The process of Document Processing (divided into two steps) is to make clear the border of each language structure and to eliminate as much as possible the language dependent factors, tokenization, stop words removal, and stemming. Removing stop words and stemming words is the preprocessing tasks. The documents in text classification are represented by a great amount of features and most of them could be irrelevant or noisy. The steps taken for the feature extractions are:

- **Tokenization:** A document is treated as a string, and then partitioned into a list of tokens.
- **Removing stop words:** Stop words such as “the”, “a”, “and”... etc. are frequently occurring, so the insignificant words need to be removed.

- **Stemming word:** Applying the stemming algorithm that converts different word form into similar canonical form. This step is the process of conflating tokens to their root form, e.g. connection to connect, computing to compute.

Next comes the process of text classification. In text classification, a text document may partially match many categories. We need to find the best matching category for the text document. Term Frequency-Inverse Document Frequency (TF-IDF) approach is commonly used to weight each word in the text document according to how unique it is. TF/IDF weights are then fed to fuzzy classifier that specifies to what degree textual content is relevant to a certain category making use of a threshold for more focused and relevant content. Using Fuzzy Logic allows participation of a single learning object in different items in course specifications with different membership degrees.

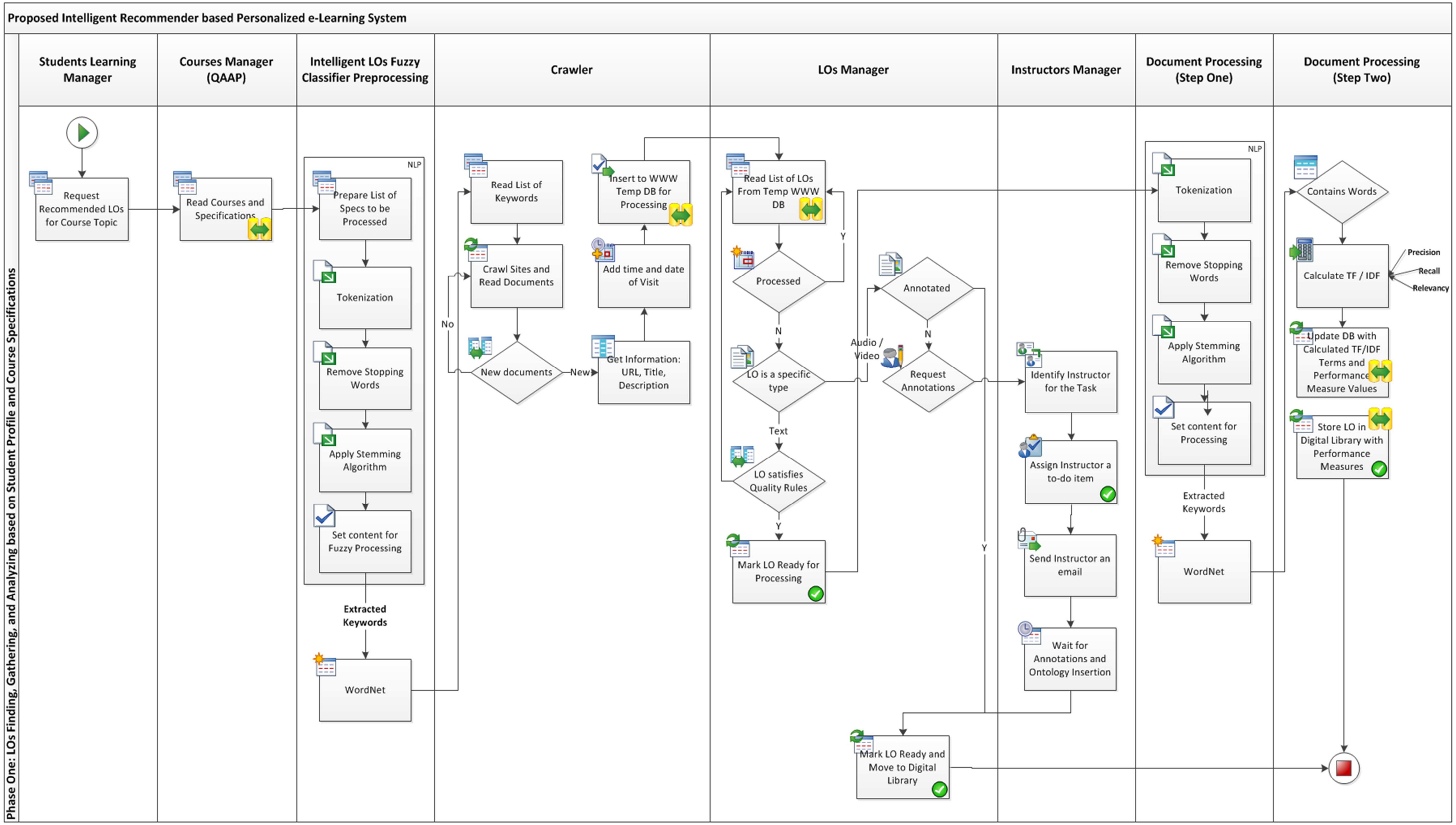


Figure 4.6: LOs Finding, Gathering, and Analyzing "Phase One"

4.3.3.2 Personalized Supervised Generated LOs “Phase Two”

Figure 4.7 presents detailed phase activities. By identifying different information resources, and getting them integrated, Intelligent LOs Recommender can generates a list of LOs that match course specifications based on the former fuzzy classifier. Generated list isn't submitted directly to students to guarantee minimum level of quality and accuracy. This list is appended for instructor to approve / modify it before submitting it. Once instructor approves the list, it becomes available to the student. Second phase entitles ranking those approved LOs based on their relevancy together with their suitability to student's preferences.

4.3.3.3 Fuzzy Logic as Intelligent LOs Classifier and Recommender

Fuzzy Logic is utilized in the two Intelligent LOs Recommender phases: Classification, and Personalization. For system to generate recommendations, it needs to classify a certain Learning Object to an objective in a certain course lecture. However, LO may be related to different objectives, that's why system needs to be flexible enough so that same learning object may belong to different objectives, lectures or courses. That’s why fuzzy classifier was a suitable solution. Fuzzy logic is a form of multi-valued logic derived from fuzzy set theory. In fuzzy set theory, each value represents a degree of probability of membership to a certain set; such value can range between 0 and 1. A Fuzzy set encompass a function to indicate the degree of membership. For the classifier to determine the final membership of LOs, it analyzes both content relevancy as well as user preferences. For any learning to be recommended to user, its membership value must be higher than threshold values. Threshold is used to provide more focused recommendations as can be show in equation 1.

$$\forall\, d \in D, \forall\, o \in O, r = rel(d, o)\ , r \geq \theta$$

- d is a learning object
- D: set of all learning objects
- o: specific lecture objective
- O: set of objectives
- r: the degree of relevancy
- θ threshold values

To make fuzzy logic utilization clear in this context, let's assume that we have Topic (Ta) with set of Keywords = {Ka1,Ka2,…,Kan} and Set of LOs objects={LO1,LO2,…,LOn}. Instructors identify Keywords with relevancy factors, meaning irrelevant keywords have relevancy value of zero, and relevant keywords have relevancy value of one, and other keywords have relevancy values in-between. Recommender utilizes fuzzy logic in identifying the relation between LO and Topic in the following manner:

Step 1: Recommender identifies Keywords Composing Topic

Step 2: For Each Keyword in Keywords:

- Recommender Searches LOs DB for matching LOs. Matching LOs are stored and retrieved within threshold boundaries and membership values.

Step 3: For Each Matching LO:

- Recommender cumulatively and fuzzily calculates the membership of LO to each Topic. Threshold boundaries here, presents LOs as multiply factor of: Percentage of LOs Coverage of this Keyword and Relevancy Factor identified by instructor. LOs are fuzzily evaluated to give weights for instructors' identified relevancy, and LOs coverage of the keyword.
- Recommender returns a non-ordered list of LOs within threshold boundaries.

Step 4: For Each Returned LO:

- Use Fuzzy membership to calculate relationship between LO and learner's class, and use this membership in ordering LOs list.

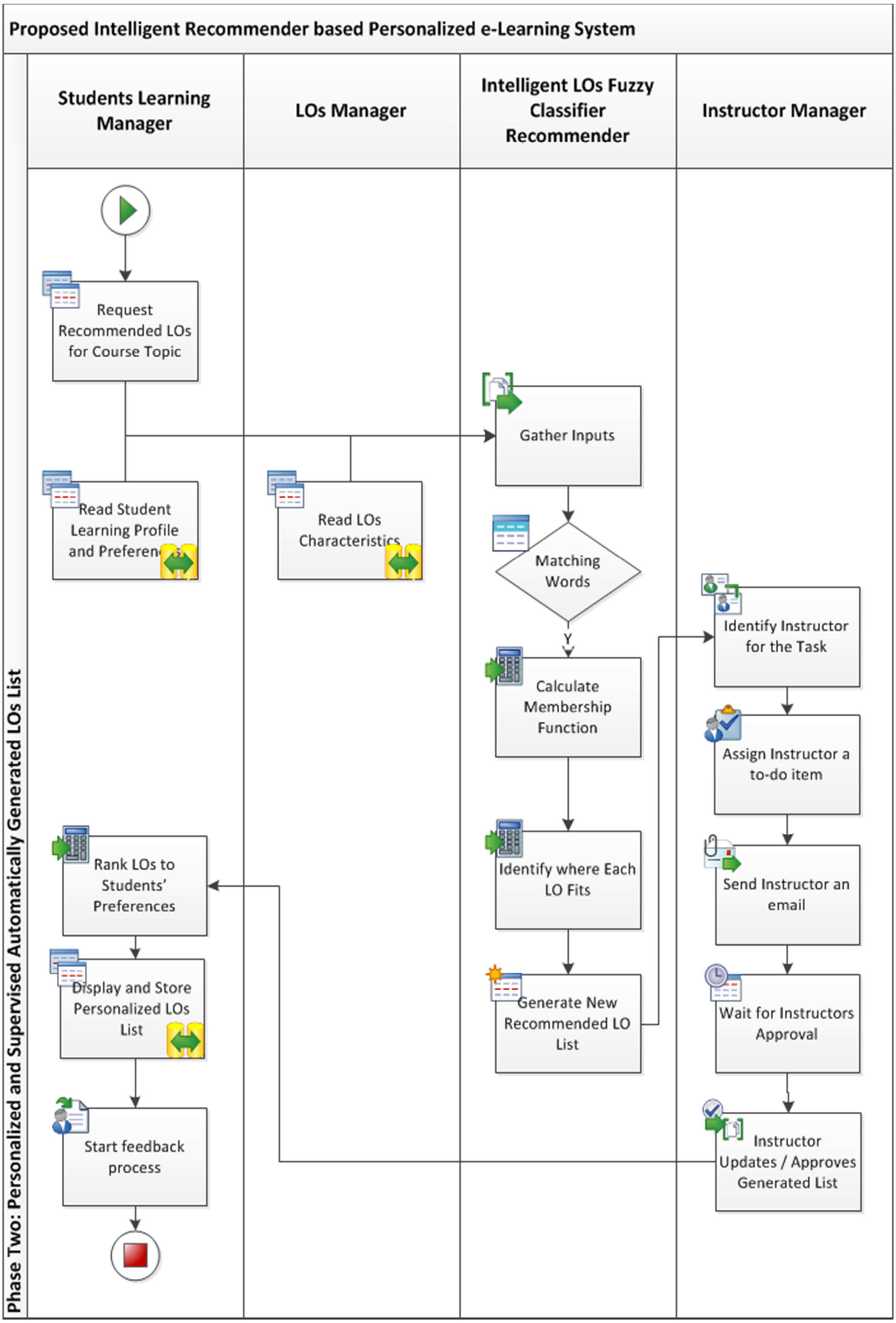


Figure 4.7: Personalized Supervised Automatically Generated LOs List "Phase Two"

4.3.4 Intelligent Agenda Study Time Planner

Helps students identify study times, and integrate activities with their agenda to improve performance. It uses the study time shift fuzzily estimated between the instructor LO study time and student actual study time, can intelligently suggest time needed for students to finish their studies. Besides, it integrates different activities in the university within student's timetable based on students' preferences. It presents students' timetables that combine: lecture times, study times, and activity times, so they are personalized for each student. Table 4.6 presents the Intelligent Agenda Study Time Planner specifications, and figure 4.8 presents detailed flow diagram.

Table 4.6: Intelligent Agenda Study Time Planner Specifications

<table>
<tr><th colspan="2">Input</th></tr>
<tr><td>Student Preferences</td><td>Model stores different learning preferences that identify student learning behavior. Besides, students register in their preferred activities.</td></tr>
<tr><td>Related LOs Specifications</td><td>Specifications of LOs those students shall study, including instructor identified study time.</td></tr>
<tr><td>Study Time Shift</td><td>Fuzzy estimation of the time shift between actual study time identified for each LO and estimation of the actual time the student needs to study this LO.</td></tr>
<tr><th colspan="2">Processing</th></tr>
<tr><td colspan="2">Intelligence in processing takes place in different activities, mainly when conflicts happen. System can resolve conflicts using fuzzy logic as follows:
▪ System identifies LOs list that student has to study, assign them the highest weight value.
▪ System identifies activities available this week that matches student's interest.
▪ System identifies lecture times, and assigns them the highest weight value.
▪ System identifies student free time.
▪ System attempts to suggest a weekly agenda for student to satisfy all of the above. When conflicts take place, fuzzy logic is used to calculate the importance of each entry. Highest priorities take place and overrides low priorities activities.</td></tr>
<tr><th colspan="2">Output</th></tr>
<tr><td>Personalized Agenda</td><td>Personalized agenda for each student that Combines: lecture times, suggested study times for LOs, and activity times.</td></tr>
</table>

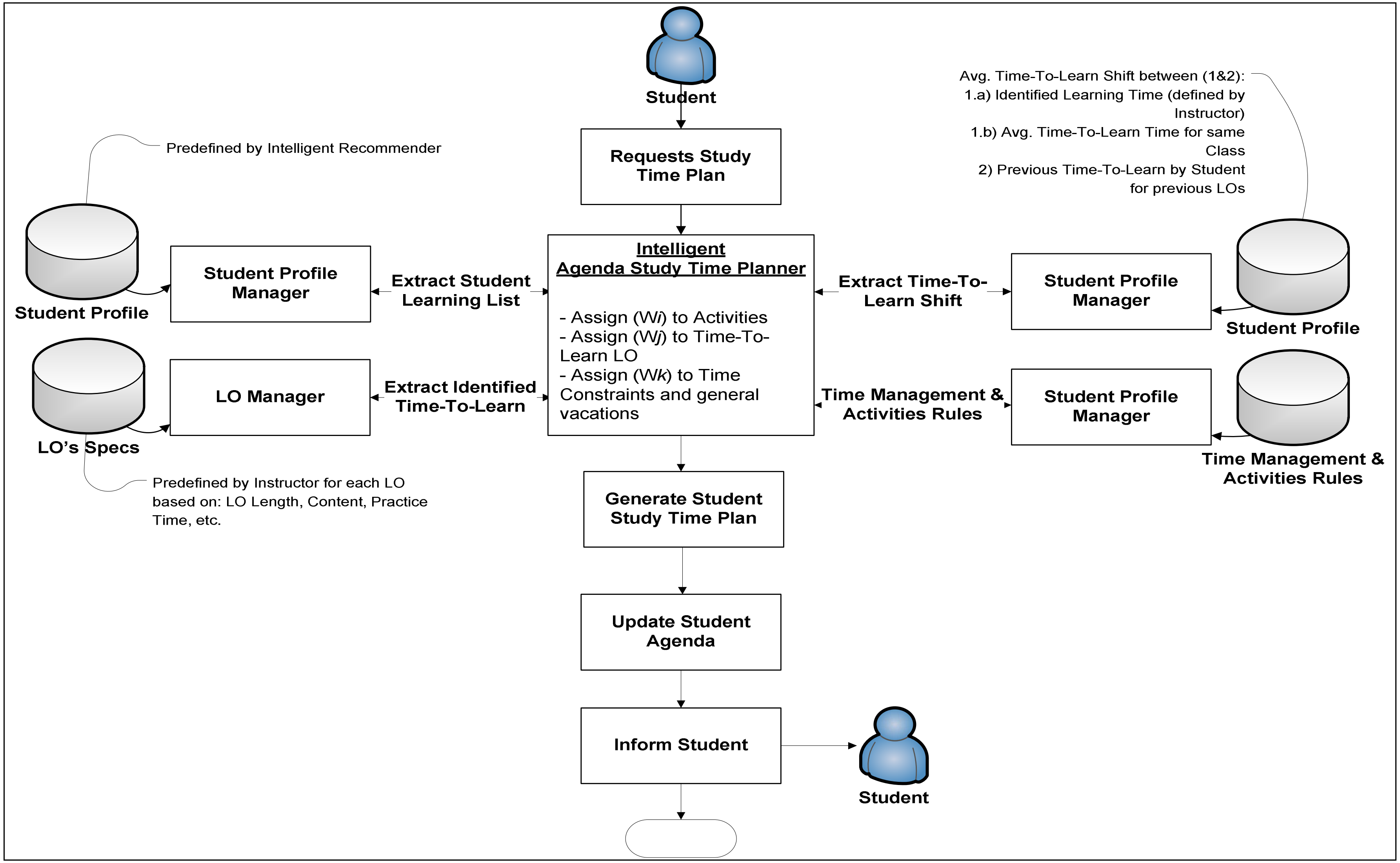


Figure 4.8: Intelligent Agenda Study Time Planner

4.3.5 Intelligent Meeting Manager for Suspended Students

Students who fail three times in passing the exam are suspended from accessing the system. Not being able to pass the exam for three times indicates that there are some pedagogical issues that need taking care of. Suspended students must meet one of the instructors to help them identify and work on solving challenges. Identifying the time for suspended students to meet instructors is an intelligent process that utilizes fuzzy logic to reach the most suitable time for both students and instructors. Table 4.7 presents specifications of Intelligent Meeting Manager for Suspended Students. Figure 4.9 depicts flowchart. Suspended students can't access the system till they are reactivated by the instructor after the meeting.

Table 4.7: Intelligent Meeting Manager for Suspended Students Specifications

Input	
Student Timetable	Proposed Model extracts suitable students' meeting times.
Instructors to Meet	Different instructors can support the same course. Students are able to give priorities for different instructors.
Processing	
By finding matches between student's available time and instructors available time, proposed meeting times are presented. Three different proposed meeting times are presented, and waiting for instructor's approval in order. Arranging meetings faces challenges, especially when there are no free times available. What happens when there is no free time slots are available is that, the system needs to break some time constraints using fuzzy logic to identify what time constraint to break. Instructors must approve meetings before they are sent to student.	
Output	
Proposed Meeting Time	Proposed Meeting is approved by instructor. If approved, student is informed with this meeting and now the instructor has full control on student's status. Instructor can reactivate student, make her/him access LOs, and attend exams if needed.

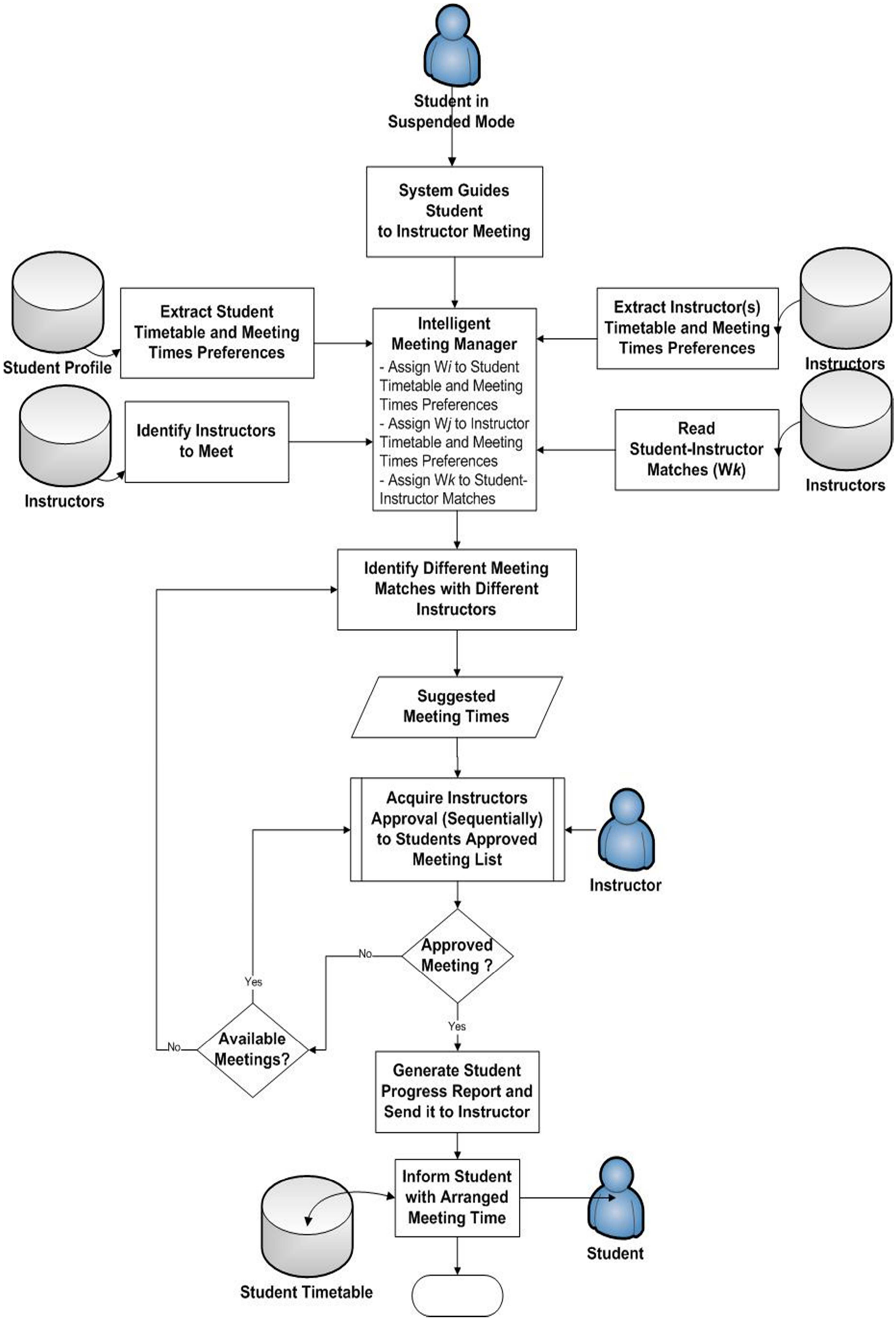


Figure 4.9: Intelligent Meeting Manager for Suspended Students Flowchart

4.4 Summary

Intelligent techniques to empower the proposed adaptive e-Learning models are presented in this chapter. Presented intelligent services utilize fuzzy logic to enable system acts intelligently. Presented intelligent services are grouped into two groups: Instructor, and Student intelligent services.

The Instructor Intelligent Services are:

- Intelligent Learning Object (LO) Classifier: used to classify LOs to categories. Useful with non-classified and new LOs presented to the system.
- Intelligent Online Lecture LOs Advisor: Advices instructors with the most suitable LOs to use during lectures. Considers attending students' preferences in the inputs, and tends to enrich online lectures.
- Intelligent Student Tracker: track the students' performances, and verify the completeness of students' learning profiles.
- Intelligent Cheating Depressor: identifies peaks in students' scores in the attempt to detect cheating instances.

The Student Intelligent Services are:

- Intelligent Time-to-Learn Topic Calculation: help students identify the time needed to finish study requirements for learning topics.
- Intelligent Study Plan Advisor: help students take decisions at course branches based on previous experiences for similar students.
- Intelligent Agenda Study Time Planner: help students manage their timetables by combining the lecture times, activity times, and suggested study times for LOs.
- Intelligent Meeting Manager for Suspended Students: arrange meetings for suspended students with instructors to help them re-join the learning.

