AN INTELLIGENT AND PERSONALIZED COURSE ADVISORY MODEL FOR HIGHER EDUCATIONAL INSTITUTES

A THESIS SUBMITTED TO THE GRADUATE SCHOOL OF APPLIED SCIENCES OF NEAR EAST UNIVERSITY

By
LEUL AYNEKULU TILAHUN

In Partial Fulfillment of the Requirements for the Degree of Master Science in Software Engineering

NICOSIA, 2019

LEUL AYNEKULU TILAHUN

AN INTELLIGENT AND PERSONALIZED COURSE ADVISORY MODEL FOR HIGHER EDUCATIONAL INSTITUTES

NEU 2019

AN INTELLIGENT AND PERSONALIZED COURSE ADVISORY MODEL FOR HIGHER EDUCATIONAL INSTITUTES

A THESIS SUBMITTED TO THE GRADUATE SCHOOL OF APPLIED SCIENCES OF NEAR EAST UNIVERSITY

By
LEUL AYNEKULU TILAHUN

In Partial Fulfillment of the Requirements for the Degree of Master Science in Software Engineering

NICOSIA, 2019

Leul Aynekulu TILAHUN: AN INTELLIGENT AND PERSONALIZED COURSE ADVISORY MODEL FOR HIGHER EDUCATIONAL INSTITUTES

Approval of Director of Graduate School of Applied Sciences

Prof. Dr. Nadire ÇAVUŞ

We certify that thesis is satisfactory for the award of the degree of Master of Science in Software Engineering

Examining Committee in Charge:

Prof. Dr. Rahib H. ABİYEV	Department Head, Computer Engineering, NEU
Assoc. Prof. Dr. Kamil DİMİLİLER	Department Head, Automotive Engineering, NEU
Assist. Prof. Dr. Boran ŞEKEROĞLU	Supervisor, Department of Information Systems Engineering, NEU

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Leul Aynekulu, TILAHUN

Signature:

Date: July 22, 2019

ACKNOWLEDGMENTS

This Master's thesis, An Intelligent and Personalized Course Advising Model (IPCAM) for HEIs is the concluding piece of my two-year Master's degree of Software Engineering in Near East University (NEU). For other researchers interested in the field of Academic Advising, I believe my work could be a good summary of the state-of the-art research results.

The thesis could not be done without the help of many dedicated people. I have a deepest gratitude to my advisor and supervisor, Asst. Prof. Dr. Boran ŞEKEROĞLU for his continuous support commenced from the beginning of the first semester up to the final submission of this paper. He also gives me ambitious guide in this current trend and admirable research area. I also kindly acknowledge NEU Software Engineering department head, Assoc. Prof. Dr. Yöney Kırsal EVER for her support regarding research area. The most important is their precious instruction at every step during my thesis. Their commentaries also help me to accomplish this thesis and for my future academic life. I would like to thank Near East Technology Ltd, software development manager Mr. Raif USAL and Mr. Ibrahim BAY for their continuous support, encouragement, and suggestions all the time.

I have also deepest gratitude for software engineering, computer science and information system students who participate in evaluating IPCAM that applied in NEU. I would like to give special thanks to Wollo University (Kombolcha Institute of Technology), NEU and Ethiopian Ministry of Science and Higher Education for generously offering me for the opportunity and scholarship to study in TRNC. This two-year international experience would definitely change my future.

At finally yet importantly, I would also like to thank my father, mother, brothers, and my best friends for their motivation, support, and patience to finalizing this thesis within a limited period.

To St. George and My Brothers…

ABSTRACT

An Intelligence and Personalized Course Advising Model (IPCAM) plays an important role in students' academic success throughout their educational life. During the course registration period of every academic semester, this model investigates student's interest, course history, rules and regulations, experts rating on the course-carrier relationship in order to recommend suitable courses. This paper improves the current trend of academic advising by analyzing, selecting and modifying effective academic advising techniques to design IPCAM for all HEIs. It also explicitly explains the detail design and implementation of the model by collecting, analyzing and discussing seven universities' system and user requirements. This model consists of seven small sub components, in which they used appropriate algorithms. Association rule, rule based expert system and simple recommendation system have been selected and modified for the development of the model based on their suitability for collected requirements. IPCAM is highly suited to every student information system (SIS) specifically to course registration module. It makes the advisor and advisee to use their knowledge effectively and efficiently by delivering the best solution that saves their time, money and other resources. In addition to this, it is very crucial for software developers, universities and developing countries who have insufficient educate manpower. IPCAM has evaluated by ISO-25010 quality model using the NEU rule and regulation with three departments. As a result, it is more effective and efficient model to provide the desired solution.