- Intelligent LOs Recommender: recommends the suitable personalized LOs for students to study that best matches their learning preferences. Used in fulfilling pre-requisites, and in additional readings and further study.

Chapter 5

Chapter Five

Technical Details and Implementation

5.1 Introduction

This chapter presents deeper technical details and implementation specifications of aspects of the proposed Adaptive e-Learning Models and highlights some of the intelligent services. Upon the start to build the proposed model, some challenges became clear. Challenges include:

- Lack of ability to access Internet in certain occasions. Internet is not available to all students all the time.
- Large size of some learning resources, mainly video lectures.
- LOs copyrights prevent us from uploading them online.
- There is a need to provide meanings of learning to students all the time.

Besides, one of the proposed solutions to cheating problem was conducting exams through a desktop application that provides more secure environment than the Web. Hence, the idea of presenting a Desktop Learning Environment that provides learning features while offline, and integrates with the Proposed Model Adaptive LMS when the application comes online became a clear need.

Technical Implementation consists of three complementary parts:

1. **Student Desktop Application:** available to registered students to download, including instructor recommended LOs for course topics based on student registration information.
2. **Student e-Learning Environment and Adaptive Features:** Adaptive LMS web site where student registers, download the application, update profile and learning preferences, and connects the application so the students' usage and learning data is synchronized automatically, and the

desktop application itself where students attends the Adaptive e-Learning Model and use the different services and features.

3. **Instructor Portal:** the Adaptive LMS administration where instructors manage students, courses, topics, learning objects, instructors' data, and other system configurations that affects the learning process.

SOA design pattern presents system and system components as collection of reusable services with standard interface that can be used within and among different applications, and allows shareability and integration between different systems. In chapter three different adaptive model components were presented, and in chapter four different intelligent services specifications are available. Technical details and implementation of selected services are presented here.

Services are needed to support proposed Adaptive e-Learning model. System Services are services that present one or more of the adaptive and intelligent functionalities proposed by the model, and include services that present functionalities needed to achieve required tasks. Services include: Students Manager Service, Intelligent Student Tracker Service, Learning Objects Manager Service, Intelligent LOs Recommender Service, Intelligent Meeting Manager for Suspended Students Service, and Intelligent Document Classifier.

5.2 Intelligent Student Tracking Service

It is the service that tracks students' status during the learning journey. Figure 5.1 displays student transition state diagram which is one of the states:

- **Complete Profile:** This is the state that a student is in when s/he fills all the learning preferences required fields.
- **Currently Studying:** Incase student has completed learning preferences profile, and is not suspended.
- **Partial Exam Completion:** Student doesn't have a due exam yet.

- **Exam Due:** Student is scheduled to have an exam.

This service provides a standard interface that can be queried to define the state of the student. Input parameters are: authentication token of the calling service, and student ID, and the service response is a number from 1 : 5 indicating the student state. Intelligent Student Tracker Service utilizes Students Manager Service Helper.

Figure 5.2 presents the loading screen students see at running the application. Figure 5.3 highlights the Intelligent Student Tracker in action for new students, when the system identifies that student has not completed the learning profile. Intelligent Student Tracker directs student to complete learning profile.

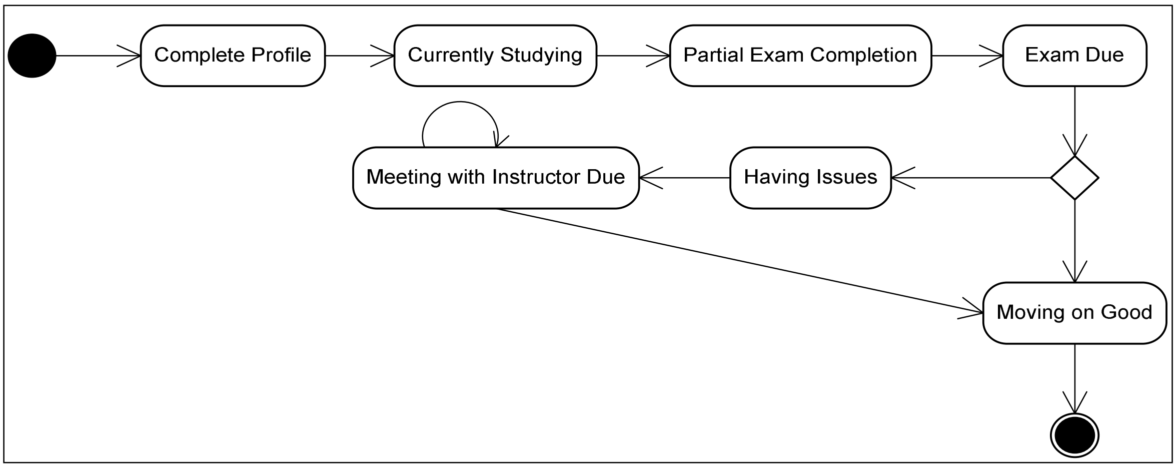


Figure 5.1: Student State Transition Diagram

Figure 5.2: Application Loading Screen

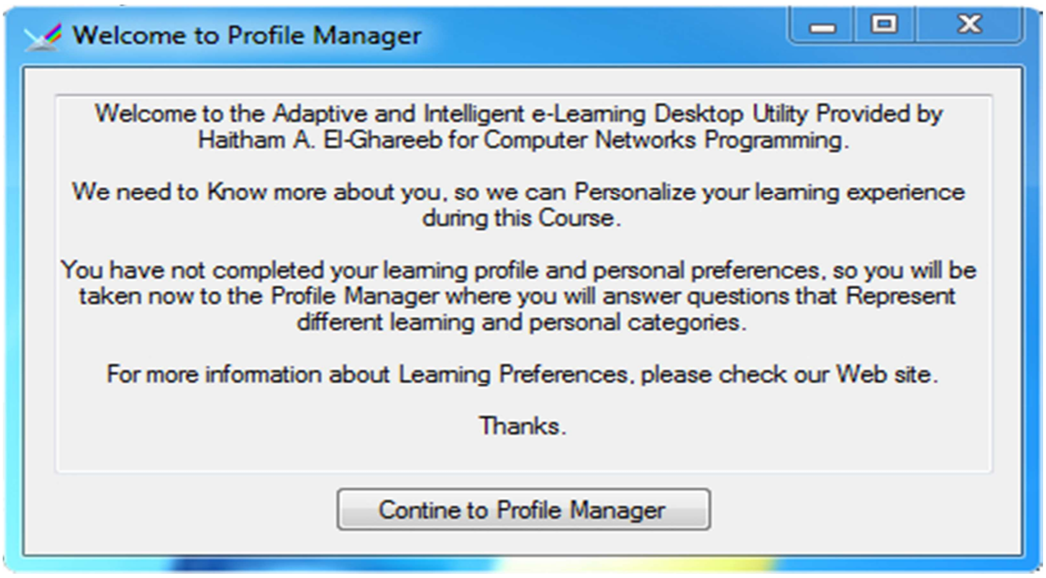


Figure 5.3: Welcome Message indicating the Need to Build Learning Profile

Intelligent Student Tracker service needs two services: Students Manager Service and Students Usage Data Manager Service. Students Manager Service tracks students' data themselves while Students Usage Data Manager Service tracks students' different usage data of the system.

5.3 Students Manager Service

Students Manager Service enables different systems to manage students' data. Students' data include learning preferences, learning profiles, time table, and students' usage data. Students Manager Service include three inner services, they are: Students General Data Manager, Students Learning Profile Manager, and Students Usage Data Manager. While Students General Data Manager handles basic Create, Retrieve, Update, and Delete (CRUD) operations for basic information like username, password, email, and other general data, Students Learning Profile Manager, and Students Usage Data Manager are more of pedagogical importance to the proposed model. Figure 5.4 presents the required database tables to support Students Learning Profile Manager Service.

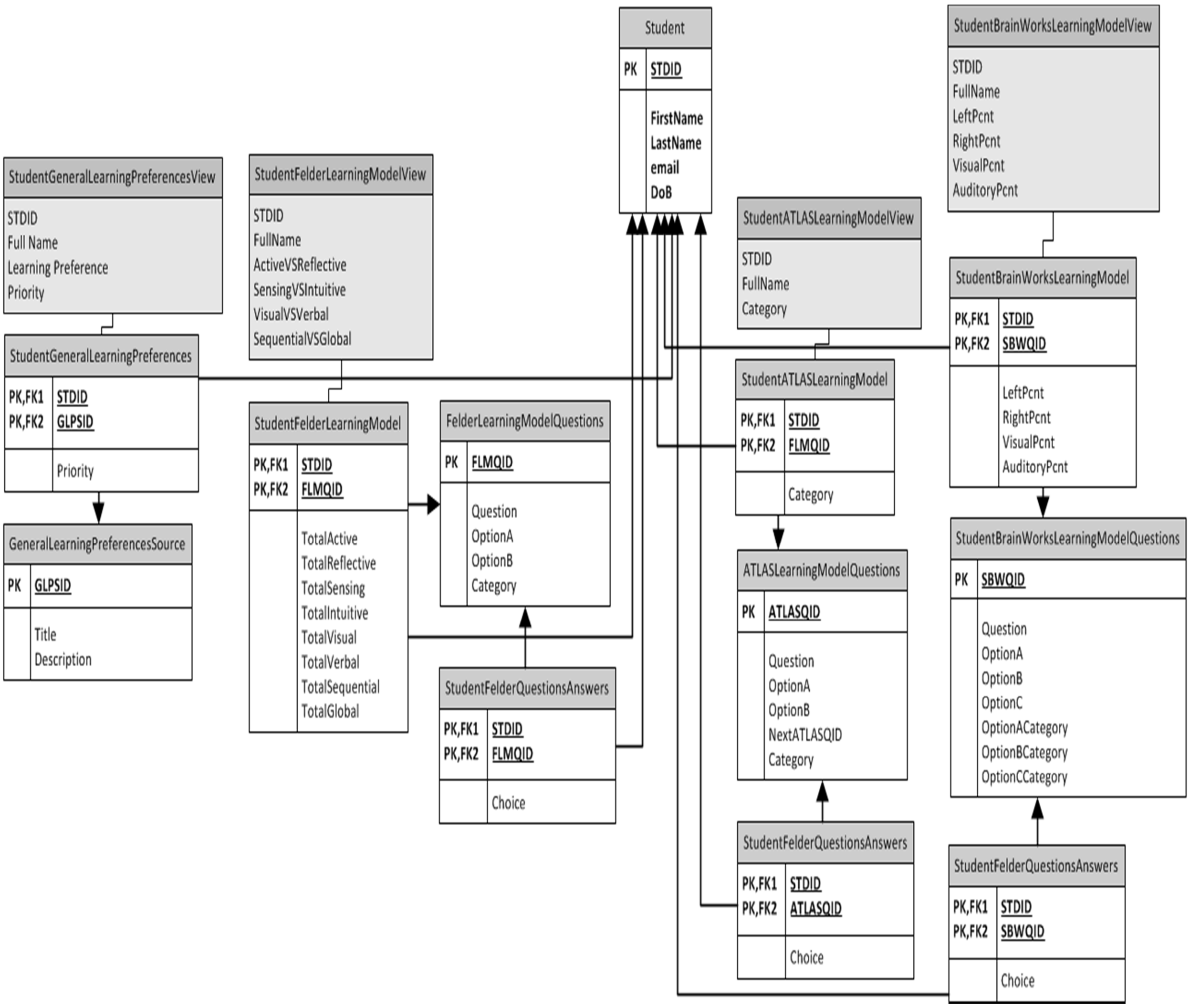


Figure 5.4: Students Learning Profile Manager Database Tables

Four different learning profiles are available for each student: General, Felder, ATLAS, and Brain Works. Figure 5.5 presents the General Profile Manager screen. Figure 5.6 presents Felder Learning Profile Manager and displays sample of Felder Questions. Felder questions sample include:

1. I understand something better after I:

   try it out. think it through.

2. I would rather be considered

   realistic. innovative.

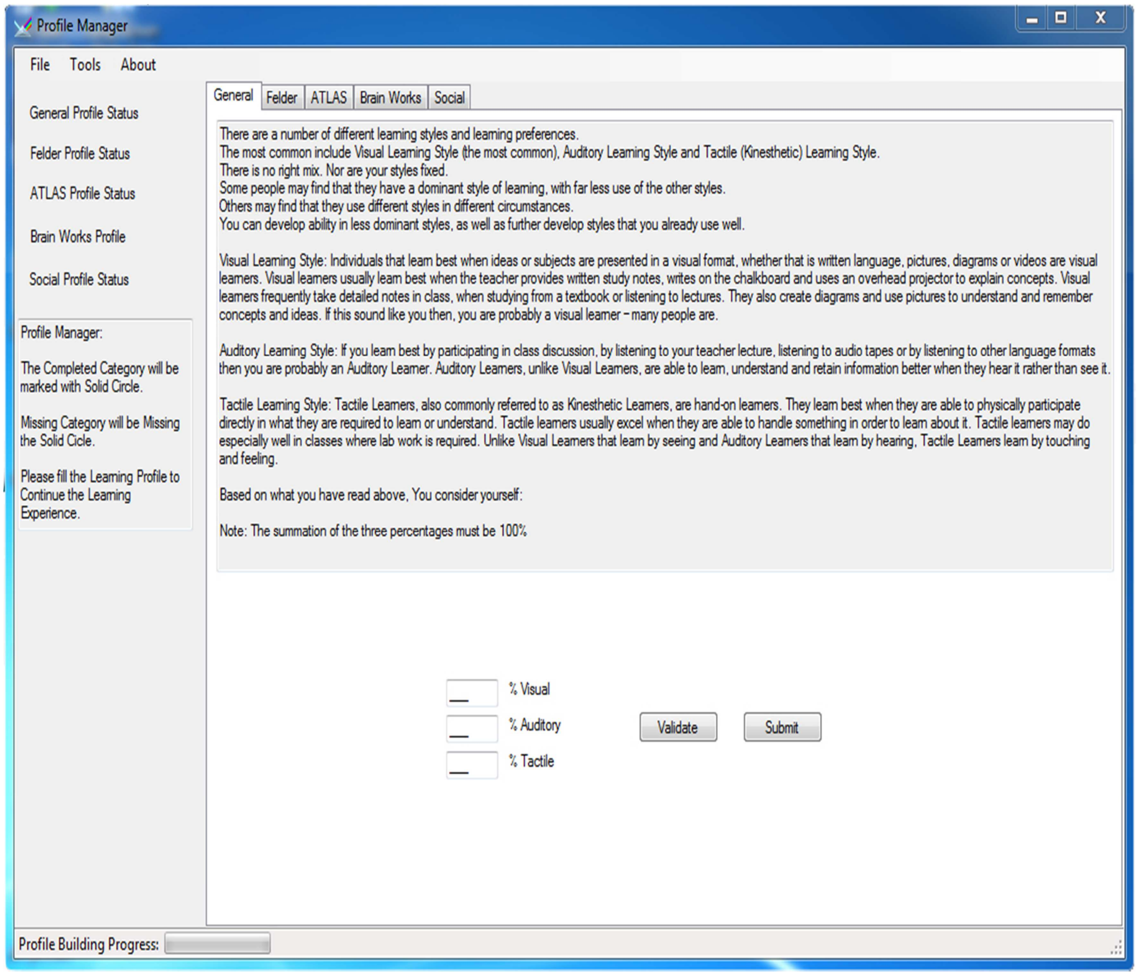


Figure 5.5: Student Learning Profile Manager Screen - General Learning Profile

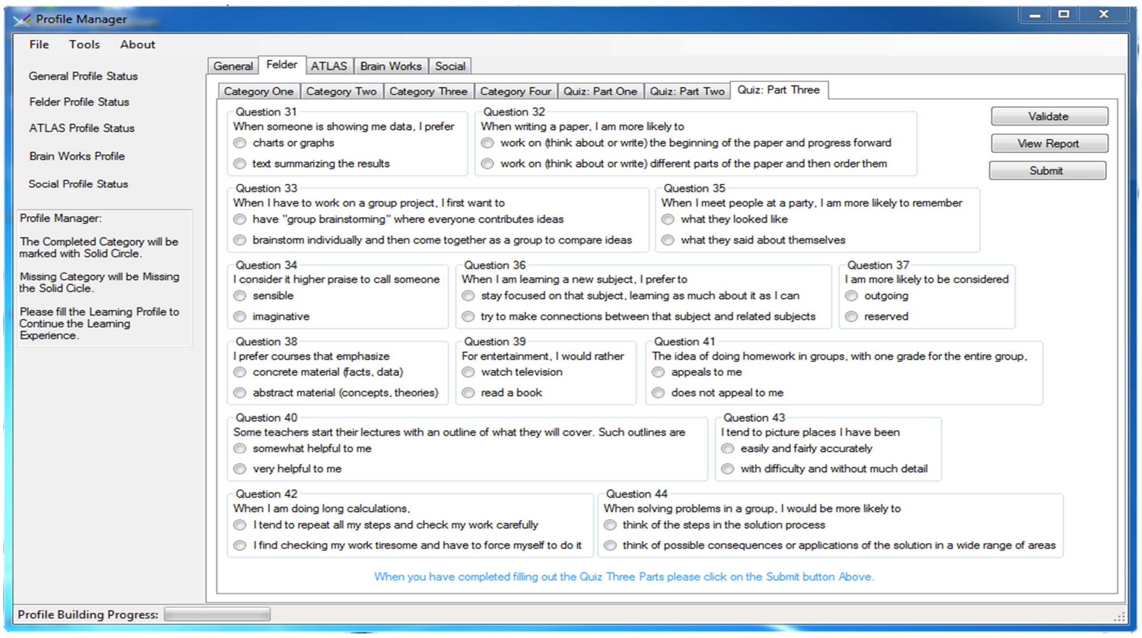


Figure 5.6: Student Learning Profile Manager Screen - Felder Learning Profile

Figure 5.7 presents Felder Learning Style Report. Felder proposes calculation method to identify to which category each student belongs. Figure 5.7 presents an instance of the Felder report.

Figure 5.8 presents ATLAS Learning Style profile manager. ATLAS learning style doesn't rely on questions heavily as Felder does. However, it is a step-by-step series of questions that are capable of identifying students' learning style at the end. Figure 5.9 displays BrainWorks questions sample, and figure 5.10 displays the BrainWorks report. BrainWorks learning style identifies two aspects for each student: auditory or visual, and left or right brain hemisphere directed.

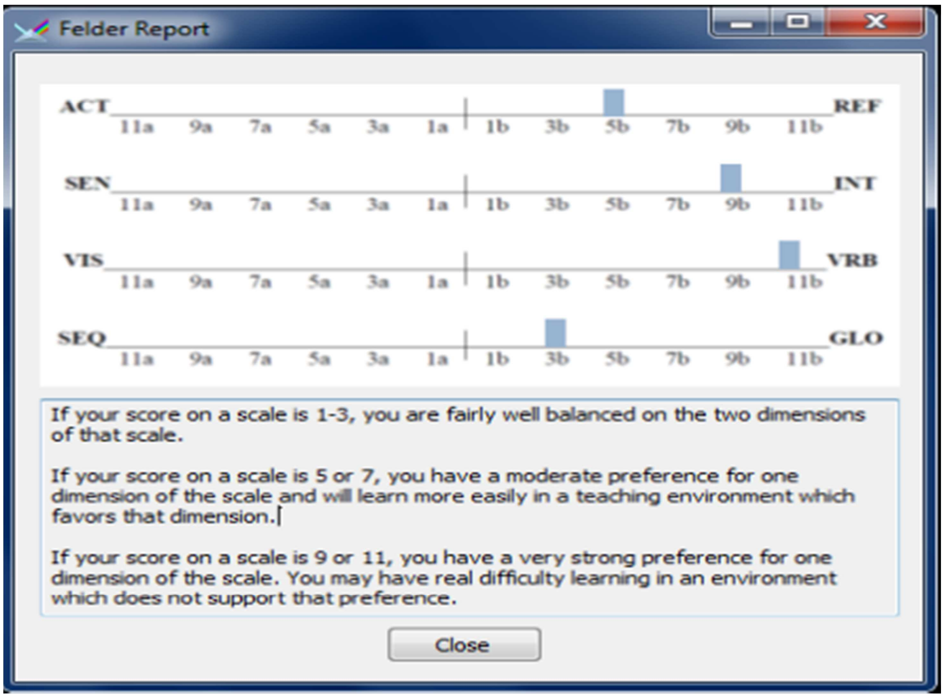


Figure 5.7: Student Learning Profile Manager - Felder Report

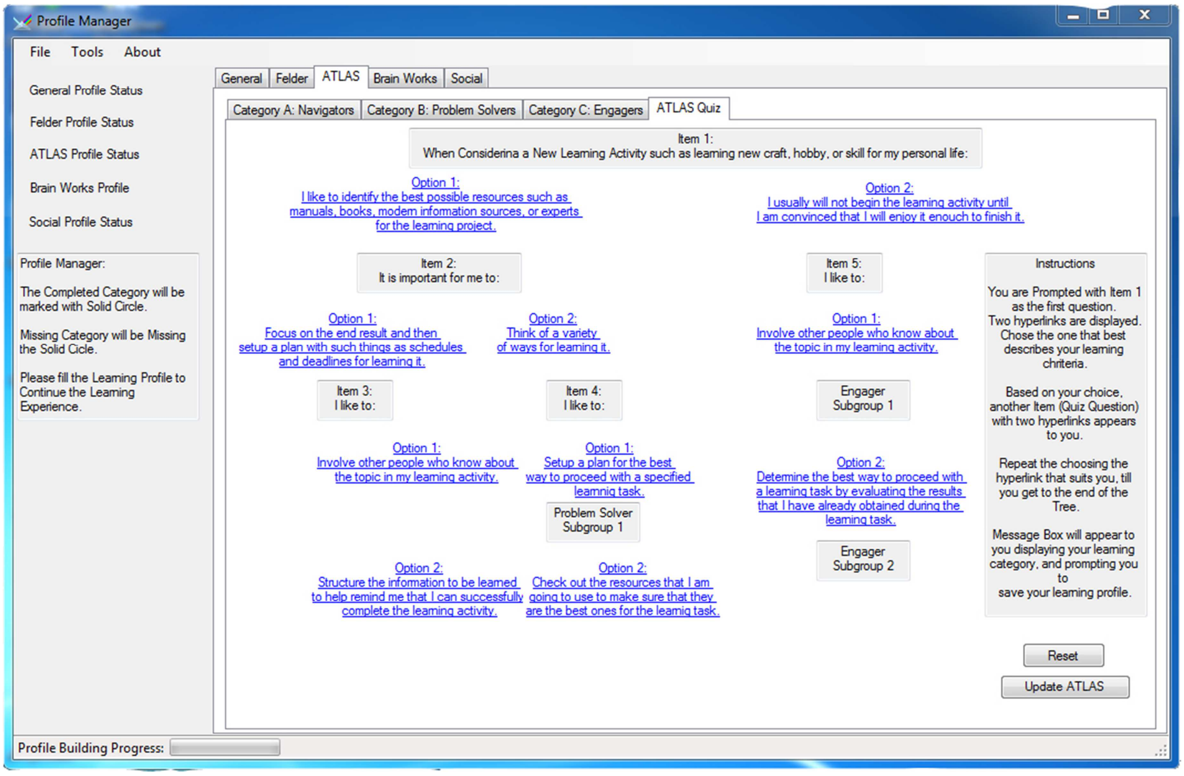


Figure 5.8: Student Learning Profile Manager Screen - ATLAS Learning Profile

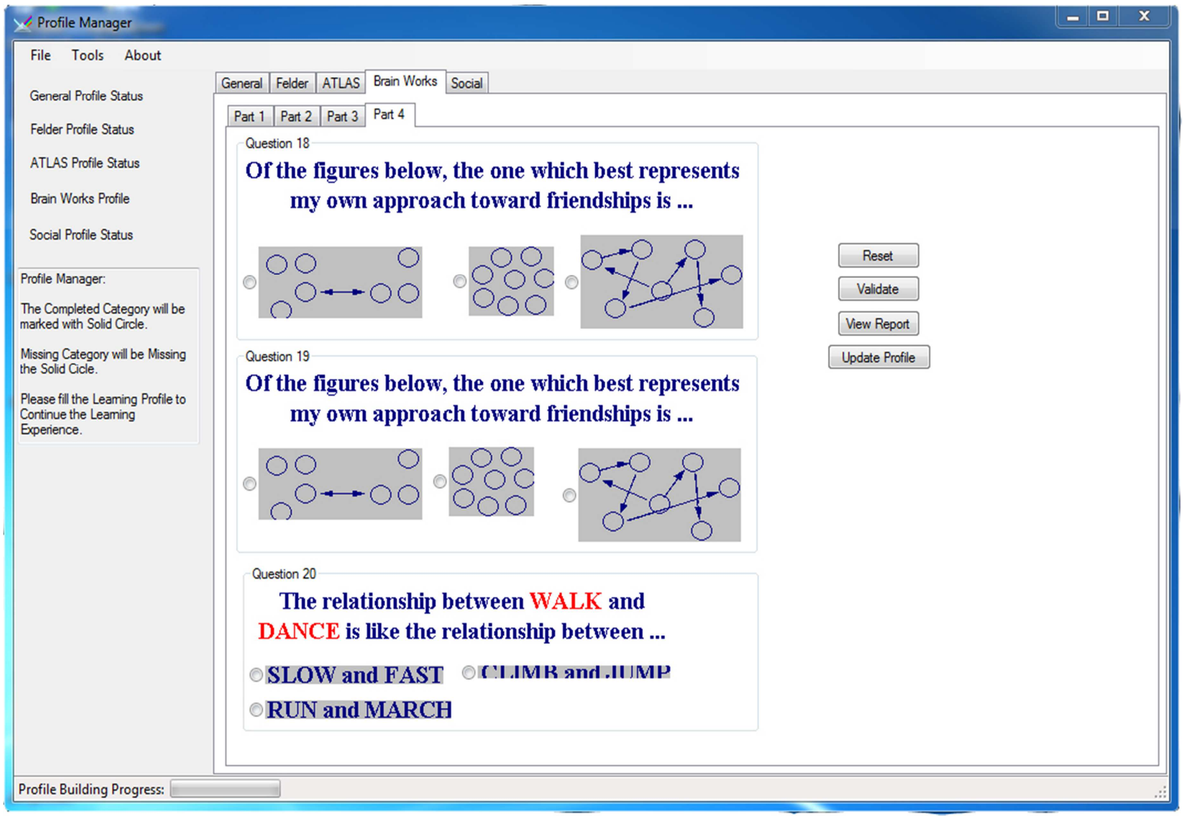


Figure 5.9: Student Learning Profile Manager Screen – Brain Works Learning Profile

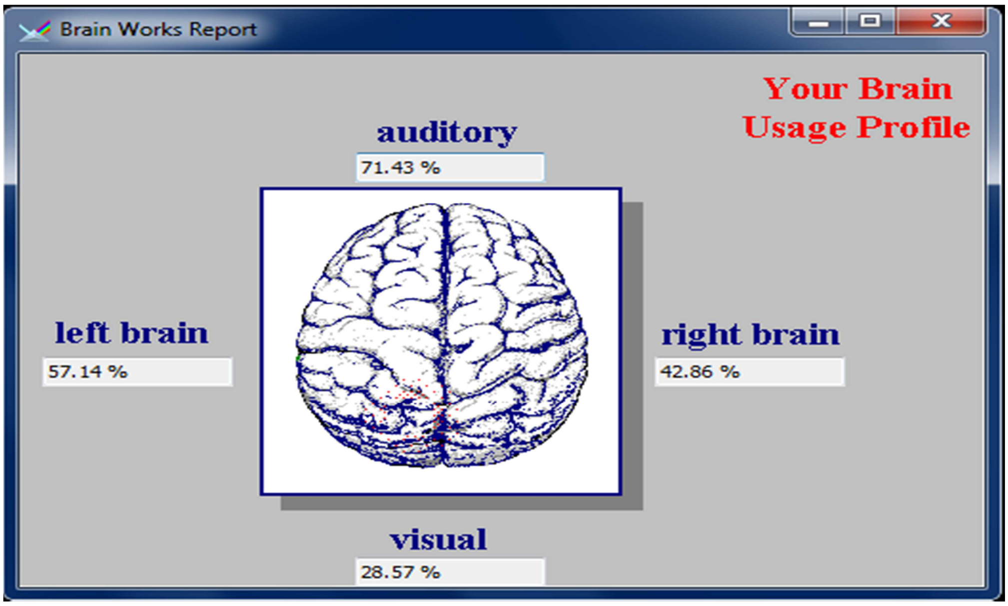


Figure 5.10: Student Learning Profile Manager - Brainworks Report

5.4 Students Usage Data Manager

Student Usage Data Manager keeps track of three student's behavior that are used to adjust the overall adaptive model performance and behavior. Figure 5.11 presents database tables to support this service. Usage data include:

1. **Browsing Behavior:** Intelligent LOs Recommender depends on students browsing through different LOs for initiating relation between different LOs. Tracking Referrer and Target URL helps the Intelligent LOs Recommender based on students browsing relates LOs together.
2. **LOs Study Time:** Among specifications that instructors associate with LOs, they identify the time needed to study it. However, that time might differ from student to another based on their personal differences. Recording the time taken by student to study certain LO is so important for the system. It helps both student and the system.
3. **LOs Ranking:** Students can express their thoughts about each LO in any of the three ways: Like/Not Like, Starring the LO from 1 to 5, or leaving comment/feedback about it.

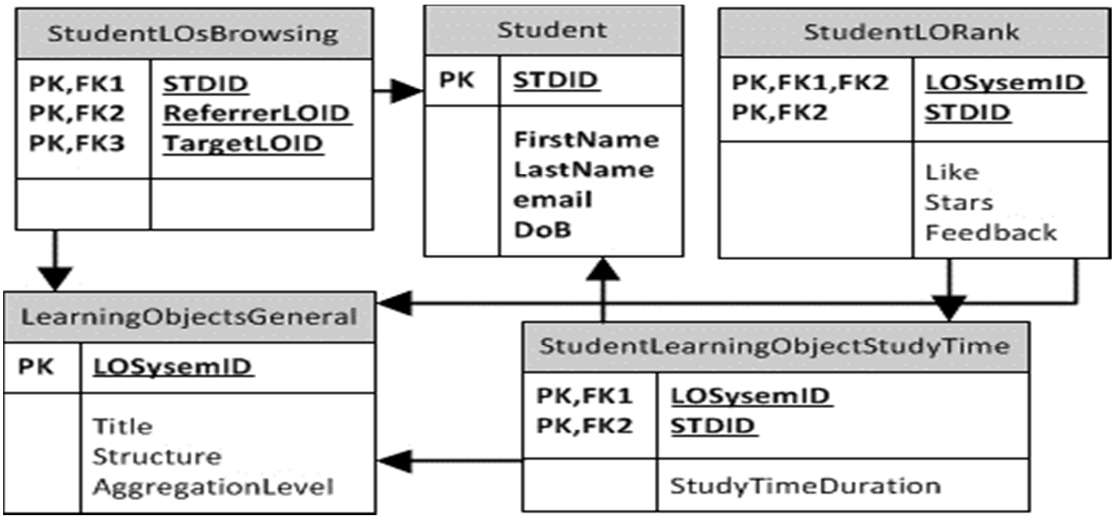


Figure 5.11: Students Usage Manager Database Tables

5.5 Learning Objects Manager Service

LOs metadata needed to support the adaptive e-Learning models is presented in details in chapter three. Figure 5.12 presents an instructor's screen from the instructor portal that is used to edit the metadata of LOs. Figure 5.13 highlights the search functionality available for instructors. Figure 5.14 in two parts presents the needed database tables to support LOs metadata.

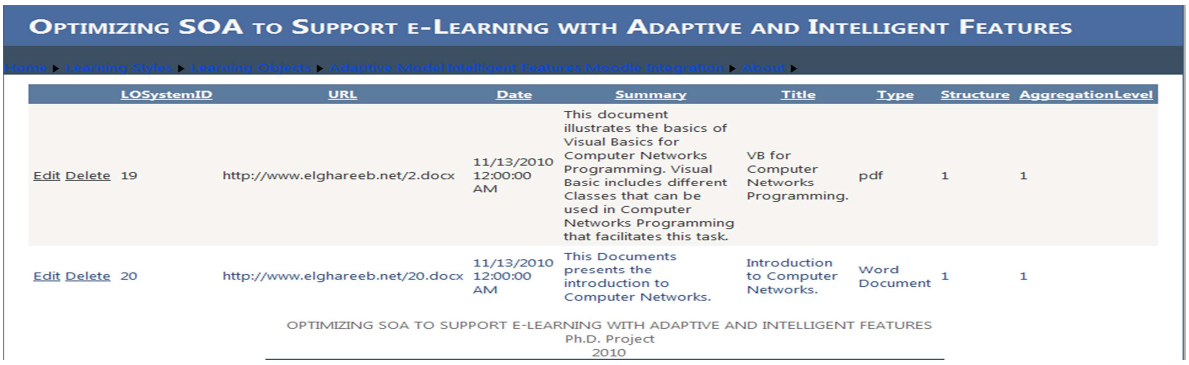


Figure 5.12: LOs Manager from Instructor's Portal

Figure 5.13: LOs Search from Instructor's Portal

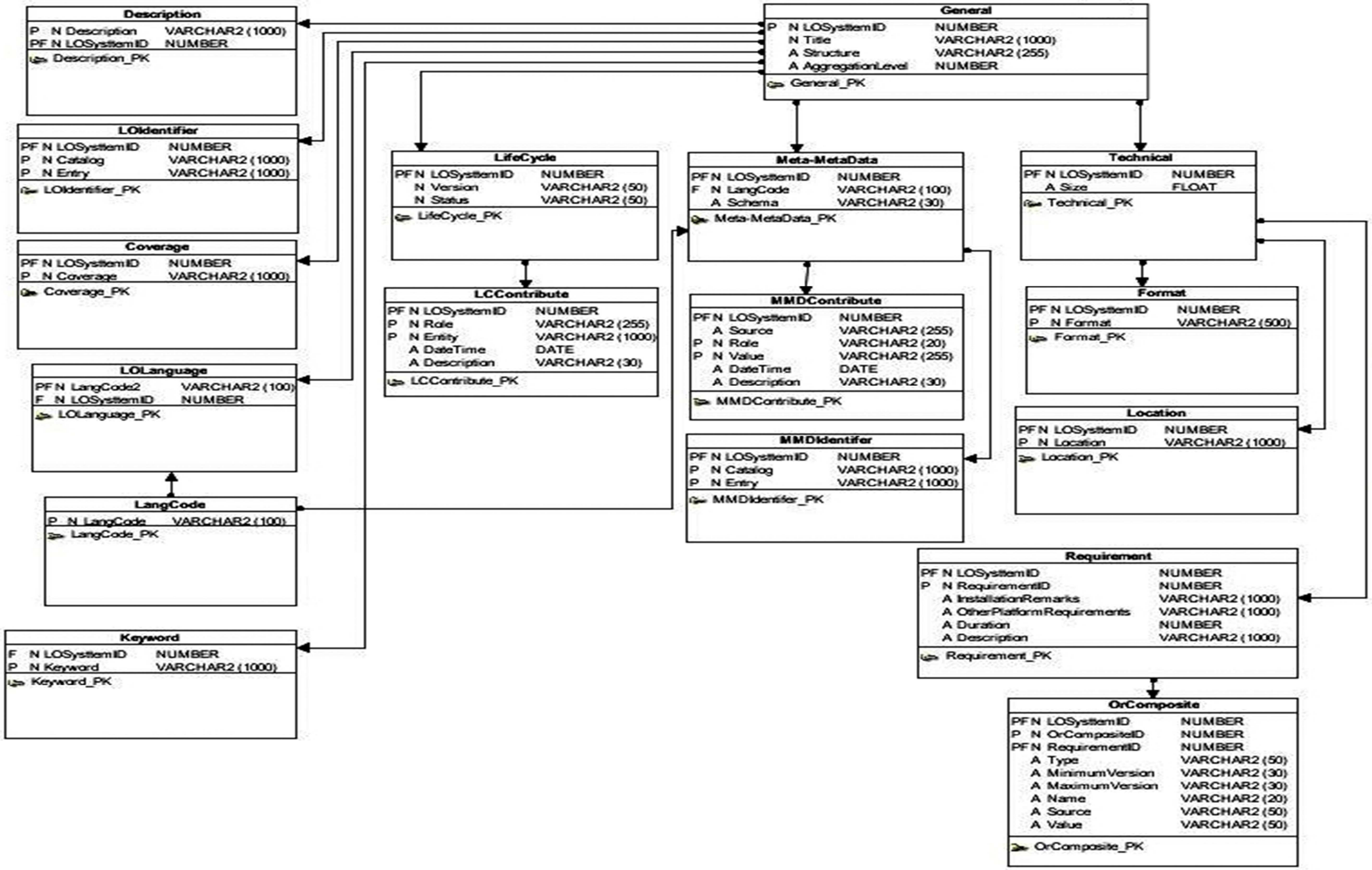


Figure 5.14: LOs Metadata Tables – Part I

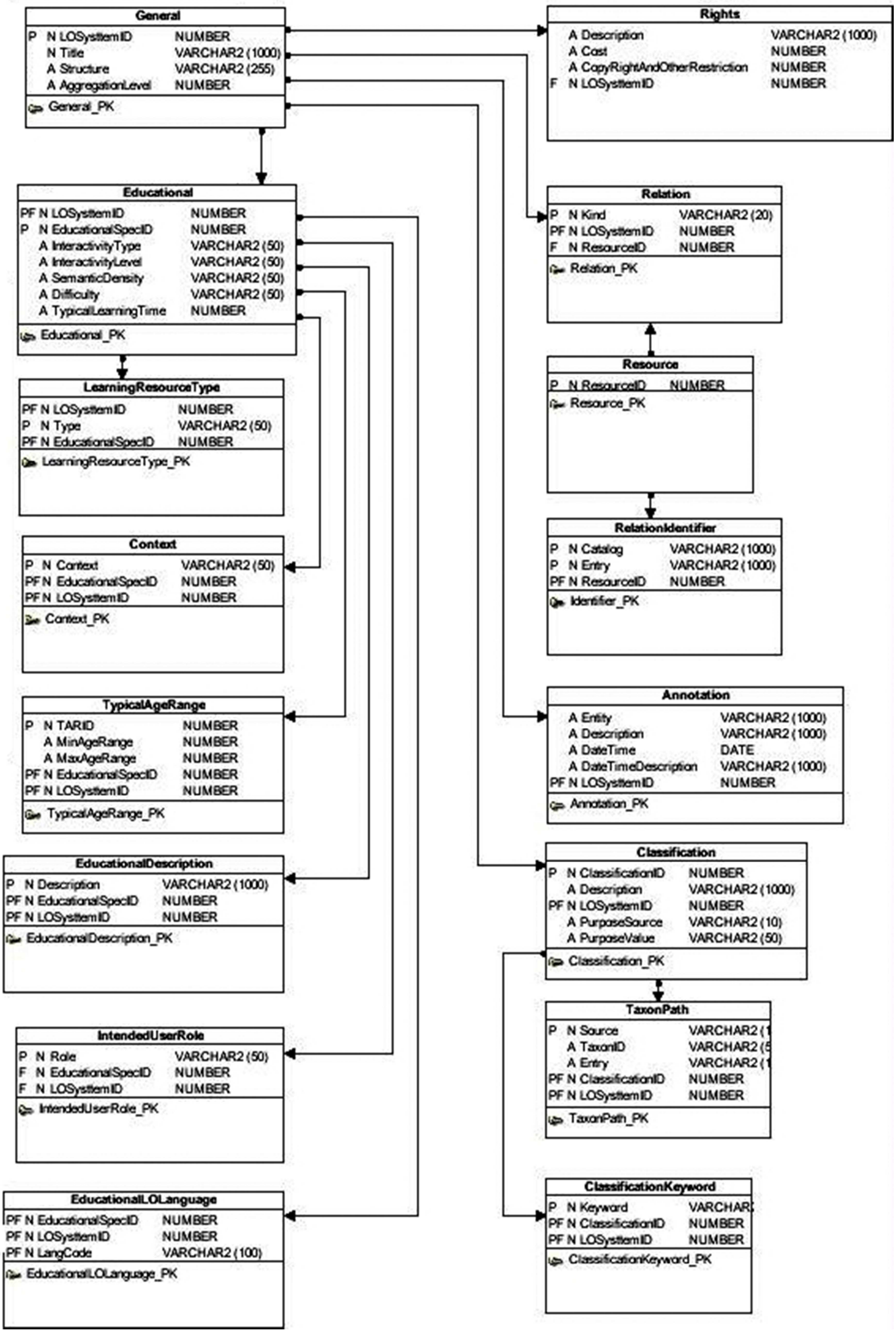


Figure 5.14: LOs Metadata Tables - Part II

5.6 Intelligent LOs Recommender

With the availability of unique 221 different learning objects varying from word documents (.txt, .doc, .docx, .pdf) and presentations (.ppt, .pptx), a service that reads the contents of digital libraries in the previous formats and extracts the contents to be further processed. Processing those 221 files yielded 1623 high quality keywords when compared to those generated from the Web pages. Intelligent LOs Recommender include different modules and services, they are: Pending LOs for Recommendation Manager Module, Crawler Module, and Document Processor Service.

5.6.1 Pending LOs for Recommendation Manager Module

Pending LOs represent LOs that crawler collected information about through querying online search engines, but haven't been processed yet. They are marked as pending waiting for processing either to be stored or deleted. Figure 5.15 presents the Pending LOs for Recommendation Manager database tables that temporarily store collected information about the two LOs types: files, and online resources.

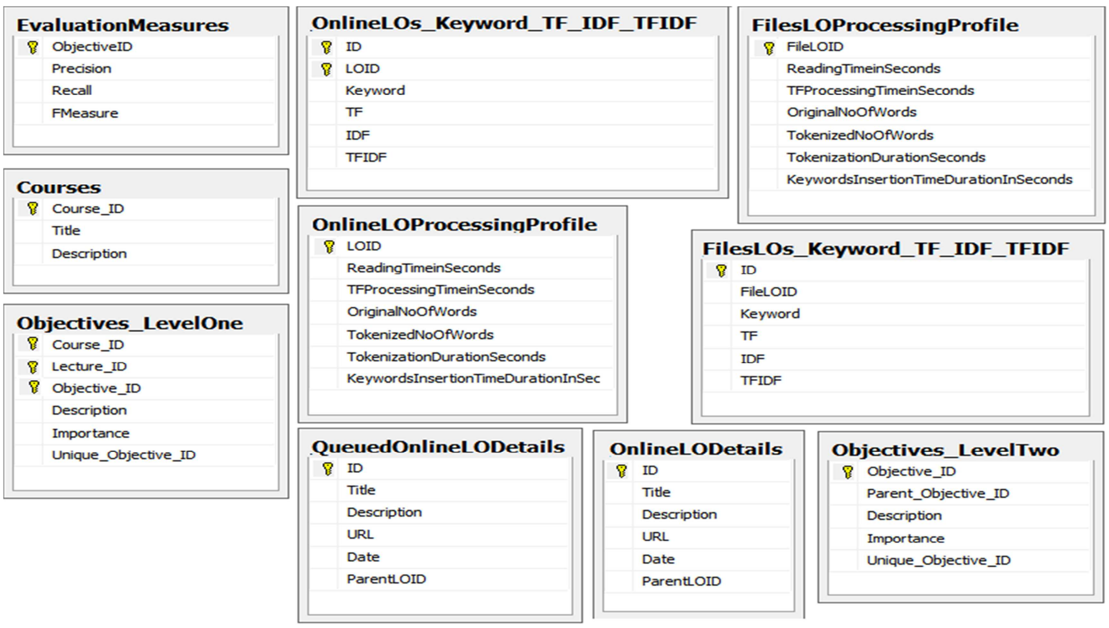


Figure 5.15: Learning Objects Recommendation Process Database Tables

5.6.2 Crawler Module

Crawler is a program that searches for information on the Web, and they are widely used by Web search engines to index all the pages on a site by following the links from page to page. Based on the crawler model, we developed a Web based java crawler that reads Google Search results via an open source library; that is Selenium. Working on the knowledge domain of information systems analysis and design, we ran the search for queries including different combinations of the keywords: System + Analysis + Design + Tutorial. Crawler extracts three main pieces of information for each search result: Title, URL, and Description. Database includes the summary of 2236 records, yielding 161 unique stemmed keywords. Figure 5.16 presents Crawler module package diagram. Figure 5.17 presents Crawler module source code needed to execute. Challenges faced this process include:

- List of search results needed further cleansing before processing because there were list of web sites that yields fake results.
- Search results can't be relied on due to intensive commercial contents.
- Stemmed keywords are sometimes meaningless. Automatically generated keywords will need revision and enhancement.

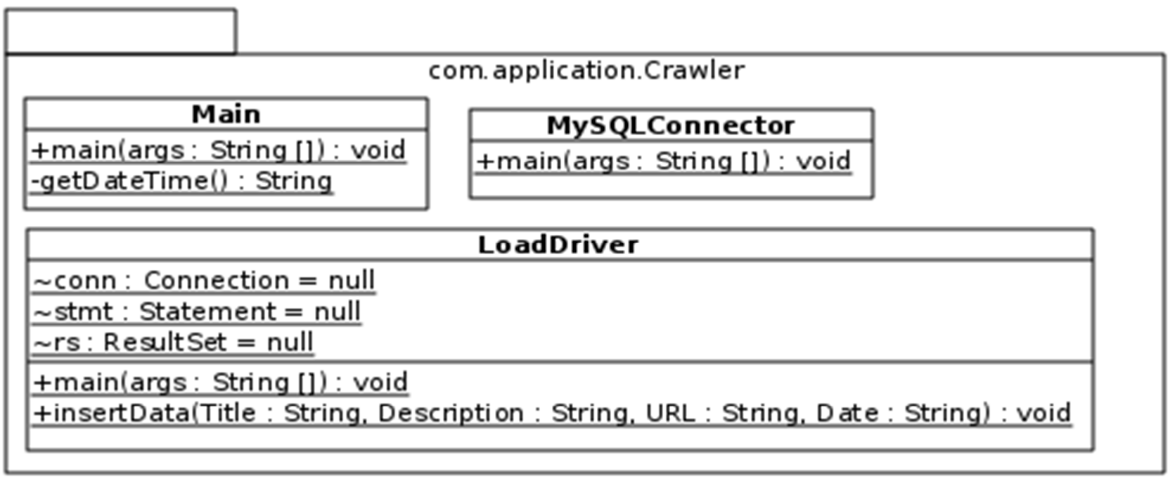


Figure 5.16: Crawler Package Diagram

```java
/*...*/
package crawler;
import java.util.List;
import org.openqa.selenium.By;
import org.openqa.selenium.RenderedWebElement;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
/**
 *
 * @author Haitham A. El-Ghareeb
 */
public class Main {
    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        WebDriver driver = new ChromeDriver();
        String searchKeyWord = "selenium";
        // number of visited page * 10    Ex: 5 page * 10 = 50
        int visitedPage = 50;
        for (int j = 0; j < visitedPage; j=j+10) {
            String url = "http://www.google.com/search?q="+searchKeyWord+"&start="+j+"&sa=N";
            driver.get(url);
            WebElement element = driver.findElement(By.name("q"));
            element.submit();
            long end = System.currentTimeMillis() + 20000;
            while (System.currentTimeMillis() < end) {
                // Browsers which render content (such as Firefox and IE) return "RenderedWebElements"
                RenderedWebElement resultsDiv = (RenderedWebElement) driver.findElement(By.className("r"));
                // If results have been returned, the results are displayed in a drop down.
                if (resultsDiv.isDisplayed()) {
                    break;
                }
            }
            List  allDescription = driver.findElements(By.xpath("//li[@class='g']"));
            List  allTitle = driver.findElements(By.xpath("//h3[@class='r']"));
            for (int i = 0; i < allDescription.size(); i++) {
               try{
                   WebElement title = (WebElement) allTitle.get(i);
                   WebElement description_link = (WebElement) allDescription.get(i);
                   System.out.println("Title :" + title.getText());
                   String des_link = description_link.getText();
                   String des_link_array[] = des_link.split("\n");
                   System.out.println("Description :" + des_link_array[0]);
                   String link = des_link_array[des_link_array.length - 1];
                   // make url from www.google.com/ - Cached - Similar --> www.google.com/
                   link = link.substring(0, link.length() - 18);
                   System.out.println("Link :" + link);
               }catch(Exception e){}
            }
        }
    }
}
```

Figure 5.17: Crawler Source Code

5.6.3 Document Processor Service

Document Processor Service includes phases presented at figure 5.18. Phases are:

- **Tokenization:** A document is treated as a string, and then partitioned into a list of tokens.
- **Removing stop words:** Stop words such as “the”, “a”, “and” are frequently occurring, so the insignificant words need to be removed.
- **Stemming:** Applying stemming algorithm that converts different word forms into similar canonical form. This step is the process of conflating tokens to their root form, e.g. connection to connect, computing to compute.