Keywords: Academic advice; IPCAM; ISO-25010 software quality model; course registration; intelligence; personalized; course analysis

ÖZET

Bir Zeka ve Kişiselleştirilmiş Kurs Tavsiye Modeli (IPCAM), öğrencilerin eğitsel yaşamları boyunca akademik başarılarında önemli bir rol oynamaktadır. Her akademik dönemin ders kayıt döneminde, bu model uygun dersler önermek için öğrencinin ilgisini, ders tarihini, kuralları ve düzenlemeleri, derste uzman değerlendirmesi ve taşıyıcı ilişkilerini inceler. Bu makale, tüm HEI'ler için IPCAM tasarlamak için etkili akademik danışmanlık tekniklerini analiz ederek, seçerek ve değiştirerek akademik danışmanlığın mevcut eğilimini geliştirmektedir. Ayrıca, yedi üniversitenin sistemi ve kullanıcı gereksinimlerini toplayarak, analiz ederek ve tartışarak modelin detay tasarımını ve uygulamasını açıkça açıklar. Bu model, uygun algoritmalar kullandıkları yedi küçük alt-bileşenden oluşmaktadır. Toplanan gereksinimlere uygunluklarına dayanarak modelin geliştirilmesi için dernek kuralı, kurala dayalı uzman sistemi ve basit öneri sistemi seçildi ve değiştirildi. IPCAM, her öğrenci bilgi sistemine (SIS) özellikle ders kayıt modülüne çok uygundur. Zamanını, parasını ve diğer kaynaklarını koruyan en iyi çözümü sunarak, danışmanını ve bilgilerini etkin ve verimli bir şekilde kullanmasını tavsiye eder. Buna ek olarak, insan gücü konusunda yeterli eğitime sahip olmayan yazılım geliştiriciler, üniversiteler ve gelişmekte olan ülkeler için çok önemlidir. IPCAM, NEU kuralı ve üç bölümlü yönetmelik kullanılarak ISO-25010 kalite modeliyle değerlendirilmiştir. Sonuç olarak, istenen çözümü sağlamak daha etkili ve verimli bir modeldir.

Anahtar Kelimeler: Akademik tavsiye; IPCAM; ISO-25010 yazılım kalitesi modeli; Kurs kaydı; zeka; kişiselleştirilmiş; ders analizi

TABLE OF CONTENTS

CHAPTER 4: ADVISORY SYSTEM DEVELOPMENT TECHNIQUES

CHAPTER 5: INTELLIGENT AND PERSONALIZED COURSE ADVISORY MODEL DEVELOPMENT

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AI:	Artificial Intelligent
ACAS	Automated Course Advising System
CSV:	Comma Separated Values
CGPA	Cumulative Grade Point average
CLIPS	C Language Integrated Production System
GPA	Grade Point average
GUI:	Graphical User Interface
HEI:	Higher Educational Institute
ID	Identification
IDF	Inverse Document Frequency
IPCAM:	Intelligence and Personalized Course Advising Model
ISO:	International Standard Organization
ICSE	Indian Certificate of Secondary Education
JESS:	Java Expert System Shell
KNN:	K-Nearest Neighbors
MCR:	Multivariate Curve Resolution
ML:	Machine Learning
MOOCs:	Massive Open Online Courses
NACADA:	National Academic Advising Association
NEU	Near East University
OPA:	Oracle Policy Automation
PAS	Process Automation System
SAES	Student Advising Expert System
TF	Term Frequency
WEKA	Waikato Environment for Knowledge Analysis

CHAPTER 1
INTRODUCTION

1.1 Background

Nowadays, every country invests on education to enhance knowledge, understanding, skills, values, and actions of the society. It is also a powerful weapon for the development of social, cultural, economic and political activities, which ensures environmental safety and conservation, promotes social fairness and encourages monetary sustainability. Education allows people to take their choice and carry out actions to enhance their life without compromising the planet. It also pursuits to combine the values inherent in sustainable improvement into all aspects and stages of learning. Hence, education is one the most pillar activity for sustainable development of the given country. Academic staffs play the main role in educational institutes by providing scientific service in human resource development, which mainly concerns on giving community service and solve social, political and economic issues of the society through research and transferring technology in addition to learning and teaching process.

Students need advice on different academic issues especially during registration period. they are unable to make a proper decision, so those advisors are taking additional responsibilities to advise, motivate and support each student in individual or in a group. Academic Advising should multi-dimensional, intentional and grounded in the education system, which has material in order to make specified outcomes for student learning. It has a lion share in helping decision-making in unstructured and more complex situations, which makes ambiguity to give one clear answer (Beemer & Gregg, 2008). When students face some difficulties, they will tell the issues to their advisor then, the advisor start giving advice by analyzing the situation according to his/her knowledge, skill and experience to support and solve the issue.

According to National Academic Advising Association [NACADA] (2018), Academic advisory system is a set of activities for the dissemination of knowledge that guides students for the career goals and development of academic plans. Academic advising quality is very important to improve students' academic performance. Good academic advice creates a desired and better result while poor advice is disappointing and has a detrimental impact on the education system and wisdom development of students (Daramola et al., 2014). The complete academic advising includes train students in order to know how to achieve their goal and provide a simple and manageable way to solve different academic, social or personal problems that happened in academic life. It realizes their academic potential by counseling, suggesting, and, recommending their students.