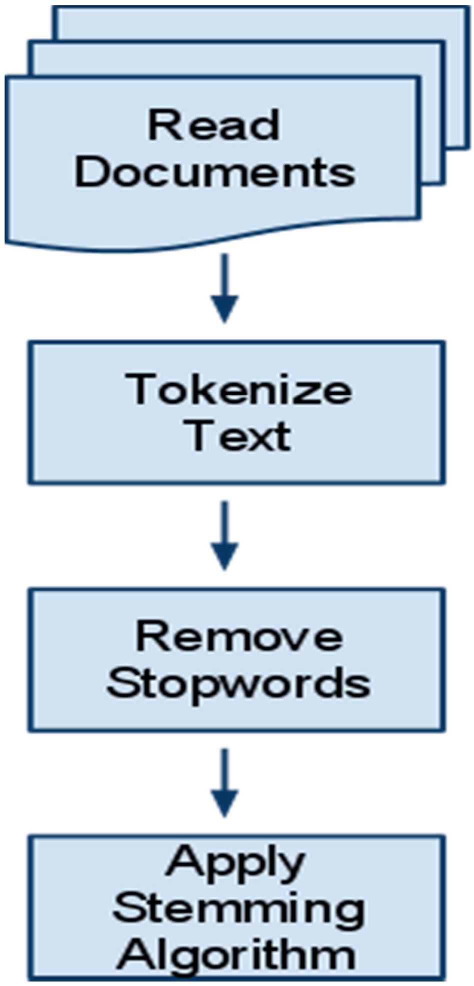


Figure 5.18: Document Processor Phases

5.6.3.1 Tokenizer Module

Document Processor Service tokenizes sentences into words based on any of the following symbols: dot, comma, space, semicolon, and colon. It takes the String to tokenize as an input and returns list of tokenized words. Those words will be the input to the next module: Removing Stop words.

5.6.3.2 Removing Stop words Module

Stop words are words which are filtered out after Tokenizer module processing of text. There is not one definite list of stop words which all tools use. Observations lead to a customized list of stop words that includes the following list as a sample. Course Objectives usually include words like the ones included in the following list.

```
stopwords = {"details", "course", "lecture", "explains", "discusses", "outside", "inside", "curriculum", "syllabus",
"explain", "discuss"}
```

5.6.3.3 Stemmer Module

Figure 5.19 illustrates the Stemmer Module pseudo code and presents the conducted activities. Stemmer module activities include:

- Split the input using delimiter
- For every processed word:
    - Declare Global Variables.
    - Split the word into two stems.
    - Search for Patterns, and replace them.
    - Return Stemmed word.

Step 1: Split input using Delimiter.

Step 2: For Every word in the Splitted Query, Do:

Step 3: Declare Global Variables

```
Vowels = {"a", "e", "i", "o", "u", "y"}
```

Step 4: Generate Stem (Word) as Stem

Step 4.1: Split the Word into two Stems at any of the characters of in the Vowels Array. Word is splitted to R1 and R2. R1 is the region after the first non-vowel following a vowel, or is the null region at the end of the word if there is no such non-vowel. R2 is the other region.

Step 4.2: Search for the longest among the following suffixes, and, if found and in R1, perform the action indicated.

- tional: replace by tion
- li+: delete if preceded by a valid li-ending

```
Friend Sub ReplaceEndingStep2()
        If EndsWithAndInR1(Indicated Value) Then
            Stem = Stem.Substring(0, Stem.Length –Length) & Replacement Value
            Return
        End If
```

Step 4.3: Search for the longest among the following suffixes, and, if found and in R1, perform the action indicated.

- tional+: replace by tion
- ative*: delete if in R2

```
Friend Sub ReplaceEndingStep3()
        'ational+:   replace by ate
        If EndsWithAndInR1(Indicated Value) Then
            Stem = Stem.Substring(0, Stem.Length – Length) & Replacement Value
            Return
        End If
End Sub
```

Step 4.4: Search for the longest among the following suffixes, and, if found and in R2, perform the action indicated. "al ance ence er ic able ible ant ement ment ent ism ate iti ous ive ize delete " ion delete if preceded by s or t

```
Friend Sub StripSuffixStep4()
        If EndsWithAndInR2(Any Previous Indicated Value) Then
            Stem = Stem.Remove(Stem.Length – Length of Indicated Value)
            Return
        End If
    End Sub
```

Step 4.5: Search for the following suffixes, and, if found, perform the action indicated. 'e' delete if in R2, or in R1 and not preceded by a short syllable, l delete if in R2 and preceded by 'l'

```
Friend Sub StripSuffixStep5()
        If EndsWithAndInR2("e") OrElse (EndsWithAndInR1("e") AndAlso
IsShortSyllable(Stem.Length - 3) = False) Then
            Stem = Stem.Remove(Stem.Length - 1)
            Return
        End If
        If EndsWithAndInR2("l") AndAlso Stem.EndsWith("ll") Then
            Stem = Stem.Remove(Stem.Length - 1)
            Return
        End If
    End Sub
```

Step 5: Return Joined Stemmed Words

Figure 5.19: Stemmer Module Pseudo Code

5.6.3.4 Query Expansion Module

WordNet is a large lexical database of English used to find synonyms of stemmed keywords generated by Stemmer Module to be further used in the search and recommendation process in later services / modules. Figure 5.20 presents WordNet relational database tables implementation to provide Query Expansion capabilities to the Adaptive e-Learning Model. Each set of synonyms (synset), has a unique index and shares its properties, such as a dictionary definition (lemma).

Query Expansion takes place as follows:

1. Take input (the term to expand).
2. Search WordNet synsets for the term. When Found: Return Synonyms to Expand Query.

Figure 5.21 presents Intelligent LOs Recommender Class Diagram.

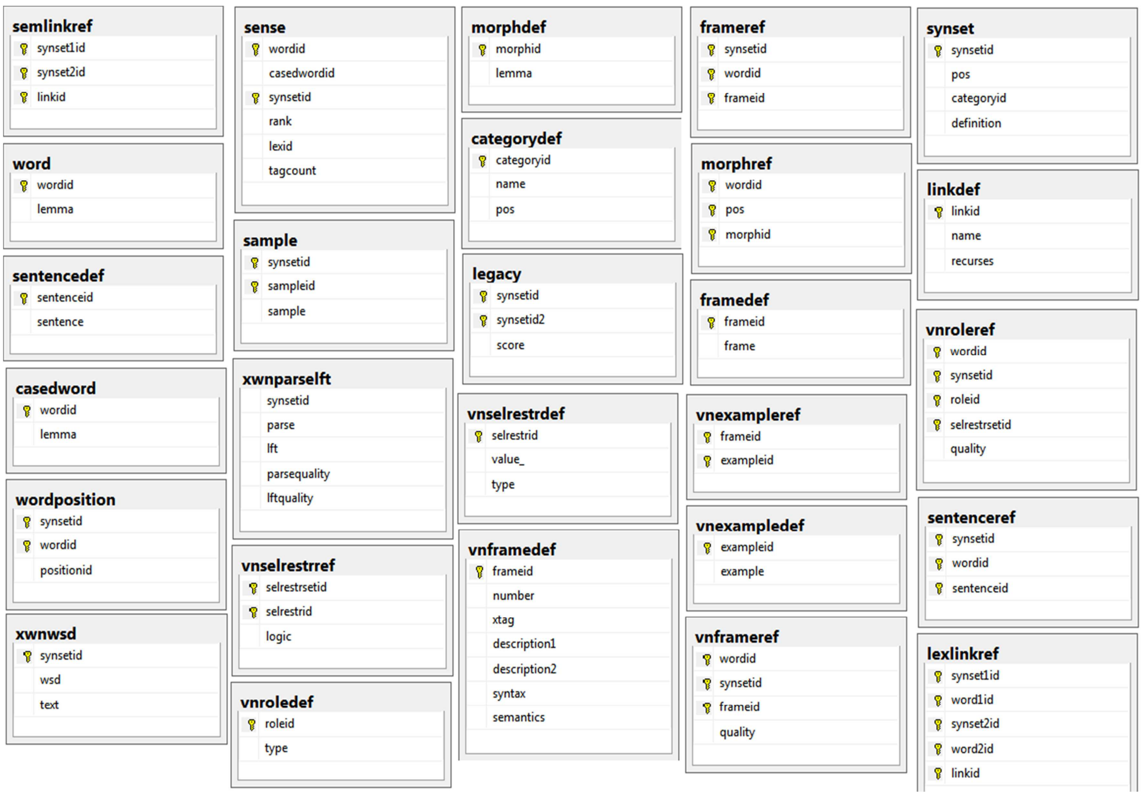


Figure 5.20: WordNet Relational Database Tables

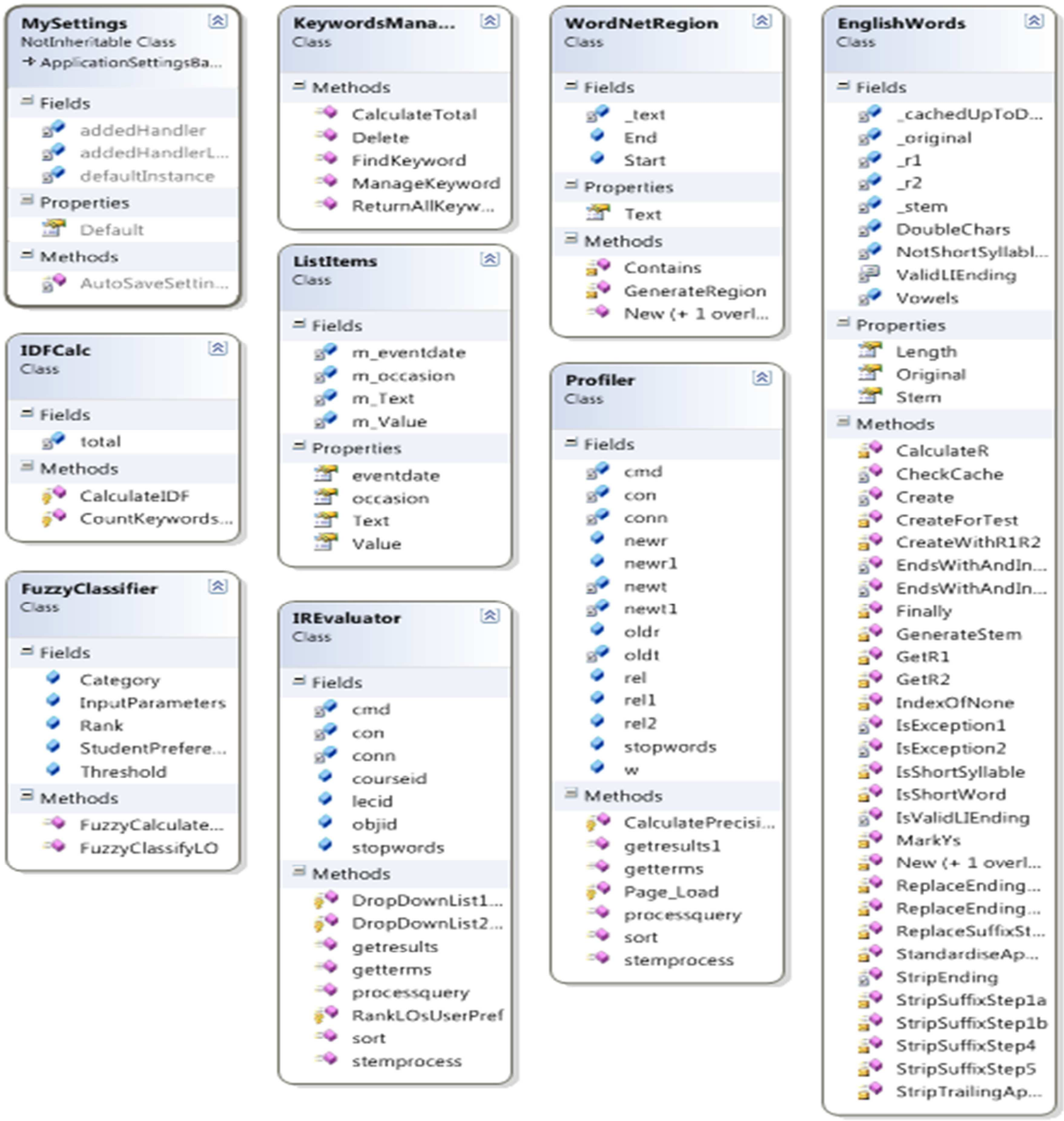


Figure 5.21: Intelligent LOs Recommender Class Diagram

Figure 5.22 presents a screen shot from the instructor's portal that manages extracted keywords from LOs and calculates each document's Term Frequency (TF). Figure 5.23 presents a LOs recommended list from instructor's portal. Instructor can edit and delete recommended LOs list. This functionality is used to fine tune the automatic recommendation process. Figure 5.24 presents LOs recommender in action from student's portal. Figure 5.25 presents Evaluation results of LOs recommender. Evaluation is discussed in details in the next chapter.

Please Choose an LO to Display Keywords TF:

Proposed Adaptive e-Learning Model.pdf

	ID	FileID	Keyword	TF
Delete	1	1	propos	0.0128786072024803
Delete	2	1	adapt	0.0221798235153828
Delete	3	1	e	0.0157405199141426
Delete	4	1	learn	0.0591461960410207
Delete	5	1	model	0.0140710708323396
Delete	6	1	inform	0.0062008108752683
Delete	7	1	system	0.01669449081803
Delete	8	1	technolog	0.00739327450512759
Delete	9	1	process	0.0062008108752683
Delete	10	1	featur	0.00548533269735273

1 2 3

Figure 5.22: Extracted Keywords and TF from Instructor's Portal

	ID	Name	FullName	Extension	Length	CreationTime
Edit Delete	1	Proposed Adaptive e-Learning Model.pdf	http://www.haitham-online.net/SundayMeeting/Site/aquatic/Proposed Adaptive e-Learning Model.pdf	.pdf	2630435	16/10/2010 02:48:12
Edit Delete	2	p105-bollacker.pdf	http://www.haitham-online.net/SundayMeeting/Site/aquatic/p105-bollacker.pdf	.pdf	123522	16/10/2010 02:48:21
Edit Delete	3	Important Student Modelling-case study.pdf	http://www.haitham-online.net/SundayMeeting/Site/aquatic/Important Student Modelling-case study.pdf	.pdf	292533	16/10/2010 02:48:29
Edit Delete	4	Student Model Components.pdf	http://www.haitham-online.net/SundayMeeting/Site/aquatic/Student Model Components.pdf	.pdf	113495	16/10/2010 02:48:30

Figure 5.23: LO's Recommendation List from Instructor's Portal

New Tab | localhost:49523/work/De...

localhost:49523/work/Default.aspx

Haitham A. El-Ghare... | Editorial Manager® | MIT OpenCourseWa...

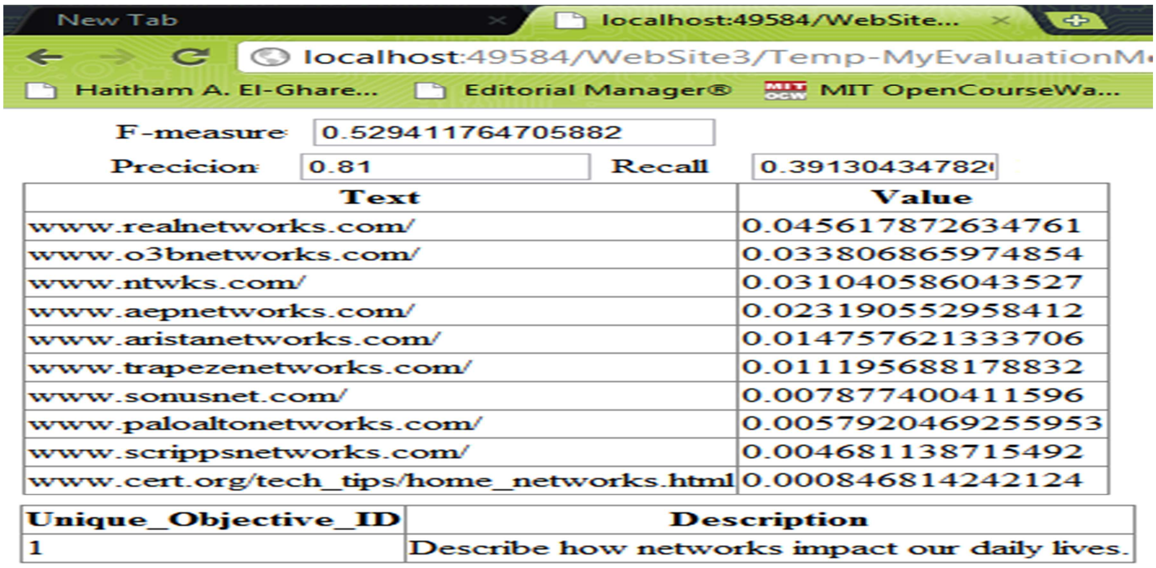
Course: Computer Networks Programming

Lecture ID: 1

Objective ID	LO URL	Rank
1	http://en.wikipedia.org/wiki/Computer_net	0.03744496
1	http://www.bramjnet.com/vb3/forumdispla	0.04021487
2	http://compnetworking.about.com/	0.04253591
3	http://www.networktutorials.info/	0.0398329

Figure 5.24: Intelligent LOs Recommender in Action

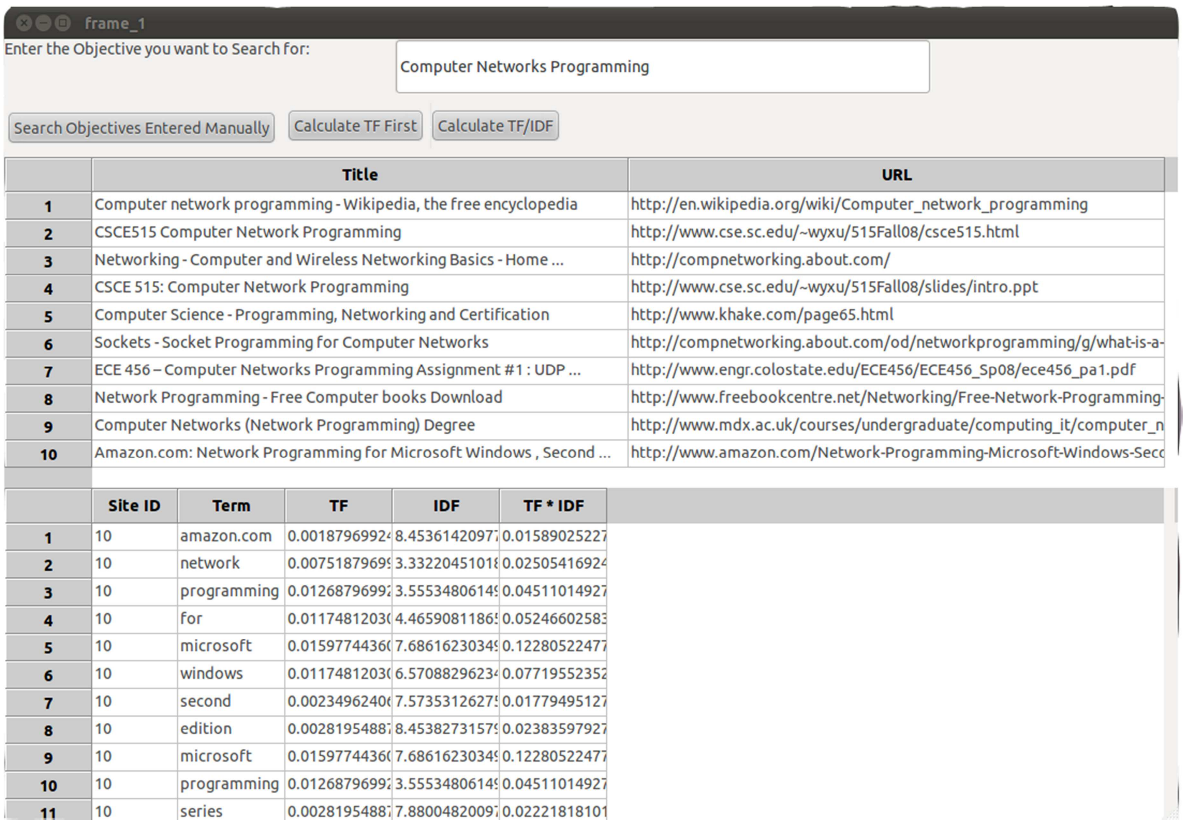
Figure 5.26 presents the Search for Keywords within LOs, LOS URL returned, TF and IDF Calculation results. Figure 5.27 presents the tokenization process, defining tokenized words, and query expansion process.



Text	Value
www.realnetworks.com/	0.045617872634761
www.o3bnetworks.com/	0.033806865974854
www.ntwks.com/	0.031040586043527
www.aepnetworks.com/	0.023190552958412
www.aristanetworks.com/	0.014757621333706
www.trapezenetworks.com/	0.011195688178832
www.sonusnet.com/	0.007877400411596
www.paloaltonetworks.com/	0.005792046925595 3
www.scrippsnetworks.com/	0.004681138715492
www.cert.org/tech_tips/home_networks.html	0.000846814242124

Unique_Objective_ID	Description
1	Describe how networks impact our daily lives.

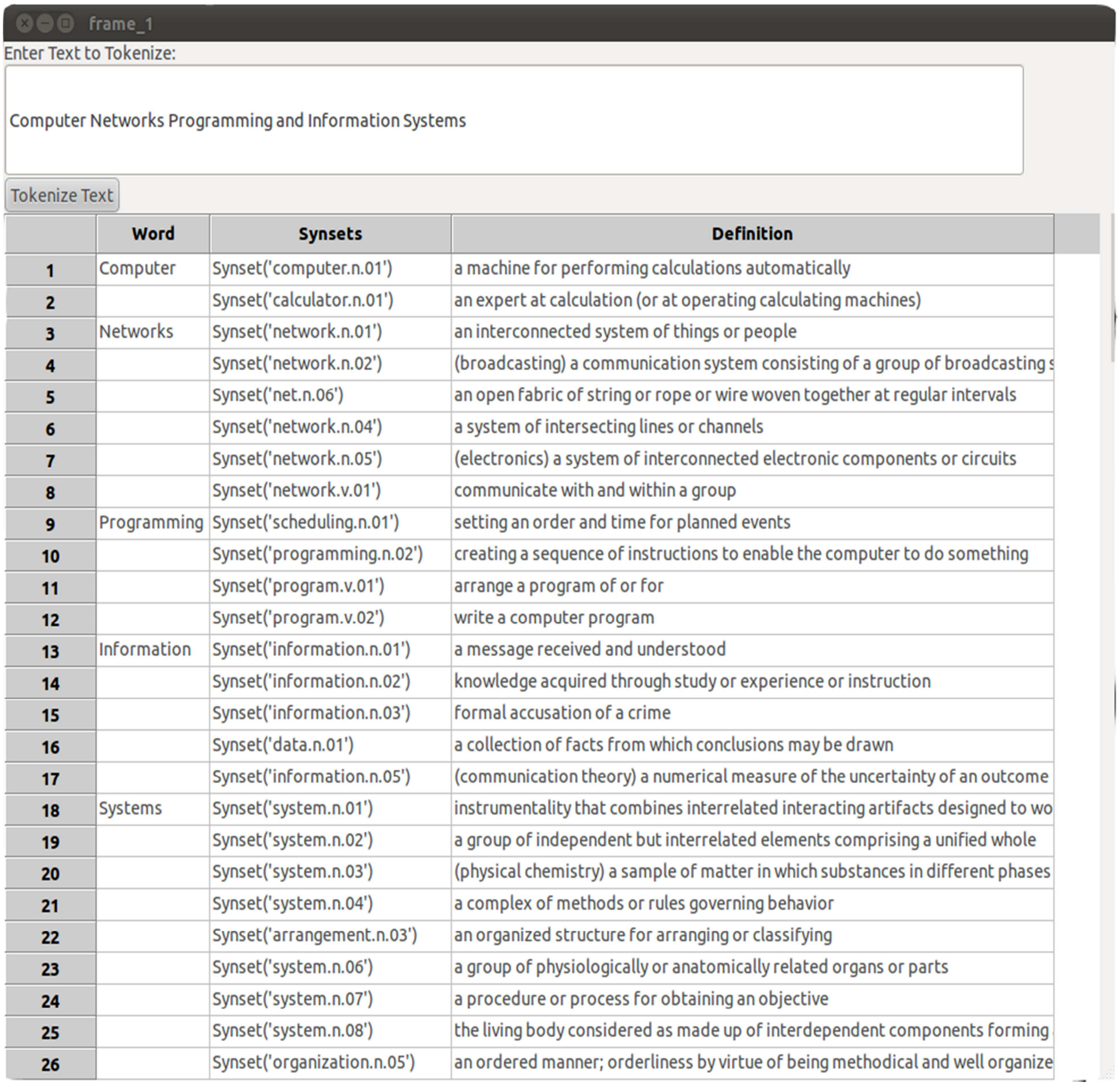
Figure 5.25: Information Retrieval Evaluation Results for Intelligent LOs Recommender



	Title	URL
1	Computer network programming - Wikipedia, the free encyclopedia	http://en.wikipedia.org/wiki/Computer_network_programming
2	CSCE515 Computer Network Programming	http://www.cse.sc.edu/~wyxu/515Fall08/csce515.html
3	Networking - Computer and Wireless Networking Basics - Home ...	http://compnetworking.about.com/
4	CSCE 515: Computer Network Programming	http://www.cse.sc.edu/~wyxu/515Fall08/slides/intro.ppt
5	Computer Science - Programming, Networking and Certification	http://www.khake.com/page65.html
6	Sockets - Socket Programming for Computer Networks	http://compnetworking.about.com/od/networkprogramming/g/what-is-a-
7	ECE 456 – Computer Networks Programming Assignment #1 : UDP ...	http://www.engr.colostate.edu/ECE456/ECE456_Sp08/ece456_pa1.pdf
8	Network Programming - Free Computer books Download	http://www.freebookcentre.net/Networking/Free-Network-Programming-
9	Computer Networks (Network Programming) Degree	http://www.mdx.ac.uk/courses/undergraduate/computing_it/computer_n
10	Amazon.com: Network Programming for Microsoft Windows , Second ...	http://www.amazon.com/Network-Programming-Microsoft-Windows-Secc

	Site ID	Term	TF	IDF	TF * IDF
1	10	amazon.com	0.0018796992	8.4536142097	0.01589025227
2	10	network	0.0075187969	3.3322045101	0.02505416924
3	10	programming	0.0126879699	3.5553480614	0.04511014927
4	10	for	0.0117481203	4.4659081186	0.05246602583
5	10	microsoft	0.0159774436	7.6861623034	0.12280522477
6	10	windows	0.0117481203	6.5708829623	0.07719552352
7	10	second	0.0023496240	7.5735312627	0.01779495127
8	10	edition	0.0028195488	8.4538273157	0.02383597927
9	10	microsoft	0.0159774436	7.6861623034	0.12280522477
10	10	programming	0.0126879699	3.5553480614	0.04511014927
11	10	series	0.0028195488	7.8800482009	0.02221818101

Figure 5.26: Keywords based LOs Search, TF-IDF Calculation for LOs Search Results



	Word	Synsets	Definition
1	Computer	Synset('computer.n.01')	a machine for performing calculations automatically
2		Synset('calculator.n.01')	an expert at calculation (or at operating calculating machines)
3	Networks	Synset('network.n.01')	an interconnected system of things or people
4		Synset('network.n.02')	(broadcasting) a communication system consisting of a group of broadcasting s
5		Synset('net.n.06')	an open fabric of string or rope or wire woven together at regular intervals
6		Synset('network.n.04')	a system of intersecting lines or channels
7		Synset('network.n.05')	(electronics) a system of interconnected electronic components or circuits
8		Synset('network.v.01')	communicate with and within a group
9	Programming	Synset('scheduling.n.01')	setting an order and time for planned events
10		Synset('programming.n.02')	creating a sequence of instructions to enable the computer to do something
11		Synset('program.v.01')	arrange a program of or for
12		Synset('program.v.02')	write a computer program
13	Information	Synset('information.n.01')	a message received and understood
14		Synset('information.n.02')	knowledge acquired through study or experience or instruction
15		Synset('information.n.03')	formal accusation of a crime
16		Synset('data.n.01')	a collection of facts from which conclusions may be drawn
17		Synset('information.n.05')	(communication theory) a numerical measure of the uncertainty of an outcome
18	Systems	Synset('system.n.01')	instrumentality that combines interrelated interacting artifacts designed to wo
19		Synset('system.n.02')	a group of independent but interrelated elements comprising a unified whole
20		Synset('system.n.03')	(physical chemistry) a sample of matter in which substances in different phases
21		Synset('system.n.04')	a complex of methods or rules governing behavior
22		Synset('arrangement.n.03')	an organized structure for arranging or classifying
23		Synset('system.n.06')	a group of physiologically or anatomically related organs or parts
24		Synset('system.n.07')	a procedure or process for obtaining an objective
25		Synset('system.n.08')	the living body considered as made up of interdependent components forming
26		Synset('organization.n.05')	an ordered manner; orderliness by virtue of being methodical and well organize

Figure 5.27: Tokenization and Query Expansion Process in action

5.7 Intelligent Meeting Manager for Suspended Students Service

Intelligent Meeting Manager for Suspended Students is responsible for defining meetings to students that didn't successfully pass the same exam for three times. This is an indication that student needs meeting. The intelligent need is addressed in finding the most appropriate time for both instructor and student from different available time slots. Figure 5.28 presents the required database table to support this service and Figure 5.29 presents the class diagram. Figure 5.30 presents Data Access Layer class diagram.

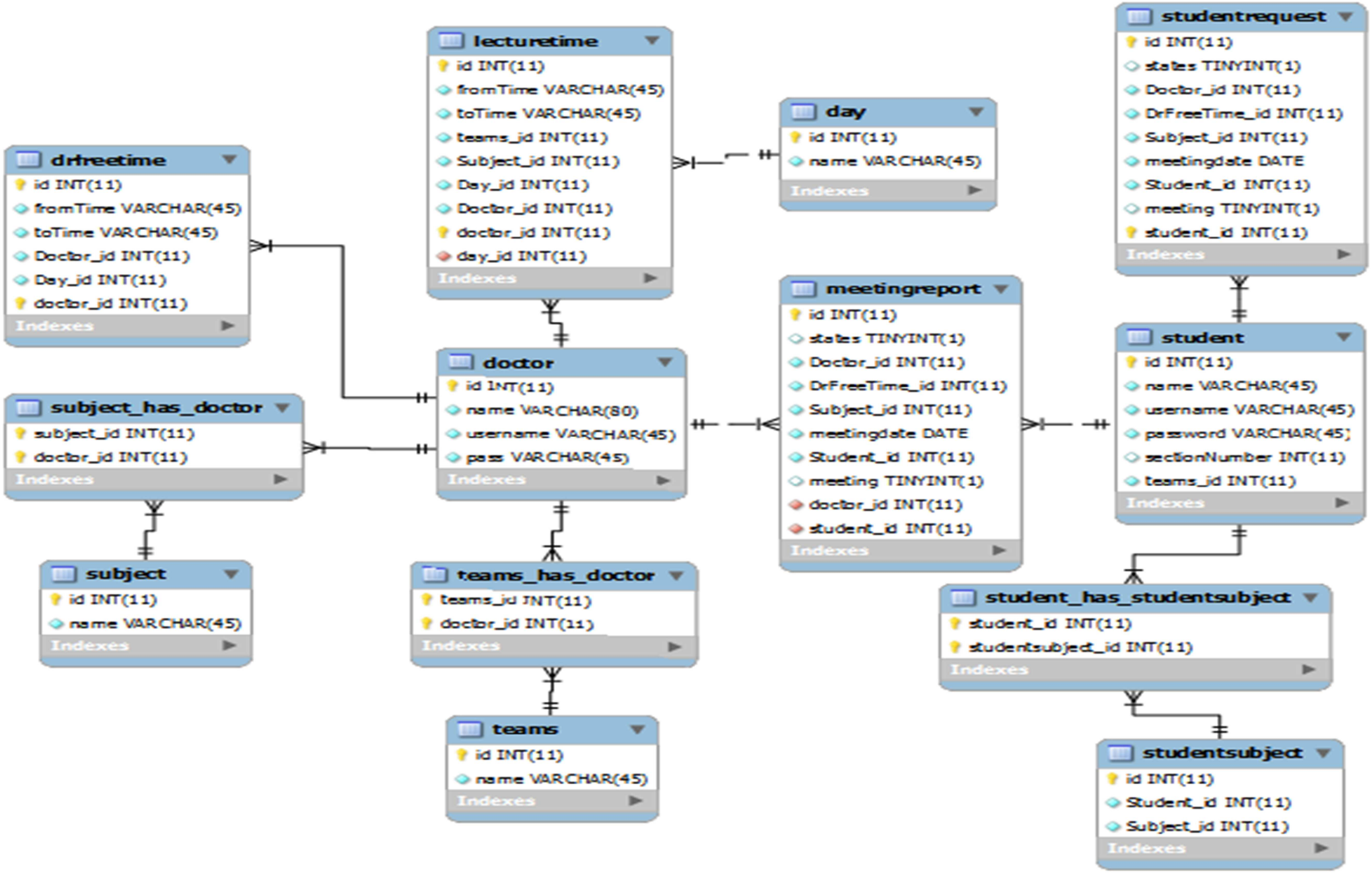


Figure 5.28: Intelligent Meeting Manager for Suspended Students Database Tables

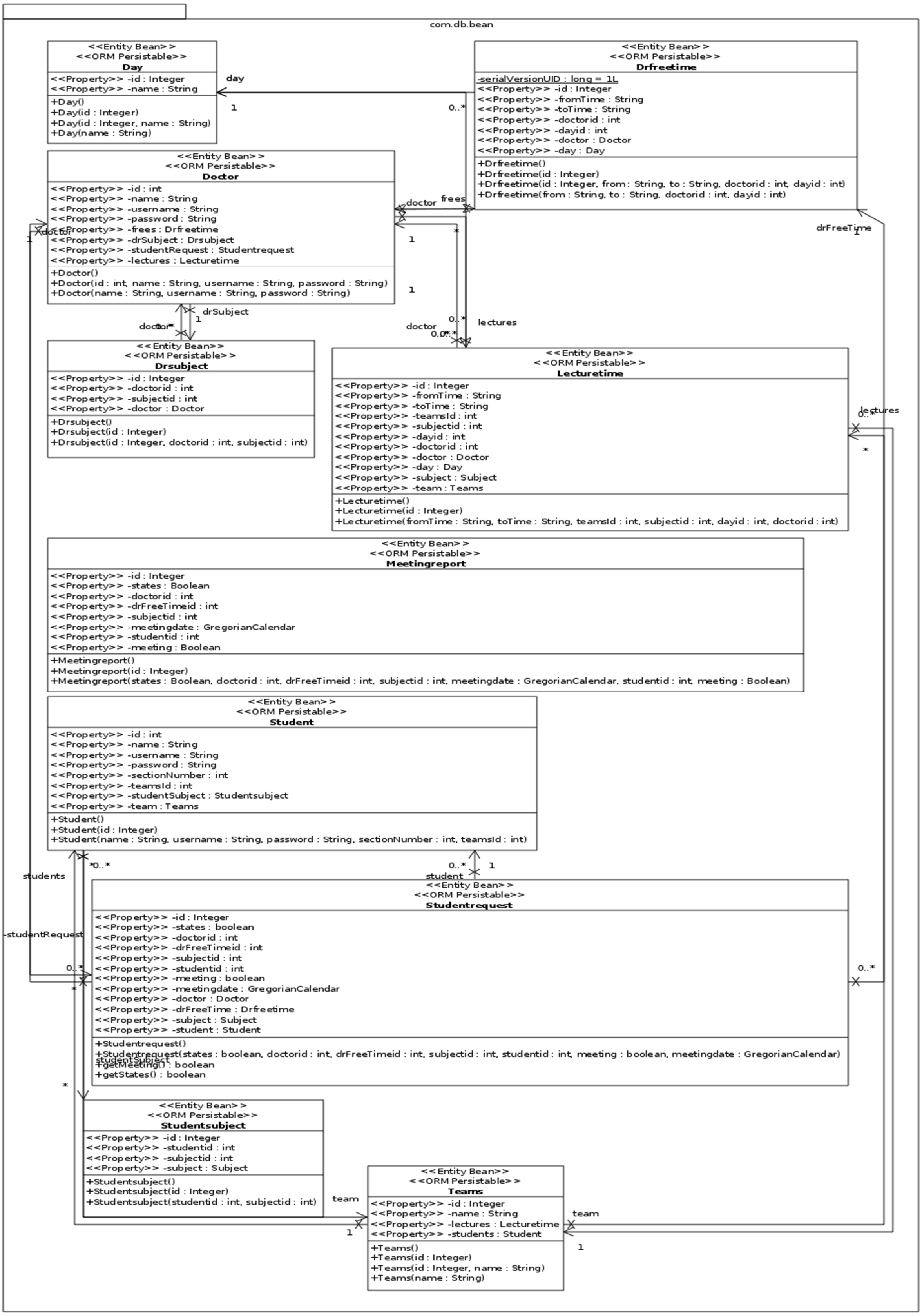


Figure 5.29: Intelligent Meeting Manager for Suspended Students Class Diagram

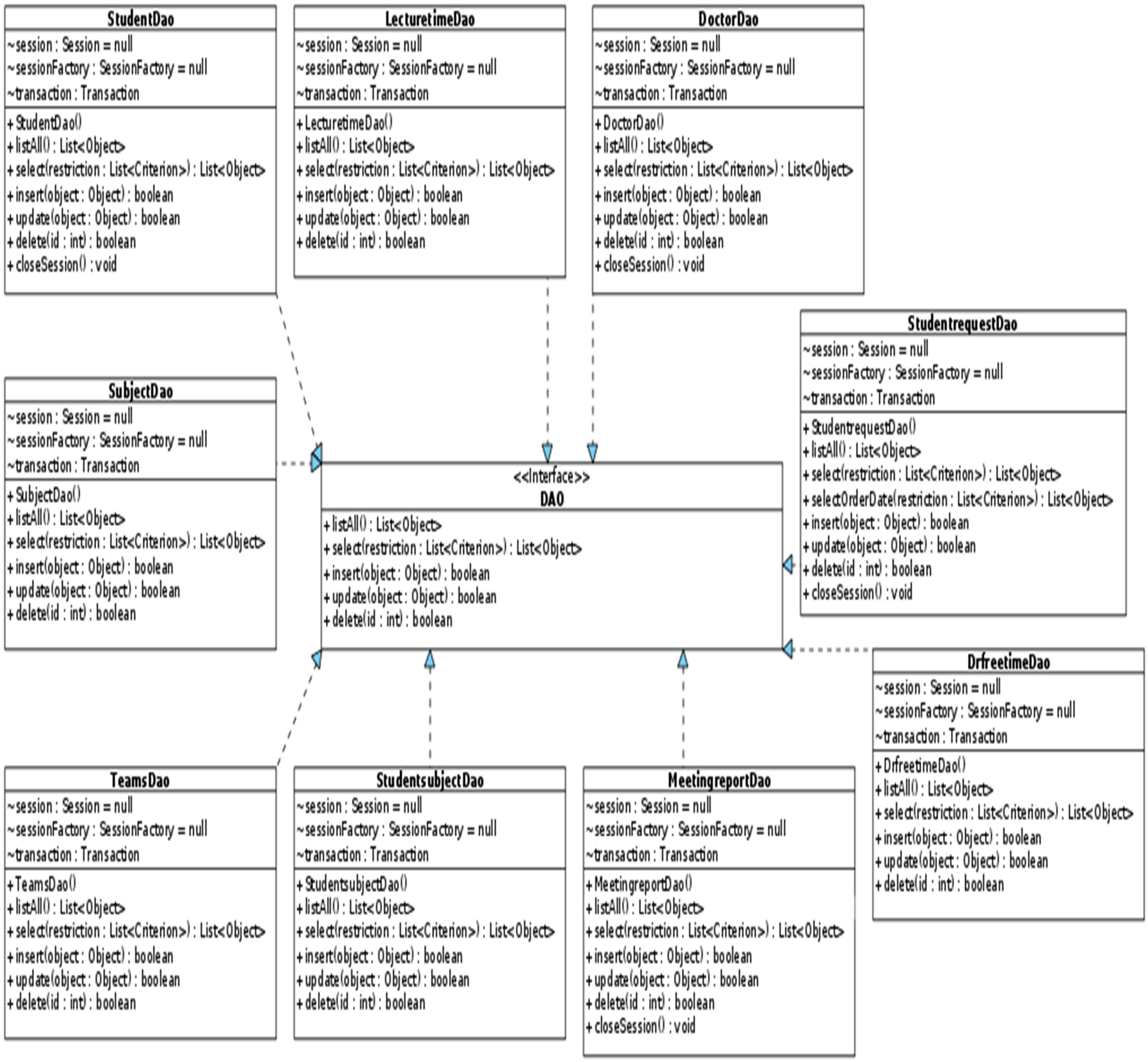


Figure 5.30: Intelligent Meeting Manager for Suspended Students Data Access Layer Class Diagram

5.8 Intelligent Document Classifier Service

Presented document classifier implements two of the Supervised Document Classification algorithms: Naive Bayes Classifier, and Term Frequency-Inverse Document Frequency (TF-IDF). Supervised learning algorithms refer to the service need to be trained first by taking the learning class and some identifying documents as an input, and then it can identify documents classes for non-classified documents. Figure 5.31 presents Intelligent Document Classifier Class Diagram.

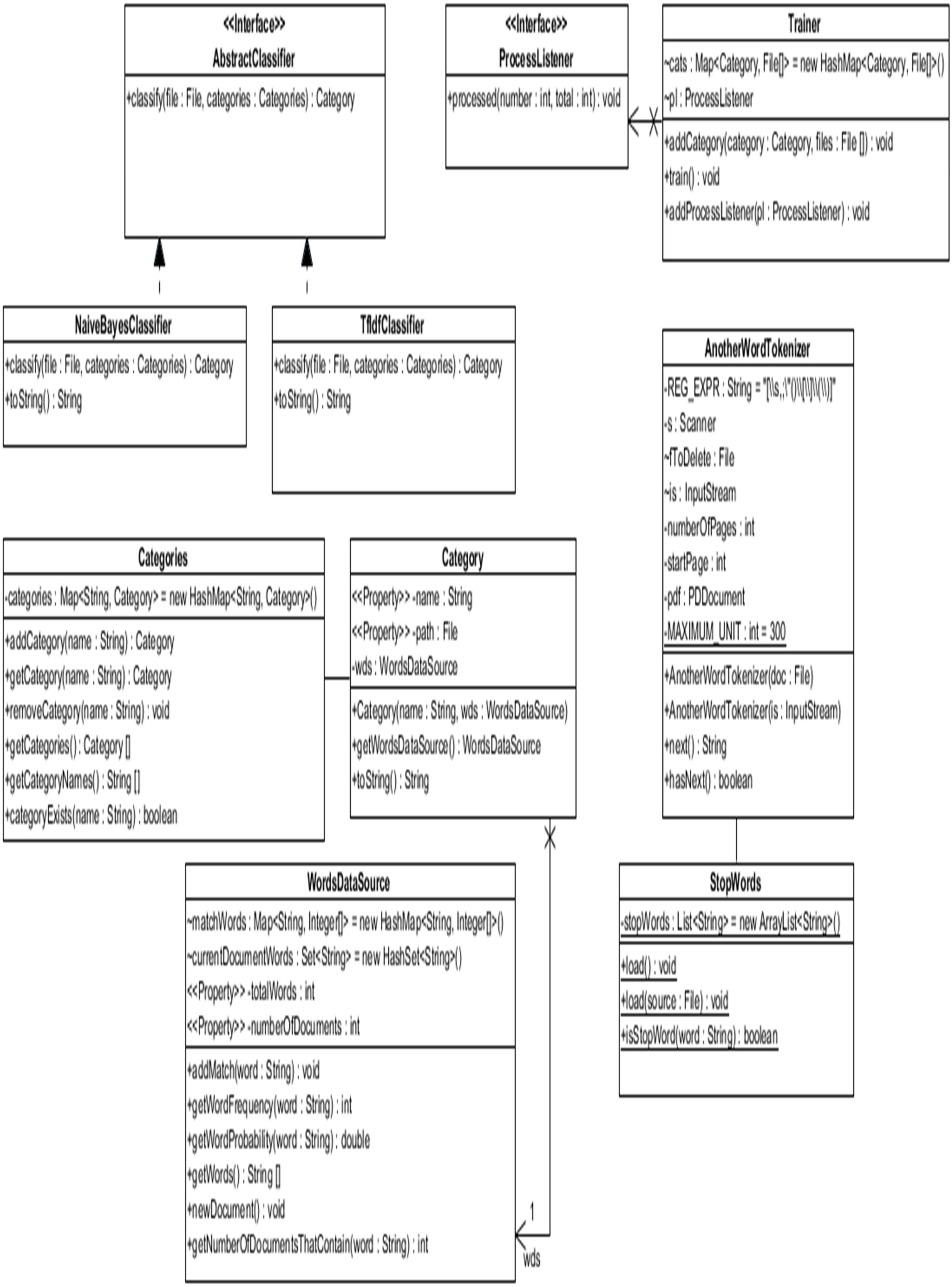


Figure 5.31: Document Classifier Class Diagram

5.9 Student e-Learning Environment and Adaptive Features

Figure 5.32 presents the students e-Learning environment presented through the desktop application. Figure 5.32 highlights different aspects presented at different parts of the application screen:

- Left Column "Top-Down":
  - Available Topics: Displays list of available topics to browse based on the student learning profile.
  - Student Calendar: highlighting next events, and giving student to add personal meetings.
  - Student Tracker Data: presenting to the student some useful information about student's habits and time consumed studying this module, total modules, and other tracking data.
  - Lecture PowerPoint Presentation: presents a link to the lecture PowerPoint presentation in case it is available by the instructor for the student to read and study.
- Middle Column "Top-Down":
  - Video Display Area: Displays recorded online lectures and can be used for further recommended videos section.
  - Chapter Study Area: Displays instructor recommended LOs for this topic.
- Right Column "Top-Down":
  - Displays recommended LOs for student based on topics keywords and students' preferences.