According to Abdelhamid et al. (2015), Academic advising become very difficult, complex and time-consuming task especially, when students number are increased and advisor to student ratio reduced. This indicates the number of advising students is directly proportional to the strenuous of academic advising and indirectly proportional to academic advisors.

The emergence of computerized course advisory system helps to reduce the responsibility of academic advisors through assisting both the advisees (students) and academic advisors. Nevertheless, the problem-solving mechanism are different from system to system, which designed under strong consideration of user and system requirements of particular HEI.

This paper primarily focuses on the exploration of better problem-solving mechanisms for course advising in order to build flexible, compatible, efficient and high-performance course advisory system for higher education institutes as framework.

1.2 Historical Overview

According to NACADA (2018), Academic advising has main roles in education sector development. Although scholarly advice has been a clear-cut region within education for only a few centuries, since the evolving American college organizations it has been a predominant problem. The advancement and evolution of academic advising have a crucial advantage in providing the best practices. Student advising and consulting had commenced in a certain form since the end 18^{th} century. During this period, America's first colleges were organized from Cambridge and Oxford's English blueprint to civilize their youthful people (Rudolph, 1990). Everybody who works in the education sector was concerned with the moral and intellectual development of students (Gallagher & Demos, 1983). The fast development of institutions in the nineteenth century given a period for academic advice to safeguard their position in education and the emergence of advisory organizations (Gordon, 1992). Progressive Education Movement of the 1920s concentrated on the student's self-direction, emphasizing the task of teacher as ' mentors ' who added their ability to the student's advancement. The first university colleges were set up during 1930 in which they completely hold advising and about 49% consulting centers. The American Education Council released the Student Personnel perspective in 1937 that centers on private concerns and the concept of holistic education (Strange, 1994). Many child boomers on colleges and universities created enhanced demand the guidance and advising for students after and during World War II. Problems of student development in the educational region have been investigated in educational area. (Gordon, 1992).

The development of students through academic advising program is a driving force to use specific academic advising approaches, which guide students towards self-discovery, set life goals, and recognize academic challenges(Harris, 2018). Noaman & Fouad (2015) state that academic advising has variety of approaches depends on advisor-advisee relationship. However, there are three basic academic advising models. The first one is developmental advising, which mainly focused on supporting students, search and define educational carrier, life goal and build problem-solving skill. It used process-oriented and collaborative advising. In this approach, students are beneficial but it needs a better commitment to time and resource (Harris, 2018). The second one is prescriptive advising approach, which is also known as traditional advising model. It mainly targets in providing academic program and its progress information including academic policies, major program requirements, and course selection processes. This approach has not aimed to help students from long-term goal above their academic goals rather it solve immediate questions to guide students in their academic life. (Harris, 2018). The last one is intrusive Advising approach, which is also called proactive advising. An adviser initiates the interaction in order to assist pupils at critical phases throughout the educational life of a student. This model helps to give more information to students before they requested it and helps to build a relationship with the student simultaneously. In this approach, specific cohorts of students focused including high achieving and low achieving students. In practice, intrusive advising has a great impact on the development or remedial course and students' retention and degree attainment rates (Varney, 2012).

Currently, more HEIs are using technology to manage students' information, control distance and continuous education, and needs to improve their course-advising program. Therefore, they are always looking for an efficient application, which aims for reducing routine, time consuming and tedious tasks.

1.3 Problem Statement

The existing course advising is a crucial activity for students to obtain the most relevant information to make the appropriate decision. Nevertheless, it is time-consuming activity for students and advisors. As the academic advisor, they have to investigate the individual cases of students to determine the available opportunities for each student to accomplish his/her academic degree as fast as possible. Existing system problems, which are the driving force to do this thesis, are explain as follow: -

- If an advisor assigned for more students in order to provide course advice service, students should meet in scheduled time through in person or using communication medias depends on institute's rules and regulations. Hence, it directly causes wasting of money and time; it also reduces the number of research and community service lead by academic staffs or advisors. Advisors spend more time and effort during course advising, which directly affects research, community service, and the normal teaching-learning process. Advisors' payment as salary and different material including internet usage and supportive materials what they used during the discussion is also expensive.
- Advisors have limited knowledge about course-profession relationship, so it is difficult to suggest course according to student's special interest.
- Individual student has varied academic records, so it is difficult to analyses it for providing accurate course suggestion for the current academic semester.
- Advising method and advisors knowledge about courses are different from advisor to advisor, so many students blame on their advisors by comparing their advisor to others.
- Often advisors could not remember all institute's academic rules and regulations to apply on a particular student's case.