Figure 5.33 displays Learn Via Question "LVQ" Main Screen. LVQ is one of the adaptive features presented through the Adaptive e-Learning Model to give students opportunity to practice final exams before attending real exams. Immediate feedback and simulation of the exam environment is presented at this phase. Figure 34 presents LVQ sample test.

5.10 Instructor Portal

Instructor Portal is the central are where instructor can handle and manage all the functionalities related to the system from courses, topics, keywords, students, registration, and all other activities. Figure 5.35 presents an instructor screen of the instructor's portal.

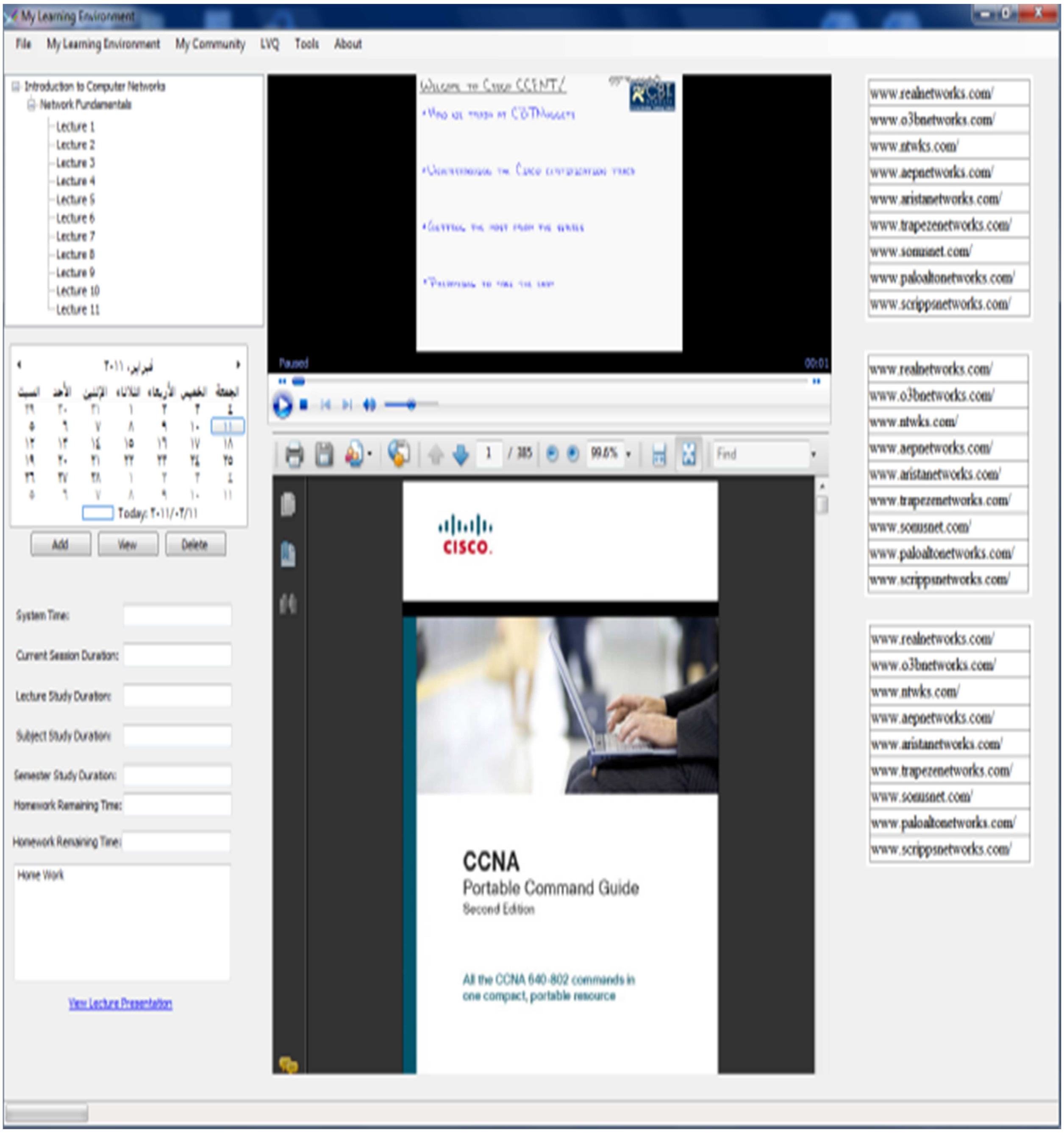
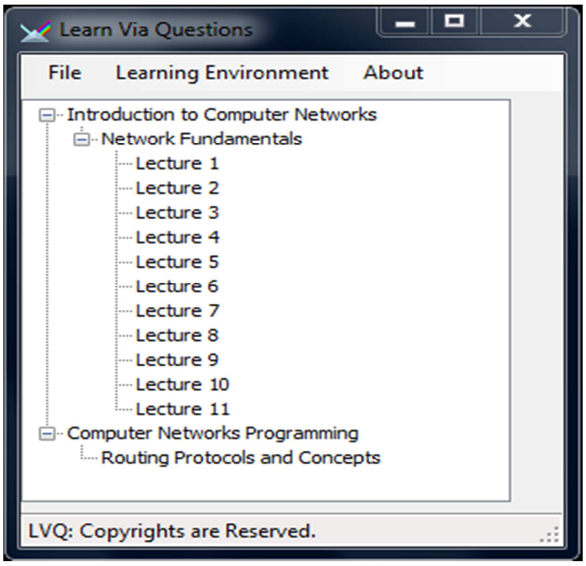


Figure 5.32: Student Learning Environment



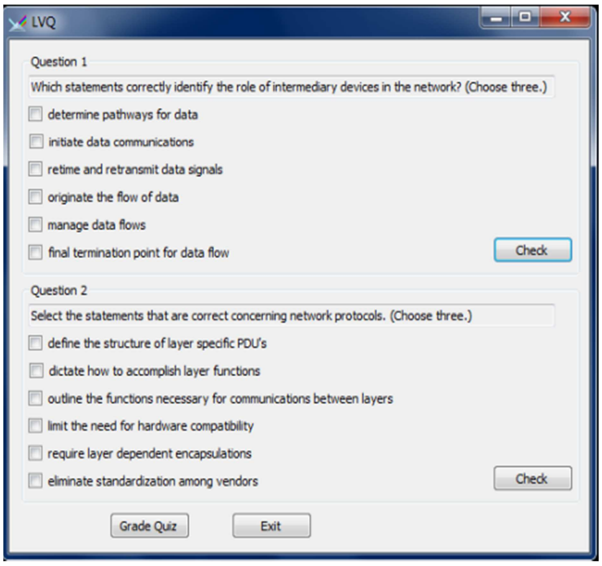


Figure 33: Learn Via Questions Main Screen

Figure 34: Learn Via Questions - Quiz Screen

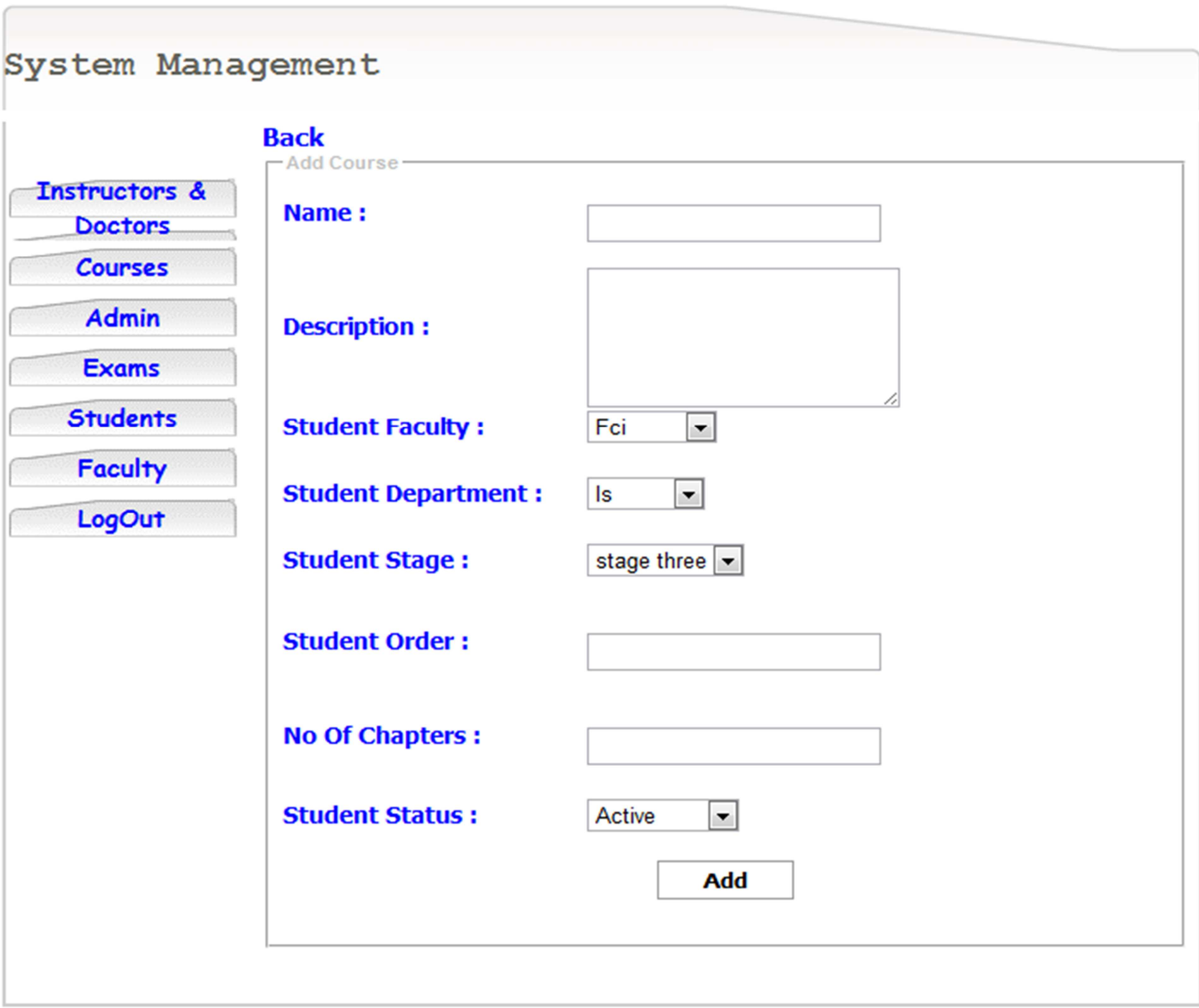


Figure 35: Instructor's Portal

5.11 Summary

This chapter presented different implementation details of the presented Adaptive e-Learning Models and Intelligent Services. Different database tables, class diagrams, and interfaces are presented. Presented details covers:

- Intelligent Student Tracking Service
- Students Manager Service
- Students Usage Data Manager
- LOs Manager Service
- Intelligent LOs Recommender: it includes:
    - Pending LOs for Recommendation Manager Module
    - Crawler Module
    - Document Processor Service: it consumes the following:
        - Tokenizer Module
        - Stopping Words Removal
        - Stemmer Module
        - Query Expansion
- Intelligent Meeting Manager for Suspended Students
- Intelligent Document Classifier

Chapter then presents some of the Student's e-Learning environment adaptive features mainly presented through the desktop application learning environment. The desktop application is not isolated from the web environment available to the student; however it is presented to overcome bandwidth and lack of internet availability challenges. Chapter then displays the instructor portal used mainly for managing the entire system.

Chapter 6

Chapter Six

Optimization and Evaluation

6.1 Introduction

Optimization is a continuous process that happens iteratively and recursively to achieve better: addressing and understanding the problems in hand, presenting different proposed solutions, optimizing the selected solution, applying the optimized and enhanced solution, evaluating the applied solution, and then back to the beginning of the optimization lifecycle. Optimization lifecycle is presented in figure 6.1. In this chapter, we review what we have presented through the dissertation quickly and focus on evaluation and optimization aspects of the proposed Adaptive e-Learning Models and Intelligent Services.

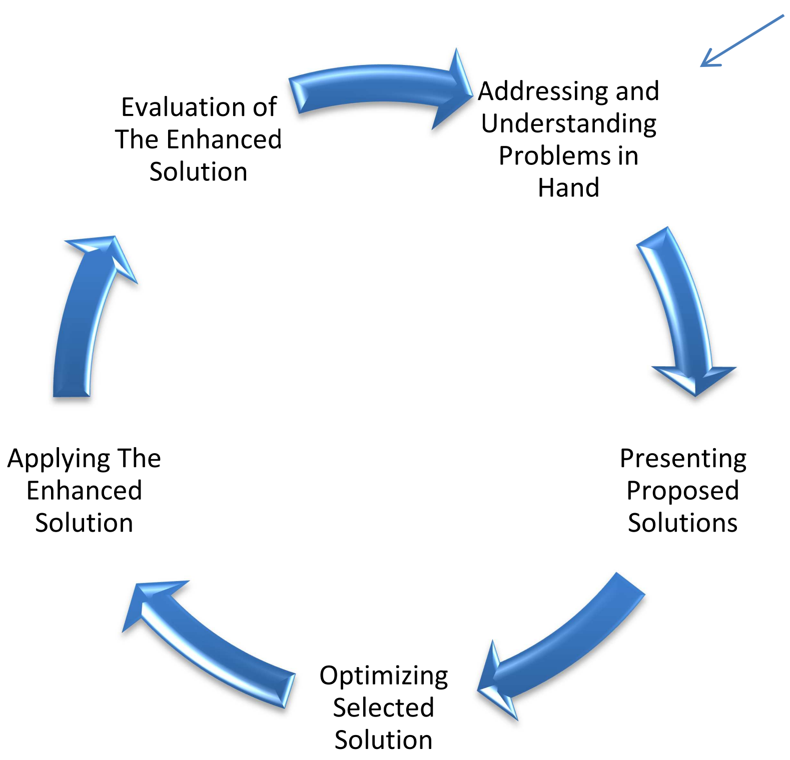


Figure 6.1: Optimization Lifecycle

6.2 Addressing Problems

Proposed Adaptive e-Learning Models addressed two problems categories: Technical and Pedagogical problems.

6.2.1 Pedagogical Problems

Reviewing current e-Learning status shows that current Blended Learning Model Paradigm faces many challenges; mainly pedagogical. To take a closer look at the problem, we conducted a pilot study for fourth year Information Systems department students at the faculty of Computers and Information Sciences in Mansoura University, Egypt, later at 2008. The good thing is that students believe in the efficiency of e-Learning and they are willing to use e-Learning and participate in e-Learning experiments, however there were problems. Pedagogical problems include:

- Students don’t use internet as their source of information although Internet is the main updated source of information and needs to be integrated in the learning environment.
- Most Students don’t know what “Tutorials” is. That means they haven’t utilized an online learning resources in a sequenced manner that helps them learn new topics at the end of a certain stage.
- Most students don’t access internet via their mobile phones, and they are not willing to participate in mobile learning experiments. They even believe mobile learning will not become popular in the near future.
- Students agreed that we utilize different forms of e-Learning. Though the faculty of the authors does not provide an official site for e-Learning, any online courses, assessment site or any other form of e-Learning other than the

authors' attempts, students still believe that e-Learning is efficient - even if they have not experienced it at all.

For more details about the pilot study, the reader can refer to chapter 3.

6.2.2 Technical Problems

Integrating University Management Information Systems (UMIS) and Learning Management Systems (LMS) is a need to achieve better e-Learning systems. Different software architecture patterns can be utilized in integrating both systems. Studying and analyzing many of the available software architectures has led to the realization of efficiency and effectiveness of utilizing Service Oriented Architecture (SOA) however it has led to different challenges. Challenges include the lack of performance efficiency when transmitting large amount of data. Besides, efforts of presenting adaptive and intelligent features of e-Learning have been spent without integration and presentation in current e-Learning systems. Web Service Software Factory design pattern as an example of Web services design patterns that builds everything within systems as Web services will yield to big performance degradation to proposed integrated e-Learning systems. Another technical optimization includes utilizing another integration technologies rather than Web services.

6.3 Presenting Proposed Solutions

Keeping the focus on Web and Desktop applications and shifting away from mobile learning is a direct result of students' unwillingness to experience and participate in mobile learning experiences. The pedagogical solutions focus on introducing adaptive and intelligent features in e-Learning to reach personalized environment. While the technical solutions focus on utilizing SOA in presenting those features. However, we still provide capability to integrate mobile learning activities using SOA.

6.3.1 Pedagogical Solutions

Different adaptive and intelligent pedagogical e-Learning solutions are presented to enable personalized e-Learning environment that will enhance students' e-Learning experience. The need for a personalized e-Learning system that incrementally and gradually takes students through an e-Learning experience is clear from the problems in hands section.

6.3.1.1 Adaptive Pedagogical Solutions

There are four main approaches that can be used to give a historical overview of Adaptive e-Learning. They are *Macro Adaptive*, *Aptitude-Treatment Interaction* (ATI), *Micro-Adaptive* and *Constructivistic-Collaborative* approaches. Proposed adaptive e-Learning model proposed in chapter three addresses those different adaptive e-Learning approaches and utilizes all of them. The four addressed approaches are:

- **Macro Adaptive Approach:** The proposed model addresses this capability by testing the student profile and learning preferences before establishing learning material.
- **Aptitude-Treatment Interaction (ATI) Approach:** The proposed model addresses allows the students to choose among the topics to learn (within the constraints of the pre-requisites). This gives them the partial control experience. Also, the proposed model provides the capability to arrange meetings between the instructors and students that have issues with certain learning topics. Students are given the chance to self-study the subjects and attend the exams 3 times. If the student fails to pass the exam 3 times, a

meeting must be arranged between the instructor and the student to submit a repost by the instructor to the student profile, so the student can continue the learning process again in the adaptive way. This sort of blended learning gives strength to the proposed model.

- **Micro-Adaptive Approach:** The proposed model addresses this approach by providing the capability to calculate the required time to study for each learning topic.
- **Constructivistic-Collaborative Approach:** Online forum, wiki and blog services will be available to students to enhance collaborate and help each other. Facilities to enable online study groups - like chatting applications - can be made available. Arguments around the effectiveness of Web 2.0 features in e-Learning are taking place all around.

6.3.1.2 Intelligent Pedagogical Solutions

Intelligent techniques are presented to empower the proposed Adaptive e-Learning Models. Proposed models shed lights on supporting e-Learning with intelligent features. One of the main features supported by our model is supervised intelligent curriculum sequencing to present adaptive e-Learning fine-tuned by the instructor. Nine intelligent services that can be utilized in different e-Learning functionalities were presented. These intelligent services are grouped into two categories based on their aims: *Instructor Services* and *Student Services*.

The Instructor Intelligent Services are: *Intelligent Learning Object (LO) Classifier service, Intelligent Online Lecture LOs Advisor*, *Intelligent Student Performance Tracker and Intelligent Cheating Depressor.* The Student Intelligent Services are: *Intelligent Time-to-Learn Topic Calculation, Intelligent*

Study Plan Advisor, *Intelligent Agenda Study Time Planner*, *Intelligent Meeting Manager for Suspended Students and Intelligent LOs Recommender*.

Fuzzy Logic is the intelligent technique used in different aspects of the services to enable different functionalities as presented in chapter four.

6.3.2 Technical Solutions

Combining both Business Process Management (BPM) and Service Oriented Architecture (SOA) is proven to achieve numerous advantageous features for systems. Presenting adaptive and intelligent features as services with standard interfaces will allow different e-Learning systems to adopt them, so they will be reusable and newly introduced information systems will not have to redo the work again, besides, wrapping adaptive and intelligent features with standard interfaces will present a separation of interests that help adaptive and intelligent features' researchers and developers to focus more on their target and transfer the responsibility of utilizing those features in different information systems to information systems specialists. Presented Adaptive e-Learning Models presented an adaptive learning process that adaptively changes based on students' performance. Proposed model composing services can be categorized in the following layers:

- **Orchestration Layer:** holds services responsible for maintaining learning process logic and activities. It includes services that utilize both composite services' layer services, and data services' layer services.
- **Composite Services:** are services that hold other services and don't complete functioning unless all composing services execute successfully; however it is not controlling them.
- **Data Services Layer (Information as a Service "IaaS"):** is the layer that holds services responsible for transforming Meta-data into meaningful information to other utilizing information systems, instructors, and students.

▪ **Model Layer (Database):** it is the database layer that holds data tables.

Though Web services provide technology neutral interface to be utilized online through standard URL, J2EE Connector Architecture (JCA) connects Java based applications in an enhanced performance measures when compared to XML Web services. Another technical optimization point includes using emerging protocols that are designed for Web. The Open Data Protocol (OData) is a web protocol for querying and updating data. OData applies web technologies such as HTTP, Atom Publishing Protocol (AtomPub) and JSON to provide access to information from a variety of applications, services, and stores. OData is being used to expose and access information from a variety of sources, including but not limited to relational databases, file systems, content management systems, and traditional web sites.

6.4 Optimizing Selected Solution

Applying the presented solution yielded some challenges and bottlenecks that need optimization. Optimization in this phase focuses on the system performance. Intelligent LOs Recommender is the one of the core services and lies in the heart of the proposed Adaptive e-Learning Models. It is utilized in different intelligent services. However, upon deployment, Intelligent LOs Recommender met challenges that affected its performance and efficiency. Optimizing it will affect the presented models overall performance. Intelligent LOs Recommender is evaluated from Performance perspective in order to accelerate overall system performance.

6.4.1 Intelligent LOs Recommender Challenges

To empower the presented adaptive e-Learning models, we designed and built the Intelligent LOs Recommender. Intelligent LOs Recommender was tested on different files (221) and online LOs (342). Files LOs generated 7388 tokenized Term Frequencies (TF) and online LOs generated 169876 TFs. Challenges in designing and implementing the Intelligent LOs Recommender include:

1. **Identifying Seeds:** with the increasing number of online content, defining seeds for the crawler is an important task.
2. **Identifying Online and Offline Phases:** identifying bottlenecks in the Intelligent LOs Recommender performance is an important issue to avoid dead ends and long times of processing that affects systems.
3. **Evaluating Accuracy of Generated Terms:** generated terms shall be evaluated to avoid non-related terms.

Unleashing an online crawler to collect different LOs available online, and storing meta-data about them in an offline database, with URLs made available for later visits of the Intelligent LOs Recommender service was the first task achieved in building this service. Visiting those URLs later, retrieving the LOs, tokenizing and stemming, calculating Term Frequencies, and storing calculated Term Frequencies in the database for later matching with course objectives. Random LOs group of the crawlers results are used as the test set with capacity of 254 LOs. Reading time in seconds for each of those LOs are presented in Figure 6.2. Table 6.1 presents summary of the main statistical measures of LOs reading times in seconds.