- Advisors and students feel tired to explore complete and efficient information about the courses and get difficulties to make a decision in order to choose appropriate courses.
- Unusual egocentrism puts advisor in a negative psychological complex, so when he/she advise, it is easily attack students.
- Sometimes students or advisors could not briefly understand their discussion after wasting a lot of time due to communication problem.
- Often, most students difficult to get more help due to advisors scheduled time means they may not be available on time.
- Students are scared to tell their confusion and to ask the same question several times to their course advisors.
- Most governmental HEIs have the quite similar rule and regulations within one country but they government fund for system development individually.
- Mostly SIS development team has difficulty when they attempt to customize or integrate course advisory system in accordance with particular HEI's requirements. In the development of SIS with course advising service, it needs some effort in both designing from scratch and customize other institute's course advising system such as incompatibility issue and lack knowledge to all programming language and design paradigm.

1.4 Objectives of the Study

The objective of this thesis articulated as general and specific objectives. General objective specifies what we strive to accomplish in summarized way. Specific goals indicate what we are planning to do in every stages of our research and give clarified specific operations to attain the overall goal of the research.

1.4.1 General objective

The ultimate goal of this paper is designing IPCAM for HEIs and evaluating the model using ISO-25010 quality model.

1.4.2 Specific objectives

In this subsection, we are going to list out different tasks as specific objectives that helps the achievement of the main aim of this paper.

- Explicitly illustrate about academic advising.
- Discuss with related works in order to discuss about various course advising system development techniques in various HEIs.
- Analyze rule and regulations of course advising system of different HEIs to construct a common attribute.
- Analyze requirements of course advisory model.
- Discuss different problem-solving algorithms.
- Choose better course advisor algorithm in accordance with HEIs common course advisory system requirements.
- Design compatible, IPCAM for HEIs including explicitly describe components of a model.
- Implement IPCAM for experimentation, in accordance with NEU academic rules and regulations to evaluate the model by using ISO-25010 software quality model**.**

1.5 Significance of the Study

This thesis has different advantages for different stockholders in various perspectives.

i. ***Universities*****:** This paper supports to know their student's area of interest and helps to use harmonized knowledge, which will be gathered and analyzed from many experienced academicians as experts. It helps to reduce communication between advisor and student so they will mainly focus on research and innovative ideas to solve social, economic and political problems. This is one of the major factors to increase universities qualification and rank. It also helps to reduce the cost of advising that budgets to advisors.

ii. ***Advisors*****:** In traditional way of academic advising, many academicians spend their time on advising communication with their students to solve different many academic issues. Even though they collect small amount money, the time of academician that spend for advising could not compared with the advising cost so advisors time more important than advising income. This paper helps advisors to focus on best innovative business ideas as alternative income. They also give special attention and time to teaching learning process, research development and providing community service, this leads to develop the university and the country.

iii. ***Students*:** It gives the way to support student's decision-making process in course registration by suggesting appropriate course depend on student's previous academic history, HEI's rule and regulation, course nature and area of interest. It saves the time, money and effort that spend for exploring different information about courses or communicating with their personal advisor. They do not have direct communication with their advisory, which leads to prevent corruption, sexual harassment, egocentric attack and other academic misbehaviors. Students do not blame advisors about advising performance by comparing with other's advisor. At last but not list, Students also do not afraid to ask the system many times to get appropriate course suggestion.

iv. ***Software developers and researchers:*** This paper puts a roadmap and future direction in course advisory model, which helps researchers to know, discuss, extend and add their contribution. It exponentially reduces the effort of developer that spends for customization and integration of the model with SIS in accordance with particular HEI's requirements. It also enables the developers to easily maintain, integrate and upgrade in a compatible and reusable manner. This will happen without compromising the design paradigm and programming language it uses.

v. ***Country*:** Even though many countries especially developing countries have the deficiency of academician, they have the responsibilities to train their citizen in different areas. More often academic advising is tedious work, especially when the number of advisee or students is more and academic advisors is small. This forces the department coordinator to assign many students to one advisor, which leads direct and indirect impact on a quality of education and the development of a country. If students can get specific professional skill, the advising load of advisors are reduced. Therefore, Advisors will give more focus on the research development and community service. Those all lead to develop a country from various perspectives. Most governmental HEIs of a country have a very similar rule and regulations, but they are always funding for SIS individual institutes. Therefore, this software solution brings all HEIs to share course advisory model in comparatively low cost. It also helps to give attention to education quality through the efficient use of the educated human resource.

1.6 Scope of the Study

This paper has a boundary for its contribution area, which states up to what boundary we are going to focus and explains a frame for included and excluded features. It explained as follow: -

- It mainly focuses on course advising model in order to suggest appropriate courses based on student's academic history, special of interest and HEI's specific rules and regulations. It consists analysis of selective algorithm to use more efficient and appropriate algorithm for IPCAM design. It also consists of analysis, architectural and detail component level design of the model including intended algorithms, which enables to guide student's decision-making process in course registration period.
- The model can be in use for every HEIs as per condition of their rules and regulation as well as department's curriculum. It will specify the way to update knowledge from database and academicians for efficient course advising but not develop the interface. The model will also use student's academic data as input from HEI's SIS to intelligently analyze and suggest appropriate courses in customized manner as output in SIS course registration student's port.
- An IPCAM enables to provide course suggestion after a serious of automated process from filtering information up to delivering semester wise appropriate courses for students pursuant to rules and regulations of particular HEI without attentive participation of advisors and students. It also automatically suggests elective courses through intelligent analysis of the academician course-profession rating dataset and calculate the rating for the best fit in accordance with student's special interest.
- Courses that suggest for all students are quite different unless they have the similar academic history, special interest and ruling under the same department and HEI rules and regulations.

1.7 Limitations of the Study

In the evaluation phase of a model, it is difficult to access original students' academic data so it wastes a lot of time and effort during the construction of a dataset from scratch rather than automatically generating or exporting from SIS database as designed in the model development phase. Due to information privacy, NEU could not allow accessing Einstein system to integrate with the model. Therefore, desired information of Einstein system shall substitute by accepting from their stockholders. time and data privacy limitation are also the main bottlenecks to select more HEIs for further evaluation.

1.8 Thesis outline

This thesis suggests an IPCAM for HEIs that takes into student's special interest of field and academic history, institute's rules and regulations. The next chapter presents different works that related to this area with their main findings. Chapter 3 states about the methodology in which the research follow to attain the required goal. In chapter 4, this paper briefly explains different algorithms that used for the development of academic advising in different point of view. This helps to select better algorithm for the development of quality IPCAM. Chapter 5 explains the development of IPCAM through analysis, architectural and component level design. Chapter 6 presents about sample implementation of a model in NEU for evaluation and finally states the result by comparing to the existing system. Finally, Chapter 7 summarizes the main finding of this paper and provides suggestion for the further improvement of IPCAM.

CHAPTER 2
LITRATURE REVIEW

2.1 Introduction

There are many research papers those provided relevant contribution in electronic advising system, but we review the most vital works to my research area in order to demonstrate understanding, identify related gaps in the literature, set a theoretical framework, develop a methodology and to support my finding. Finally, it has summarized in the next topic.

2.2 Related Works

Dash & Vaidhehi (2017) propose an advising system for the school students of eighth standard of Indian Certificate of Secondary Education (ICSE) board to choose their electives using machine-learning algorithms in order to assist human advisor efficiently by using WEKA tool. This tool is a collection of machine learning algorithms for data mining tasks. It has tools for data preparation, classification, regression, clustering, association rules mining, and visualization. It assisted Indian school students in analyzing their academic history and help them to choose their electives wisely.

Huang et al. (2016) proposed a course recommendation system for MOOCs platform that contribute an improvement of course recommendation model and algorithm based on distributed computational framework. MCR used distributed association rule mining algorithms in order to improve priori algorithm that is useful to mine the hidden course rule in course registration. In this paper, researchers did experiment and in Hadoop and spark then they concluded, "MCRs is the most efficient than a prior algorithm."

Those researchers have modeled and tested Naïve Bayes, J48, PART, Random Forest and IBK on different data set. As the result, the accuracy of classification algorithm random forest has increased from 99.07 % to 99.69 %. In case of PART, the accuracy has increased from 96.29 % to 96.91 %. IBK also shows an increase in efficiency from 99.07 % to 99.69 %. Generally, it can be deduced that the classification algorithms, Random Forest and IBK are equally suitable for current data set. They express graphically as follow.

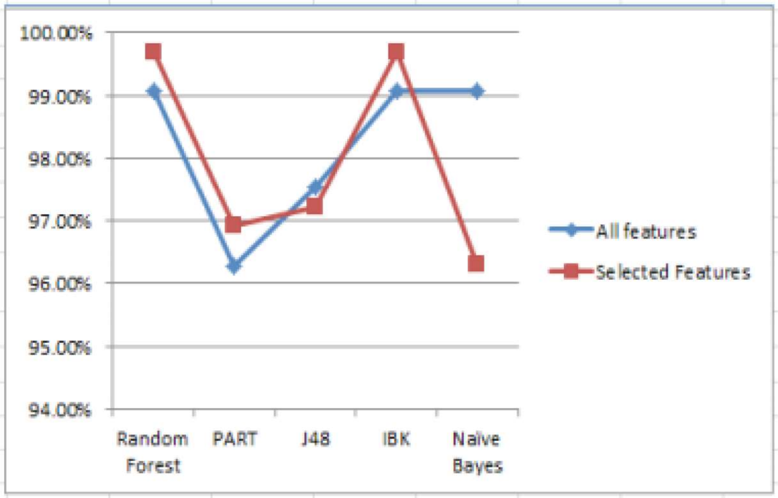


Figure 2.1: The comparison of all classifications' performance of using feature extraction methods (Dash & Vaidhehi, 2017)

Abdelhamid et al. (2015) propose a multi-agent technology for course recommendation and timetable scheduling. The academic advisor agent imitates the course-planning task of the expert academic advisor. It uses case data of each students consisting transcript, which is the academic report about students' academic level, passed subjects, grades and also credit hour, attendance hours of each subject and prerequisites. As the result, the academic advisor recommends course to add into a student's schedule. Researchers also focus on solving the contradiction of timetable by making it automatically modifiable based on courses that students registered. They describe four methods to solve timetable contradiction such as sequential, constraint-based, cluster-based and meta-heuristic.