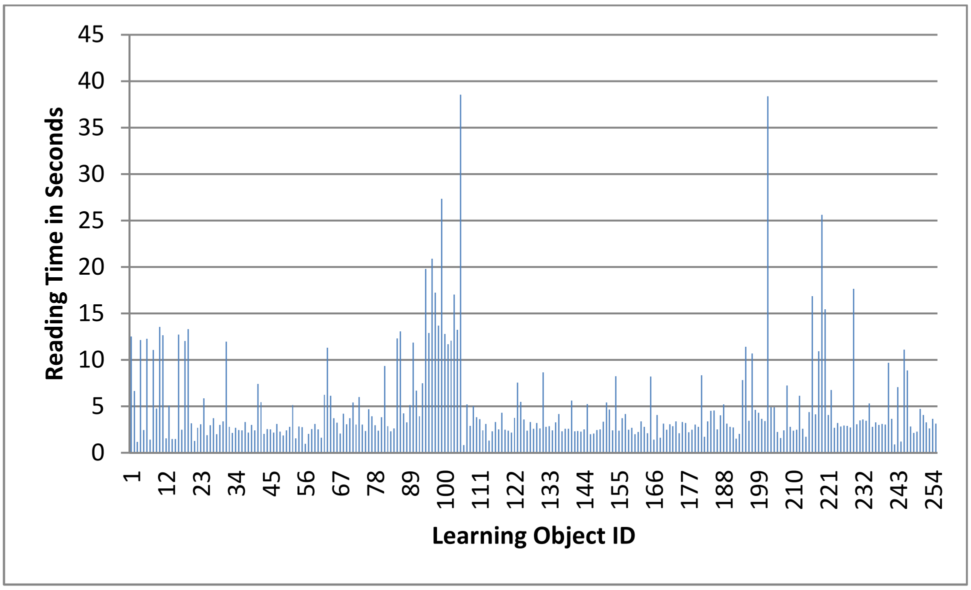


Figure 6.2: Learning Objects Reading Time in Seconds

Table 6.1: Summary of the Main Statistical Measures of Learning Objects Reading Times in Seconds

Min.	0.819568872	**Mean**	5.108815395
Max.	38.5457058	**Mode**	N/A
Range	37.72613692	**Median**	3.159333944

Reading times fall in an average time of five seconds in retrieving the LO. Optimizing LOs retrieval can be done through increasing network bandwidth and the servers' memory that affects window sizing. Tokenization duration for retrieved LOs is presented in Figure 6.3 followed by Table 6.2 summarizing the main statistical measures of tokenizing LOs times in seconds. Tokenization duration falls below half a second at its worst case, and it is believed that tokenization is in an optimized form.

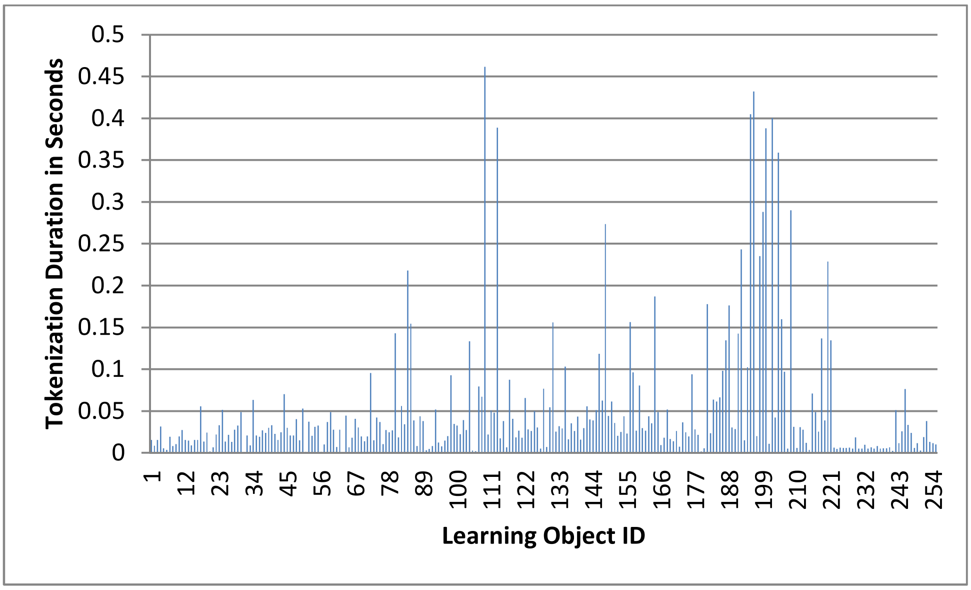


Figure 6.3: Learning Objects Tokenization Duration in Seconds

Table 6.2: Main Statistical Measures of Learning Objects Tokenization Times in Seconds

Min.	0.000496149	**Mean**	0.051718733
Max.	0.461540937	**Mode**	N/A
Range	0.461044788	**Median**	0.026743531

Term Frequencies calculation time for each LO is presented in Figure 6.4 followed by Table 6.3 that highlights a summary of the main statistical measures of Learning Objects Term Frequencies calculation time in seconds. TF processing ranges are from fractions of milliseconds to less than three seconds for the LO. TF processing is efficient enough to be utilized in the recommendation process.

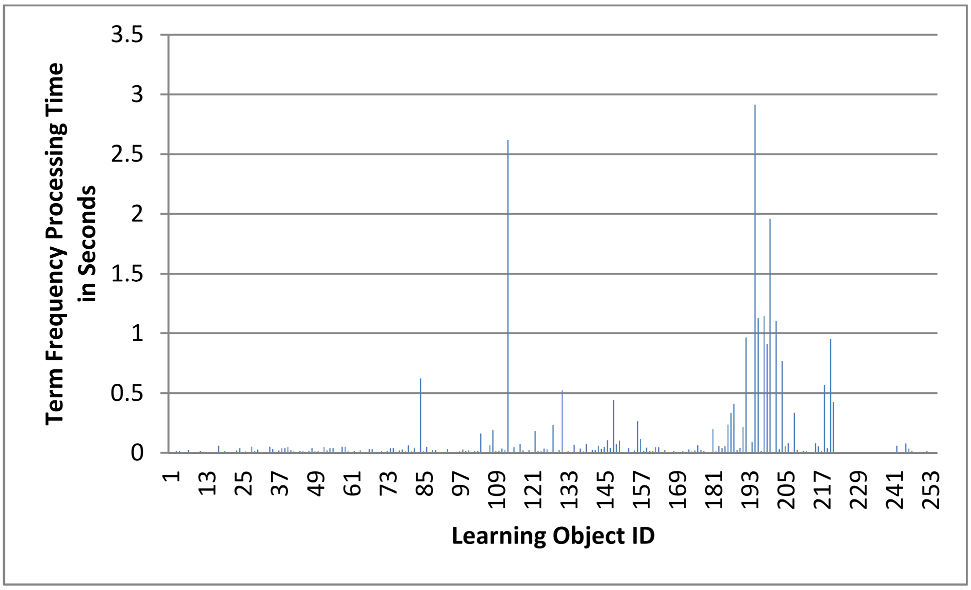


Figure 6.4: Learning Objects Term Frequencies Calculation Times in Seconds

Table 6.3: Main Statistical Measures of Term Frequency Calculation Time in Seconds

Min.	6.48499E-05	**Mean**	0.095572824
Max.	2.913298845	**Mode**	0.001230955
Range	2.913233995	**Median**	0.015192032

The first performance bottleneck appears in extracted terms database insertion times. Figure 6.5 presents LOs keywords insertion time in seconds. As summarized in Table 6.4, average keywords insertion time for LO is 373 seconds, with worst cases exceeding 2536 seconds. Such performance issue is not accepted and leading to dead ends in the system. It takes a lot of time to insert keywords in the database for later recommendation processing.

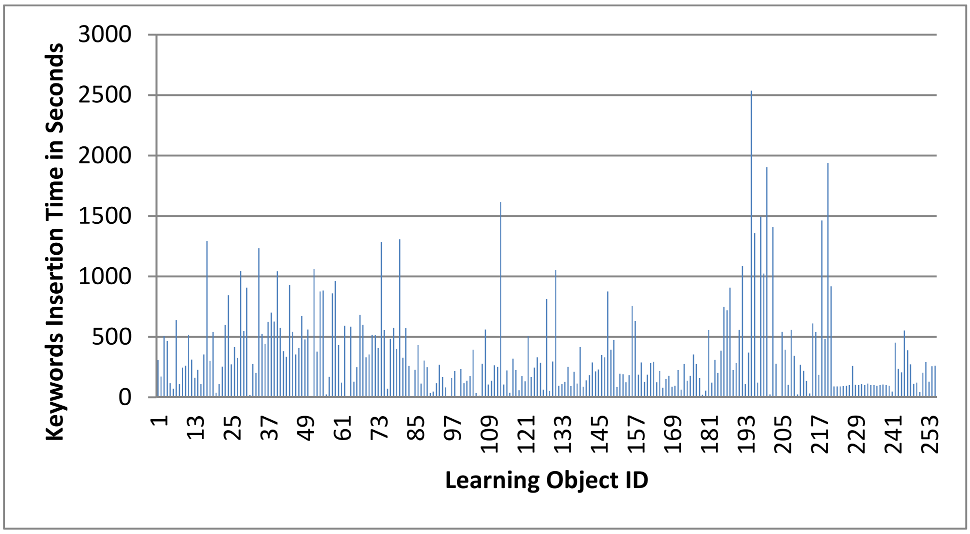


Figure 6.5: Learning Objects Keywords Insertion Times in Seconds

Table 6.4: Main Statistical Measures of Learning Objects Keywords Insertion Time in Seconds

Min.	4.576031208	**Mean**	373.0938357
Max.	2536.919661	**Mode**	N/A
Range	2532.34363	**Median**	260.3440971

Figure 6.6 compares the percentage that tokenized number of words and the total number of words contributes to the total. Tokenized number of words when compared to total number of words doesn’t exceed 15% by anyhow. One challenge with online LOs is the tremendous amount of Hyper Text Markup Language (HTML) used for web based user interface. Tokenization process is responsible for handling this challenge. Regular Expressions (RE) are used to extract text from online LOs. Tables 6.5 and 6 present statistical measures about total and tokenized number of words respectively.

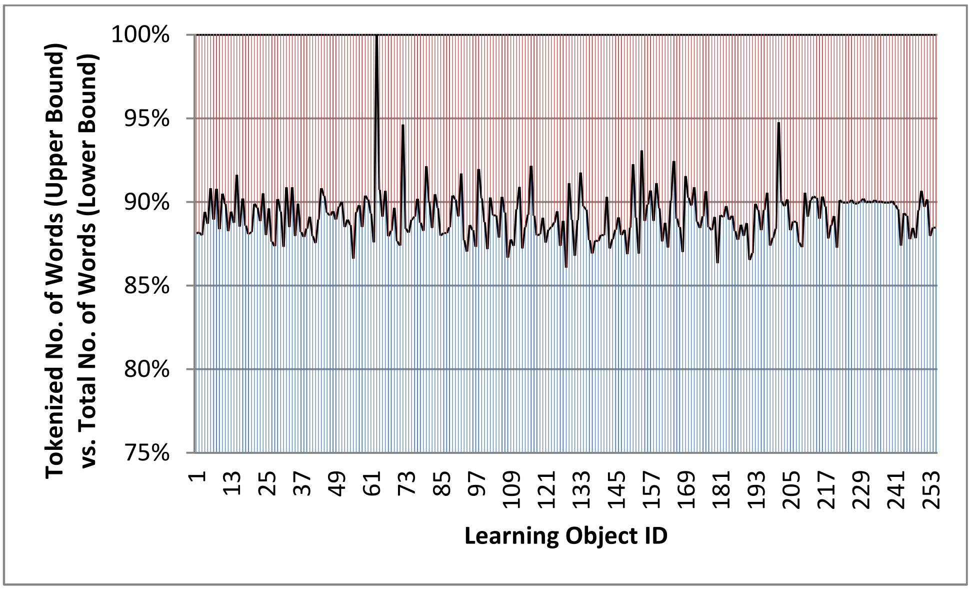


Figure 6.6: Learning Objects Tokenized No. of Words vs. Total No. of Words

Table 6.5: Main Statistical Measures of Learning Objects Original No. of Words Count

For Total Learning Object Words			
Min.	5	**Mean**	7238.443137
Max.	66491	**Mode**	1085
Range	66486	**Median**	3803

Table 6.6: Main Statistical Measures of Learning Objects Tokenized Number of Words Count

For Tokenized Words			
Min.	0	**Mean**	912.6745098
Max.	9737	**Mode**	121
Range	9737	**Median**	469

Though tokenized number of words percentage when compared to total number of words doesn't exceed 15%, there is still a challenge facing our proposed Intelligent LOs Recommender which is the tremendous amount of extracted keywords that is not related to course objectives. Processing books for example, is a big challenge. One of the processed books was read in less than 0.25 a second with 12522837 original words number, generated 258457 tokenized words (1.6 MB of data) in 2188 seconds for tokenization processing, and Term Frequency calculation of 521 seconds. Such processing never finished uploading extracted keywords into the database.

Situation changes a lot when adding the course objectives into inputs. Course objectives extracted keywords and expanded by WordNet to increase system's efficiency percentage when compared to number of tokenized words and total number of words is presented in Figure 6.7. Handling more than the really needed keywords yields a big performance degradation that we can get rid of by presenting a design solution and taking the decision of including course objectives as an input parameter, thus coming over the performance bottleneck. Figure 6.8 followed by Table 6.7 presents the optimization gained in insertion.

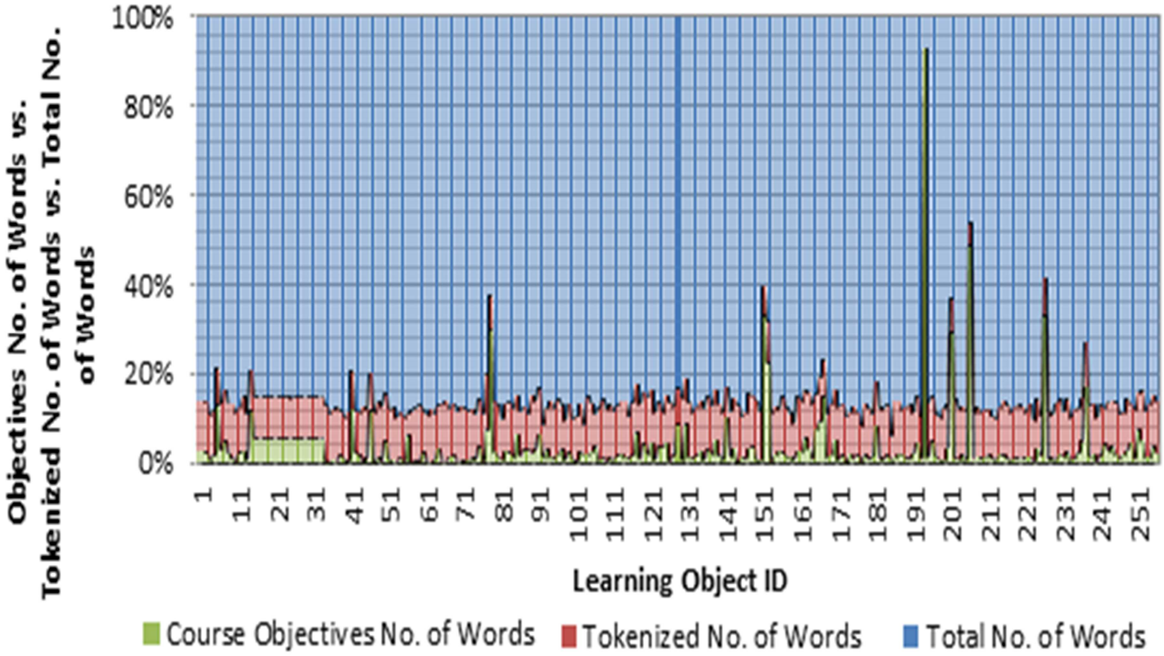


Figure 6.7: Course Objectives Extracted and Expanded Keywords vs. Tokenized No. of Words vs. Total No. of Words

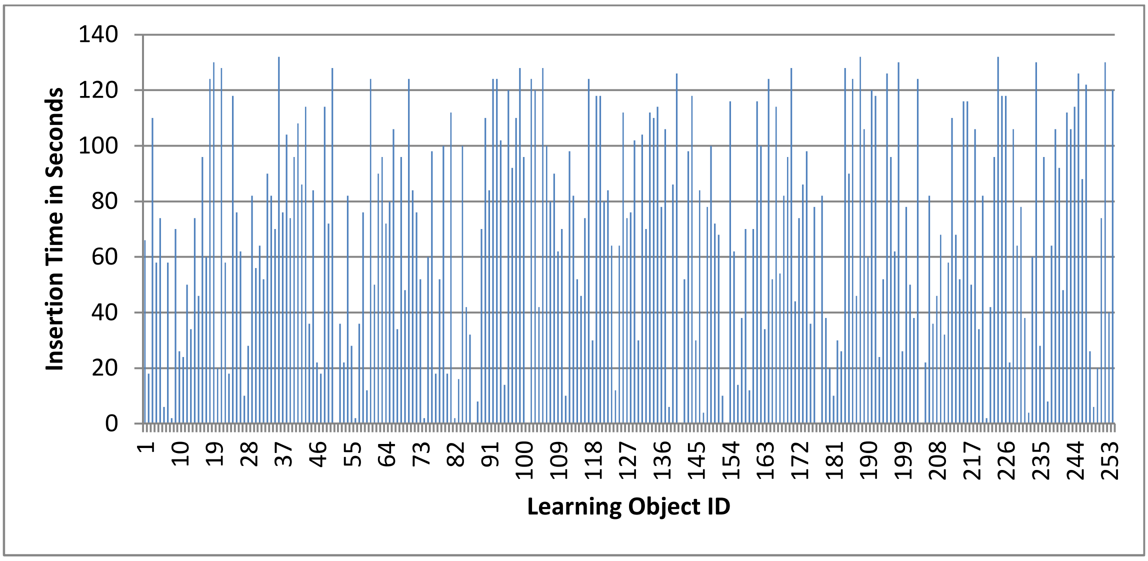


Figure 6.8: Learning Objects Optimized Keywords Insertion Times in Seconds

Table 6.7: Main Statistical Measures of Learning Objects Keywords Insertion Time in Seconds

Min.	0	**Mean**	124
Max.	132	**Mode**	N/A
Range	132	**Median**	74

6.4.2 Intelligent LOs Recommender Optimization Techniques and Comments on Results

Optimizing and Enhancing the Intelligent LOs Recommender can be done through different design decisions:

- **Taking course objectives into consideration while determining crawler seeds:** a preprocessing step that processes course objectives and generate search keywords that are used to find related web sites using one of the many internet search engines helps in finding more near and accurate seeds for the crawler. In our proposed model, we developed a java based crawler that takes course objectives keywords as input, uses Google search engine to invoke the search query, loops through search results, and stores meta-data about found URLs in the database.
- **Query Expansion:** to increase the accuracy of search terms resulting from processing course objectives and specifications, query expansion methods

can be utilized. WordNet is a lexical database for the English language that groups English words into sets of synonyms to provide short, general definitions, and records the various semantic relations between these synonym sets. Extracted keywords are expanded by WordNet synonyms and then passed to the crawler when seeking seeds.

- **Taking course objectives into consideration while calculating Term Frequencies for Learning Objects:** the main objective of Intelligent LOs Recommender is to intelligently contribute in personalizing the learning experience for the student by recommending the most accurate LOs, not indexing the online LOs. So, there is no need to get stuck in analyzing what doesn’t matter for the recommendation process. Enhanced solution will not calculate frequencies for terms that don’t exist in curse objectives and will not store term frequencies at all. The processing time and cost is much cheaper when compared to storage cost.

6.5 Evaluating Optimized Solution

Presented Adaptive e-Learning Models and Intelligent Services shall be evaluated from different perspectives. Evaluation perspectives include:

1. **User Satisfaction:** Users here are both students and instructors.
2. **Information Retrieval** Evaluation of Intelligent LOs Recommender.
3. **Intelligent LOs Classifier** Evaluation.

Following sections present those three evaluation aspects and results.

5.1 User Satisfaction Evaluation

Preparing a Computer Networks Course and presenting it to students in the form of the presented Adaptive e-Learning Models and experiencing it, resulted in the following satisfaction measures. Table 6.8 presents summary of students' opinion about presented features and how they evaluate the need for it and its performance. Presented Adaptive e-Learning Model was tested on sample of 10 students. Table 6.9 presents summary of instructors' thoughts about presented features and how they evaluate the need for it and its performance and behavior.

Table 6.8: Summary of Students' Evaluation of Presented Adaptive e-Learning Models Features

Feature	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Learning Preferences	90%	10%	--	--	--
Learning Profile	85%	15%	--	--	--
Customizing Course within Rules	80%	10%	10%	--	--
Separate Groups	30%	10%	10%	25%	25%
Exams Check Points	30%	10%	10%	30%	20%
LVQ	90%	5%	5%	--	--
Video LOs	90%	5%	5%	--	--
Intelligent LOs Recommender	70%	20%	--	5%	5%
Intelligent Agenda Study Time Planner	63%	17%	5%	8%	7%
Intelligent Study Plan Advisor	65%	30%	--	5%	--
Intelligent Meeting Manager for Suspended Students	40%	10%	8%	2%	40%

Table 6.9: Summary of Instructors' Evaluation of Presented Adaptive e-Learning Models Features

Feature	Strongly Agree	Agree	Neutral	Disagree	Strongly Disagree
Learning Preferences	90%	10%	--	--	--
Learning Profile	90%	10%	--	--	--
Customizing Course within Rules	90%	10%	--	--	--
Separate Groups	70%	20%	10%	--	--
Exams Check Points	70%	15%	15%	--	--
LVQ	90%	5%	5%	--	--
Video LOs	90%	5%	5%	--	--
Intelligent LOs Recommender	70%	20%	10%	--	--
Intelligent Online Lecture LOs Advisor	73%	17%	10%	--	--
Intelligent Cheat Depressor	50%	10%	10%	20%	10%
Intelligent Student Tracker	80%	20%	--	--	--

6.5.2 Information Retrieval Evaluation

One of the fundamental problems in Information Retrieval (IR) is the ranking problem, ordering the results of a query such that the most relevant results show up first. Ranking algorithms employ scoring functions that assign scores to each result of a query at hand. So, ranking the results of a query consists of assigning a score to each result and then sorting the results by score, from highest to lowest. Many performance measures are used by IR community to evaluate the effectiveness of ranking functions.

In order to measure the relevance of the Ranked Recommended LOs list, three performance measures are used namely: Precision, Recall, and F-measure. The first performance measure used is Precision. Precision measures the ratio of relevant documents within a given number of documents returned to the number of returned documents. The second performance measure is Recall. Recall is defined as the number of relevant documents retrieved by a search divided by the total number of existing relevant documents (which should have been retrieved). The Recall measure quantifies what fraction of all the relevant results was ranked to fall within the first k documents. Precision and Recall scores are not discussed in isolation. Instead, both may be combined into a single measure, such as the F-Measure. F-Measure is the weighted harmonic mean of precision and recall. Figure 6.9 shows the precision measures of the proposed Intelligent LOs Recommender followed by Table 6.10 highlighting a summary of the main statistical measures. Figure 6.10 shows the recall measures of the proposed Intelligent LOs Recommender followed by Table 6.11 highlighting a summary of the main statistical measures. Figure 6.11 shows the F-Measure of the proposed Intelligent LOs Recommender followed by Table 6.12 highlighting a summary of the main statistical measures.