Henderson et al. (2015) stated that the automated web based advising application system for science and technology faculty in West Indies University to handle general advisory cases and basic student issues that are most related to course advising, information about graduation status and exam qualifications. Researchers had presented the inference system via rest full java web server.

Laghari et al (2015) developed a student course planning software package for electrical students that used for students in order to guide course selection process during the registration period. This software guides students in selecting the most relevant and suitable six courses to register in the next semester. Prioritization occurs on all six fields in the top-down approach, which means the highest priority field comes first. The output of this software can be used as input for the university registration system. The researchers use a most powerful programming language (python) for the development of this student course planning software.

Laghari (2014) present a system that guide United Arab Emirate university students in order to choose their semester courses using java programming language. The outcome of Automated course advising system (ACAS) was storing semester wise selected courses to show a complete typical plan.

Daramola et al. (2014) proposed an intelligent course advisory expert system that designed and implemented by using rule-based reasoning and case-based reasoning in order to recommend courses during course registration for specific semester based on the academic history of all students. Those researchers also evaluate system's performance in terms of reliability, usability and credibility.

Shatnawi et al. (2014) stated that application of association rule data mining for selecting and prioritizing courses by generating a list of combination between different previously registered courses in order to improve students' performance. This system suggested and recommended courses that fit to students' status.

Engin et al. (2014) has analyzed, reported and discussed about the design, implementation and testing of course advising academic expert system and scholarship suggestion system for undergraduate students for international university. Course advising system provide course recommendations and scholarship recommendation system suggest scholar ship depends on their capability and performance. Those systems implemented and tested via Oracle Policy Automation (OPA) software.

Hingorani et al. (2014) Propose that the design of advisory system for southeastern university in order to help pretension and graduation. Those researchers had developed this system to satisfy students and advisors in the university through their direct involvement. This system developed using ASP.Net and Access database.

The table structure is suited with PeopleSoft system, which is easily migrate to live Oracle tables.

Al-Nory (2012) designed and implemented spreadsheet based academic advising system using visual basic script and Microsoft excel tools in order to automate repetitive advising activity held by advisors and their students but it was not integrated with SIS. In this system four-year schedule of each batch and translation of student transcript in the form excel documents are required from department.

Al-Ghamdi et al. (2012) designed and implemented postgraduate student advising expert system to assist and select different suitable computer science courses that provided in King Abdulaziz University by considering prerequisite and department's maximum requirements. It also provided an opportunity to select and get plan for each semester without direct consulting of advisors. This system uses Process Automation System (PAS) model, which supports postgraduate students in choosing the suitable subjects for them to solve related problems and conflicts. PAS used to reduce the burden on computer science department to offer sufficient number of advisors for all students. In addition, this it saves time for taking appointment which enables to make an appointment as fast as possible.

Ishak & Lehat (2012) states about conceptual framework of academic advising and the elicitation of requirements for existed academic advising system that had conducted through literature survey. Researchers conducted random survey had conducted to get different basic information form students and advisors' perspective then they were mainly investigated the problem in academy advising to propose conceptual framework for web based academic advising information system.

Feghali et al. (2011) developed and evaluated a web-based decision support tool in order to solve the technology-based problem in the SIS. This system has mainly used for advisors and students to guide their decisions while they have used the university SIS for better and more useful. The researchers conclude, "Online Advisor is as good as the 'freshness' of its information. Therefore, it is critical that all information is updated promptly to secure higher adoption on campus."

Al Ahmar (2011) states the development of student advising expert system model to support student's decision in selection of course for each academic semester and to help advisors on academic planning. Those researchers use prescriptive and developmental model advising models; object-oriented database paradigm and interactive user interface.

Hwang (2011) states knowledge base development of an expert system in order to assist and support students. This system was developed by analyzing the web-based problem-solving ability. Those researchers had evaluated qualitative and quantitative attributes of the system through interviewing two domain experts and conducting an experiment so, as the result show, it provided constructive and accurate suggestions to many different students.

Aslam & Khan (2011) states the design of a model for testing, measuring students' capabilities and applying the module results to the proposed system. Researchers have developed their rule base decision support system using CLIP language, which provides appropriate support for rule-based object-oriented and procedural programming. This proposed system designed and developed to support faculty selection while students taking admission in Gomal university of Pakistan. Finally, the researchers had result that suggested faculties/majors' courses form curriculum.

Nambiar & Dutta (2010) presented an expert system that developed by using Java Expert System Shell (JESS) in order to provide active and fast response to student's requests that mostly related to their department and study plan. Researchers implement this system through rules and its actual code separately to make easy for customization and maintenance by editing the rules that stored as xml file.

Deorah et al. (2010) presents Student Advising Expert System (SAES), which is an expert system that aims to help students by indicating the major courses. This system acquires knowledge of academic performance that represented using case based and rule base reasoning. Its inference developed based on those acquired knowledge, which save time and remove the stress that faced while selecting the interest area. SAES provides advice the most preferable courses for each advisee by categorizing it as strong, medium and weak based on relative probabilities of success. Those researchers conducted tests to evaluate the working model.

CHAPTER 3
METHODOLOGY

3.1 Data Collection

The detail information about course advising systems, the current trends of academic advising, different development algorithms, rule and regulation of HEIs have conducted through the following ways.

- Direct discussion with Near East Technology Ltd. Software development team has held. Then, we got more detail information about courses and students of NEU. Discussion with advisors, faculty members, administration, and students to decide on the data that needed to be used during model development. The discussion with nine students and two professors has used to collect variety user requirements.
- Via direct observation and reviewing different primary and secondary sources including academic records, curriculums and related documents about randomly selected seven universities stated on Section 5.2 has analyzed and discussed. Those universities are NEU, Wollo university, Cyprus International University, University of Kyrenia. Addis Ababa University, Gomel University of Pakistan, Black University of South Carolina, King Abdulaziz University of Jeddeh, American University of Beirut.

3.2 Data Preprocessing

In this subsection, we are going to discuss about the methods that supported for investigation and illustration of various academic advisory data, processing, analyzing data for clarification and explanations. Through it, alternative solutions also analyzed with its feasibility analysis. All collected data that explained in section 3.1 analyzed and represented in comfortable way to select appropriate development algorithm and to design the model in reusable and compatible manner. IPCAM is used one of the core components HEI's SIS course registration module. HEIs uses SIS to store and track all student information, including grades, attendance records, course registration and, it provides relevant information pertaining to a student's schooling. The database that used in this system different through university to university. All basic and detail information of courses, students and instructors should be collected and represented in order to provide automatic and an intelligent course advising services to each and every student based on the grading system and rule and regulation of the university.

Basic HEIs' data collected and represented in a formal hierarchical structure to design a common data attribute framework. We analyze system and user requirements using use case diagram, flowchart and activity diagram (Alhir, 2003) to select most suitable development algorithm, modify in accordance with the requirement or develop a new algorithm, which is suit for all requirements. we selected association data mining tool, rule based expert system and model-based recommendation system and highly modified in order to fit with the stated requirements.

3.3 Model Development Methods

As we discussed in Section 1.4, the fundamental advantage of IPCAM is saving advising time, effort, and cost of course advising. It also provides quality data for other components or software applications in the same area. Due to this, it will exponentially reduce the development and maintenance cost. It also reduces advisor involvement to interact with the system and accept academic experts' knowledge to provide better and complete course suggestion. It is highly compatible with almost all SIS system databases because it has capable of using data without directly attached to them rather it uses intermediate data structured and analysis format type. To keep the advantages, we conducted design methods.

Architectural diagram used to describe the interaction of SIS authentication, course registration, persistence database and transitional datasets with IPCAM. The detailed design explained by component design (Alhir, 2003) in which it explains the interaction between course priority configuration, taken course analysis, prerequisite course analysis, elective course analysis, course suggestion, personalization and academic term analysis subcomponents. Each subcomponent expressed through the effective algorithm.

3.4 Evaluation Methods

The experiment in this thesis was conducted in three departments of NEU. Those are Software Engineering, Information System and Computer Science. From those departments, nine students were selected as evaluators. Each student test the model by using five test cases, which have unique student ID. Then, a set of questionnaires based on the quality characteristics described in Section 6.1 was distribute to those evaluators. The questionnaires have questions regarding the use of IPCAM in NEU and quality as perceived by the users. The evaluation proceed depends on the scale of Very high, High, Moderate, Low and Very low for the selected quality characteristics in the form of Appendix 1. Then the result of individual characteristics could be calculated and converted to percentile.

CHAPTER 4
ADVISORY SYSTEM DEVELOPMENT TECHNIQUES

4.1 Recommendation Techniques

Research on recommendation systems has been going on both on academic and industry for almost twenty-five years. Recommendation systems development varies from domain to domain and the type data to work on. For instance, five-star is used in Amazon, like/dislike in Facebook and soon. Which means the user feedback is recorded into a data source in such a way. The data filtering process targeting at exploring the matching pairs also differs.

Recommender systems have advantages for students by delivering course suggestions during the beginning of every academic semester. Students (advisee) and course (items) are basic building of academic recommender system. Advisee is an individual that uses this system and giving his or her experiences about many courses and get a suggestion about opened courses back.

The ultimate aim of an academic recommender systems is generating suggestions about the next semester courses for a particular student. An advising process depends on the input provided, which related to the preferences of that user.