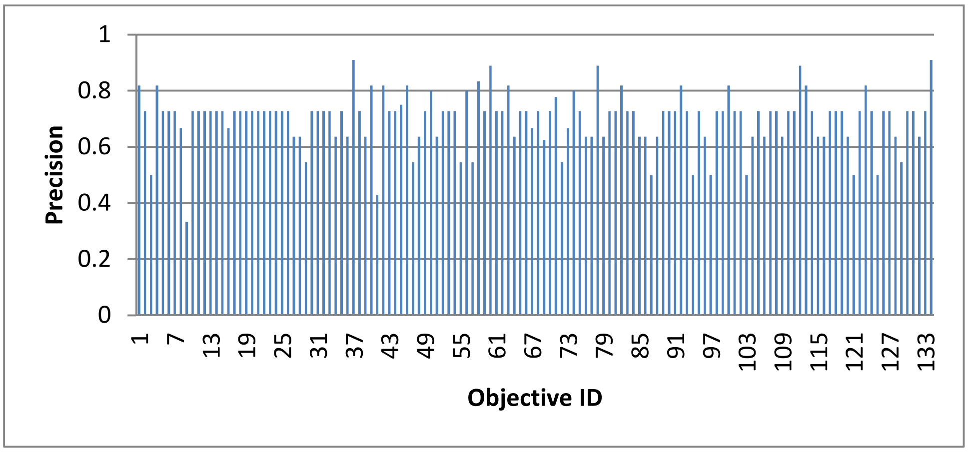


Figure 6.9: Precision Evaluation of Proposed Intelligent LOs Recommender

Table 6.10: Summary of the Main Statistical Measures of Precision

Min.	0.333333333	**Mean**	0.700674926
Max.	0.909090909	**Mode**	0.727272727
Variance	0.008883086	**Median**	0.727272727

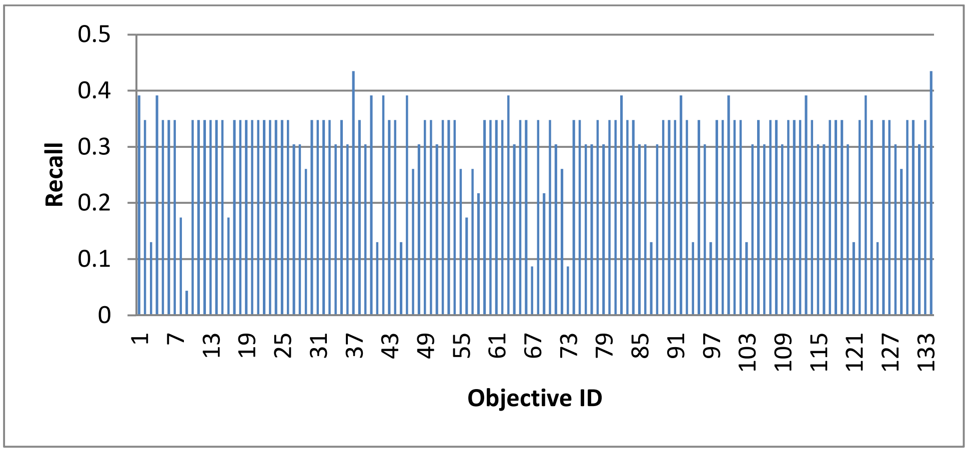


Figure 6.10: Recall Evaluation of Proposed Intelligent LOs Recommender

Table 6.11: Summary of the Main Statistical Measures of Recall

Min.	0.434782609	**Mean**	0.31440623
Max.	0.434782609	**Mode**	0.347826087
Variance	0.005640442	**Median**	0.347826087

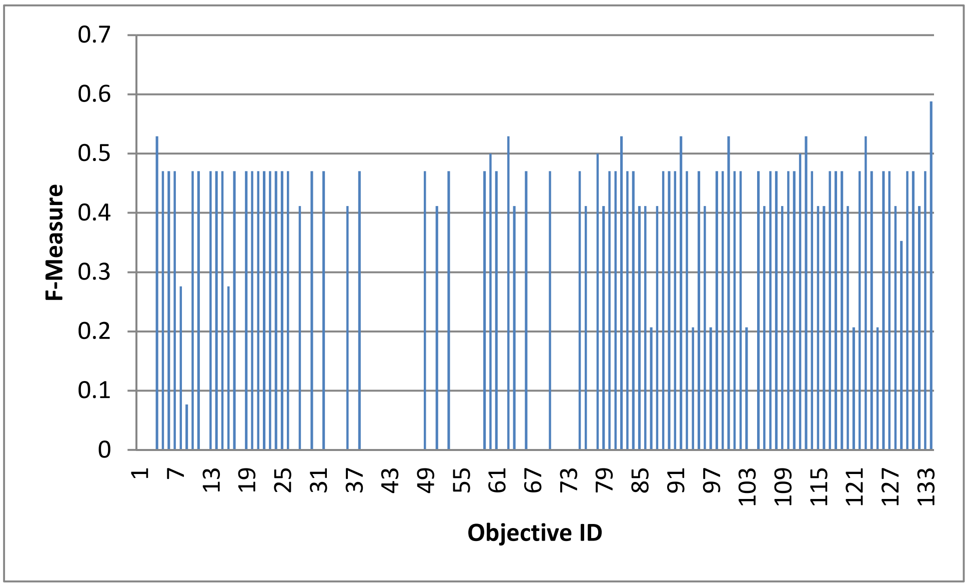


Figure 6.11: F-Measure Evaluation of Proposed Intelligent LOs Recommender

Table 6.12: Summary of the Main Statistical Measures of F-Measure

Min.	0	**Mean**	0.308731541
Max.	0.588235294	**Mode**	0.470588235
Variance	0.04558944	**Median**	0.441176471

6.5.3 Intelligent LOs Classifier Evaluation

Presented Intelligent LOs Classifier is evaluated to check its accuracy and capability to classify unclassified documents. Intelligent LOs Classifier implements Naïve-Bayes classifier algorithm. The following training set was given to the classifier. Table 6.13 presents categories and documents for each document that the classifier was trained on. Table 14 presents the results of testing. Testing set is presented to the trained classifier. Intelligent LOs Classifier resulted in 100% accuracy.

Table 6.13: Summary of Intelligent LOs Classifier Training Set

Category	Training Document
Bioinformatics	
	Biologically Inspired Computation
	Dynamical Systems in Neuroscience
	Molecular Analysis of Cancer
	Probabilistic Models of the Brain Perception and Neural Function
BPM	
	BPM Success: How a Travel Giant Turned its Ship Around CIO
Computer Networks	
	Interconnecting CISCO network devices Part 1
	Interconnecting CISCO network devices Part 2
	Portable Command Guide
e-Learning	
	101 free e-Learning tools
	Adaptive and Personal LMS

	E-Learning Technologies
	Questionmark-Tools-Effective-Assessments
	SCROM-v1
English	
	New interchange 1-key
	New interchange 1-student book
	New interchange 1-workbook
	Intro Workbook 3rd edition
Programming	
	ASPNet in VB
	ASPnet MVC
	Applied Numeric Methods using Matlab
	Essential Matbal for Engineers and Scientists
	Visual Basic 2008
SOA	
	SOA Description
	SOA Lab Setup Guide
	SOA Release Notes
	SOA Design Patterns
	SOA Instructor Exercises Guide
	SOA Lab Setup Guide Classroom
	SOA Lab Setup Guide Online
	SOA Student Exercises
	SOA Student Book

Table 6.14: Summary of Intelligent LOs Classifier Testing Set

Document	**e-learning by design**	
Class Percentage		
	English	NaN
	e-Learning	-17510.210584564997

	Bioinformatics	-20216.649226425725
	BPM	-17698.0930699629
	Computer Networks	-20366.694928274974
	SOA	-19893.456729741945
	Programming	-22055.679402138958
Decision	**e-Learning**	
Document	**effective e-Learning environment personalization using web usage mining technology**	
Class Percentage		
	English	NaN
	e-Learning	-21879.011486226136
	Bioinformatics	-25730.60082626081
	BPM	-25055.906020316877
	Computer Networks	-25432.70422899807
	SOA	-25199.419281366278
	Programming	-28851.363753332593
Decision	**e-Learning**	
Document	**CCNP2_SLM_v50**	
Class Percentage		
	English	NaN
	e-Learning	-769103.2978993196
	Bioinformatics	-839262.6497608843
	BPM	-695368.4959748124
	Computer Networks	-610877.2165053524
	SOA	-784774.4069853874
	Programming	-855032.0508602582
Decision	**Computer Networks**	
Document	**forrester_bpm_wave**	

Class Percentage		
	English	NaN
	e-Learning	-83003.91116782578
	Bioinformatics	-94739.36297244373
	BPM	-78479.22990639707
	Computer Networks	-89490.77817552155
	SOA	-84576.42321484128
	Programming	-101919.05505022638
Decision	**BPM**	
Document	**Network_Fundamentals_2D_IRG**	
Class Percentage		
	English	NaN
	e-Learning	-31300.31339492937
	Bioinformatics	-33352.432161580524
	BPM	-29577.875485405537
	Computer Networks	-29199.402824383295
	SOA	-32654.278796223974
	Programming	-36143.44494075163
Decision	**Computer Networks**	
Document	**Prentice.Hall.SOA.Principles.of.Service.Design.Jul.2007**	
Class Percentage		
	English	**NaN**
	e-Learning	**-810529.5972492553**
	Bioinformatics	**-991026.2503207972**
	BPM	**-962086.7346663276**
	Computer Networks	**-933969.3054615034**
	SOA	**-730561.9506475903**
	Programming	**-1049997.9439706628**
Decision	**SOA**	

6.6 Summary

This chapter presents the optimization activity we have been through this dissertation and highlights the optimization concepts have been focused on. Helping e-Learning in presenting new Adaptive e-Learning Models that enhances the e-Learning experience, presenting new Intelligent Services to provide advanced functionalities that cannot be achieved using standard methods, and enhancing the presented services are the activities that form the complete lifecycle of optimization. Optimization is an iterative and recursive operation that shall take place all the time, in order to enhance systems.

Optimized Solution shall be evaluated from different perspectives. To evaluate the Adaptive e-Learning Models, we surveyed the two target categories of the models: Students and Instructors. Both of them showed interest in the presented Adaptive e-Learning Models and feel that it can enhance the e-Learning experience greatly. Students have issues with the repeated exams process, and grouping students in smaller groups. However, they liked the adaptivity features presented. Instructors suspected the applicability of intelligent cheat depressor service, however they still agree to use it as an indicator, and the final decision remains their decision of course. The second perspective to evaluate the optimized Adaptive e-Learning Models and Intelligent Services from is Information Retrieval (IR). Information retrieval measures of proposed Intelligent LOs Recommender shows an achievement in precision measure, with challenges at Recall and F-Measure due to the increased number of relevant learning objects as a result of including the course objectives at the crawling phase. That means, almost all of the Learning Objects stored in the database is already relevant. Future work includes expanding the proposed Intelligent LOs Recommender automatic annotation of media LOs.

Finally, optimized Adaptive e-Learning Models and Intelligent Services evaluated another service that is: Intelligent LOs Classifier. Presented Intelligent LOs Classifier uses Naïve-Bayes Classifier, and it showed 100% classification capability.

Chapter 7

Chapter Seven

CONCLUSION

E-Learning is an important part of the future however it is facing challenges. Optimizing e-Learning is a need. Optimization is a continuous and iterative process that goes through different activities. Addressing and understanding problems and challenges is the first activity. One of the e-Learning challenges is the absence of current e-Learning systems that adaptively and intelligently touch students' capabilities. Adaptive e-Learning is the solution to exploit unlimited e-Learning advantages. Adaptive e-Learning that is supported by intelligent methods and techniques is a need to move e-Learning to the efficiency and effectiveness. Services based e-Learning systems are the third generation that supports integration, interoperability, and scalability. Services based e-Learning Systems is a need to fulfill e-Learning systems pedagogical, technical, and architectural requirements. Adaptive e-Learning that is supported by intelligent techniques is the solution to present efficient and effective learning.

This dissertation presented Adaptive e-Learning Model that blends instructor lead education with e-Learning capabilities to provide an enhanced e-Learning environment as the solution to current e-Learning challenges. Presenting adaptive and intelligent features in the form of services with standard interfaces allow different e-Learning systems to adopt them, so they will be reusable, and the newly introduced information systems will not have to reinvent the wheel again. Besides, wrapping adaptive and intelligent features with standard interfaces will present a separation of interests that help adaptive and intelligent features' researchers and developers to focus more on their target, and transfer the responsibility of utilizing these features in different information systems to information systems specialists.

This dissertation presented different intelligent features to improve students' habits and their performance through e-Learning and to empower the presented adaptive e-Learning model. Services' specifications and IT-Infrastructure architecture required to enable the proposed Adaptive e-Learning Models were presented, highlighting the different challenges and their solutions. Intelligent services were categorized into two categories based on the user of those services: Instructor, and Student Services.

The Instructor Intelligent Services are: *Intelligent Learning Object (LO) Classifier service, Intelligent Online Lecture LOs Advisor, Intelligent Student Performance Tracker,* and *Intelligent Cheating Depressor*. The Student Intelligent Services are: *Intelligent Time-to-Learn Topic Calculation, Intelligent Study Plan Advisor, Intelligent Agenda Study Time Planner, and Intelligent Meeting Manager for Suspended Students* and *Intelligent LOs Recommender*.

Services based adaptive and intelligent e-Learning systems face performance challenges, mainly because they rely heavily on messaging. Dissertation presented Optimization techniques for the entire presented Adaptive e-Learning Model and intelligent services solely. Optimizing the entire SOA model through increasing granularity level, presenting SOA between systems not within systems, and utilizing new data protocols presented on the web to enhance performance is addressed. Optimizing performance of Intelligent Learning Objects Recommender as one of the presented intelligent services is another aspect that is presented as optimizing SOA on service level.

This dissertation presented an Intelligent Learning Objects Recommender that uses Fuzzy Logic to intelligently determine the membership of a Learning Object with Course Objectives. Components of the presented Intelligent LOs Recommender are: *Crawler*, *Intelligent Recommender, Data repositories* that holds Course Specifications, Student Learning Preferences, and Approved

Learning Objects Lists. Proposed Intelligent LOs Recommender faced some performance challenges in the first place due to the tremendous amount of generated keywords that affect processing and insertion times. Including Course Objectives as an input parameter in the keywords Term Frequency processing optimizes the performance by focusing on the important and needed keywords instead of wasting the processing time and storage spaces for non-important keywords.

Presented Model is evaluated from different perspectives: User Satisfaction, Performance, and Information Retrieval. Users of the presented Model are: Instructors and Students. Both categories accept the presented model in general with the needs to further consider some of the presented functionalities. Performance measurements was addressed on service level focusing on the Intelligent LOs Recommender as it has to deal with tremendous amounts of data, showing enhancements in Intelligent LOs Recommender performance measures after adopting the presented optimization techniques. Intelligent LOs Recommender performance increased almost 10 times and is capable of analyzing presented LOs, extracting the relevant keywords, and expanding the queries based on course specifications. Information Retrieval (IR) measures of proposed Intelligent LOs Recommender shows an achievement in precision measure, with challenges at Recall and F-measure due to the increased number of relevant LOs as a result of including the course objectives at the crawling phase. That means, almost all of the Learning Objects stored in the database is already relevant. Intelligent LOs Classifier is evaluated showing high accuracy levels in classifying test set.

Future work includes the focus on what e-Learning would look like in a Web 3.0 world, and how it might differ from current e-Learning. E-Learning 3.0

is the e-Learning empowered by Web 3.0 technologies. e-Learning 3.0 will have four key drivers: Distributed computing, Extended smart mobile technology, Collaborative intelligent filtering, and 3D visualization and interaction. e-Learning 3.0 will cross the boundaries of traditional learning institutions, and there will be an increase in self-organized learning. With cloud computing and increased reliability of data storage and retrieval, the mashup is a viable replacement for the portal which will lead to less reliance on centralized provision. Mobiles will play a big part in the e-Learning 3.0. There will need to be ubiquitous access to tools, services and learning resources, including people - peer learning group, subject specialists and expert support. Collaborative e-Learning will be possible in all contexts. e-Learning 3.0 will make collaborating across distance much easier. 3D visualization will become more readily available.

References

References

1. A. M. Riad, H. A. El-Ghareeb, “Empowering Adaptive Lectures Through Activation of Intelligent and Web 2.0 Technologies”, International Journal on E-Learning – Corporate, Government, Healthcare, & Higher Education (IJEL), ISSN: 1537-2456, Association for the Advancement of Computing in Education (AACE), Vol. 10, Issue 4, 2011.
2. Z. D. Baghdadi, “Best Practices in Online Education: Online Instructors, Courses, and administrators ”, Turkish Online Journal of Distance Education-TOJDE, Volume: 12 Number: 3 Article 4, July 2011.
3. S. Abramovich, C. Schunn, “Studying Teacher Selection of Resources in an Ultra-large Scale Interactive System: Does Metadata Guide the Way?”, Computers & Education, In Press, Available online 8 September 2011.
4. C. D. Dziuban, J. L. Hartman, T. B. Cavanagh, P. D. Moskal, “Blended Courses as Drivers of Institutional Transformation”, Blended Learning across Disciplines: Models for Implementation, IGI, 2011.
5. L. Wang, “Implementing and Promoting Blended Learning in Higher Education Institutions: Comparing Different Approaches”, Comparative Blended Learning Practices and Environment, IGI, 2010.
6. A. M. Riad, Hamdy K. El-Minir, Haitham A. El-Ghareeb, "Review of e-Learning Systems Convergence from Traditional Systems to Services based Adaptive and Intelligent Systems", Journal of Convergence Information Technology, Advanced Institute of Information Technology, Volume 4, Number 2, June 2009.
7. A. Bork, C. A. Bagley, X. Zhang, “Technology and Learning: A proposal for Adaptive Learning Tutorial”, Michael Allen's 2008 e-Learning Annual, J-B Pfeiffer Annual Looseleaf Vol1, 2008.
8. J. Ong, S. Ramachandram, “Intelligent Tutoring Systems: Using AI to improve Training Performance and ROI”, Stottler Henke Associates Inc., 2003.
9. D. G. Moursund,” Introduction to Information and Communication Technology in Education”, David Moursund, 2005.
10. F. Modritscher , V. M. Garcia-Barrios , C. Gütl, "The Past, the Present, and the Future of adaptive E-Learning", Proceedings of the International Conference on Interactive Computer Aided Learning (ICL2004), 2004.

11. N. Matar, S. Khwaldeh, Z.Hunaiti, “Adaptive Unified E-learning System for Supporting Better E-learning Approach”, The 8th Annual Postgraduate Symposium: The Convergence of Telecommunications, Networking and Broadcasting, Liverpool Johm Moores University, 2007.
12. P. Brusilovsky, "A Distributed Architecture for Adaptive and Intelligent Learning Management Systems", Proceedings of the AIED 2003. Workshop Towards Intelligent Learning Management Systems, Sydney 2003.
13. D. Burgos, C. Tattersall, R. Koper, “Representing adaptive and adaptable Units of Learning. How to model personalized eLearning in IMS Learning Design”, CELSTEC Research on Learning Networks, Open Universities, Netherland, 2006.
14. N. Stash, "Incorporating Cognitive/Learning Styles in a General-Purpose Adaptive Hypermedia Systems", SIKS, 2007.
15. C. Gutl, V. M. Barrios, F. Modritscher, “Adaptation in E-Learning Environments through the Service-Based Framework and its Application for AdeLE”, IDC: US Corporate and Government eLearning Forecase, 2003.
16. e-Learning Engineering, http://www.elearning-engineering.com/ai.htm, 2006, Last visited: 14 October, 2011.
17. D. Wen, F. Lin, "Ways and Means of Employing AI Technology in E-Learning Systems" icalt, Eighth IEEE International Conference on Advanced Learning Technologies, 2008, pp.1005-1006.
18. C. Chang, K. Hsiao, “A SOA-Based e-Learning System for Teaching Fundamental Information Management Courses”, Journal of Convergence Information Technology, Volume 6, Number 4. April 2011.
19. H. Barros, A. Silva, E. Costa, I. Bittencourt, O. Holanda, L. Sales, “Steps, techniques, and technologies for the development of intelligent applications based on Semantic Web Services: A case study in e-learning systems”, Engineering Applications of Artificial Intelligence, Elsevier, 2011.
20. R. Khalaf, T. Mikalsen, S. Tai, “Composition of Coordinated Web Services”, International Federation for Information Processing, 2011.
21. D. Rivas, D. Corchuelo, C. Figueroa, J. Corrales, R. Giugno, “Business Process Model Retrieval Based on Graph Indexing Method”, Business Process Management Workshop, 2010.

22. P. A. Castillo, J.L. Bernier, M.G. Arenas, J.J. Merelo, P. Garcia-Sanchez, “SOAP vs REST: Comparing a master-slave GA implementation”, Neural and Evolutionary Computing, Cornell University Library, 2011.
23. D. Agrawal, K. Candan, O. Po, A. Sawires, J. Tatemura, “A Scalable Middleware Architecture to Enable XML Caching for Web-Services”, International Federation for Information Processing, 2011.
24. A. M. Riad, Hazem M. El-Bakry, Haitham A. El-Ghareeb, "Mapping Different Software Architecture Paradigms to Different Integration Techniques: Highlighting Driving and Restraining Forces for Each Paradigm", Journal of Convergence Information Technology, Advanced Institute of Information Technology, Volume 4, Number 2, June 2009.
25. D. Dagger, A. O’Connor, S. Lawless, E. Walsh, V. Wade, “Service Oriented eLearning Platforms: From Monolithic Systems to Flexible Services”, IEEE INTERNET COMPUTING, 2007.
26. A. Al-Ajlan, H. Zedan, “E-Learning (MOODLE) Based on Service Oriented Architecture”, De Montfort University, ZDnet White Papers, 2008.
27. M. T. Wong, C. F. Soo, C. T. Ooi, S. L. Sow, “Service-Oriented E-Learning System”, Information Technologies and Applications in Education (ISITAE apos), 2007.
28. C. Chang, C. Yeh, Y. Yeh, “A Web-Service Oriented Framework for building SCORM Compataible Learning Management Systems”, International Conference on Information Technology: Coding and Computing (ITCC’04), 2004.
29. Z. Yin, A. El-Saddik, “A Web Services Oriented Framework for Dynamic E-Learning Systems”, Canadian Conference on Electrical and Computer Engineering; CCECE/CCGEI, 2003.
30. A. M. Riad, H. A. El-Ghareeb, “A Service Oriented Architecture to Integrate Web services and Agents in Course Management Systems”, Egyptian informatics Journal, Cairo University, 2007, Vol. 8, Issue 1, 2007.
31. M. Derntl, J. Mangler, “Web Services for Blended Learning Patterns”, Proceedings of the IEEE International Conference on Advanced Learning Technologies (ICALT’04), 2004.
32. A. Harrer, N. Pinkwart, B. M. McLaren, O. Scheuer, “The Scalable Adapter Design Pattern: Enabling Interoperability between Educational Software Tools”, IEEE Transactions on Learning Technologies, 2008.

33. A. M. Riad, H. A. El-Ghareeb,"A Service Oriented Architecture to Integrate Mobile Assessment in Learning Management Systems", Turkish Online Journal of Distance Education – TOJDE, Volume: 9 Number: 2, Article 12, April 2008.
34. P. J. Kloos, J. Naranjo, "Enabling Interoperability for LMS educational Services", Computer Standards & Interfaces, El-Sevier, 2009.
35. T. K. Apostolopulos, A. Kefala, "An XML-based E-learning Service Management Framework", Proceedings of the IEEE International Conference on Advanced Learning Technologies (ICALT'04), 2004.
36. W. Chen, H. Su, Y. Chu, K.Chen, "Design of Knowledge Management Learning System based on Service Oriented Architecture", Proceedings of the Sixth International Conference on Advanced Learning Technologies (ICALT'06), 2006.