Recommender system's input relies on the sort of the filtering algorithm used. In general, one of the following categories refers to the input:

- Votes (Rating) stating students' views on courses. Usually, students give rating that obey rating scale rules (1 for worst to 5 for best). A popular rating system enables only either null or one vote only. Student's academic history can also give ratings for specific courses.
- Demographic data, which refer to students, courses, lecturers detail information such as the ID, name, gender, course code, credit hour or ECTS etc. Usually this type of data is hard to obtain. Therefore, SIS can explicitly provide it.

- The contents of data that relies on a semantic assessment of student relation with course. The further exploration algorithm uses filtered characteristics to create a student profile.

This system's outcome can be either recommendation or forecasting.

- A forecast is express as a numerical value, Si and j representing students Si's expected view of Cj course. This expected valuation should be within the same scale (for instance 1-worst to 5- best) as students rating. It is what we call it "Individual Scoring"
- A suggestion expressed as a roster of C courses where C $\leq$ n expected to be the most appreciated by active pupils. In that case, usual approach needs this list to include only perceived or rated courses. It is also known as "Top-N recommendation or ranked scoring".

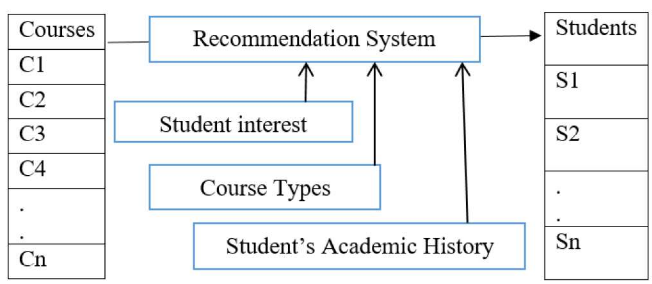


Figure 4.1: Overview of recommendation technique for course advising

There are three most common types of recommendation systems; those are collaborative filtering, content-based, and hybrid filtering, based on the information utilized to provide the suggestions.

4.1.1 Collaborative filtering recommendation system

Collaborative Filtering uses historical records over a set of products as it operates depends on existing information. The main hypothesis of this system is in the future, the user who have collaborated in the past tend to agree. Two classifications generally express user preference, namely explicit and implicit rating. Explicit scoring is a process of providing value to an item on a continuum scale by a student, for instance four stars for introduction to programming course. It is students' most recent feedback to demonstrate the level their interest to a given course. Implicit Rating indirectly suggests student's preference without offering their rates immediately, but prior documents or website opinions. (Luo, 2018). We use this filtering system, if a group of customers who acts in a similar manner and have similar requirements and needs. For example, if two students have an interest in a given course, and give similar rating score to it. If two comparable students registered, a class not known to the other customer, this assertion claims that the system can safely advice this course to the pupil, as a comparable student prefer to get in to it and value it as well.

They highly probable to share the enrollment of other courses that might suggest it. If one of two similar students enrolled, a particular course that the other user is not aware of, this premise says that the system can confidently recommend this item to the student, because similar students tend to like and rate similarly (Levinas, 2014). There are two methods; these are memory and model-based techniques.

4.1.1.1 Memory-based technique

It employs a complete user-item database loaded in the memory to create a prediction. It calculates memory similarity between neighbors. Here, various entries of the database are inserted in to memory and used immediately to advice the user (Levinas, 2014). The key benefit of this method is the ease of implementation and result interpretation. This approach does not suffer from computing speed and substantiality issues. The most prevalent kind of memory-based recommendation techniques are based on user or item.

i. *User based*: It explores to like-minded individuals and follow the assumption, which tells similar users want similar products. Different products have been suggested according to similar users.

For example,

If a student 20162020 and 20162030 like the Elective Courses in previous semester and now other Elective Course get like from 20162020, then the system recommends this course to 20162030. It performs different major task, those are:

1. Collect and organize data on users and items
2. Discovering the interests of the neighbors of an active customer.

3. Create a function that finds items that user A has not used, but which similar users have.
4. Suggest items in priority.

To suggest items for the new client, determine similar clients and calculate ratings weighted mean of items unknown by the new client. The similarity analysis relies on the active user previous rating. Schwarzkopf (2012) express in terms of pseudocode:

```
   User = C_User()
S_User = User_Similar_Neighbor(User)
for x in itemsScore_valueBy(S_User) and x not in itemsScore_valueBy(User):
   y = FinalScore(x, User, S_User)
   Suggestion += (x, y)
arrangeByFinalScore(Suggestion)
return Suggestion
```

ii. ***Item based :*** It is also known as "Item-Item Recommendation", which will explore for "item rated similarly by various users" following the assumption that if many users rated two items similarly, they will likely be similar items and worthy of recommendation; again, similar items are generally liked by similar user. This system recognize similar items depends on users' previous ratings history to recommend a new item.

For example,
If Student ID 20162001, 20162002 and 20162003 gave a five-star rating for Course Code SE500 and SE501 then when a Student ID 20162004 take the course with Course Code SE501. They also get a course suggestion to register course with Course Code SE500 because the system identifies both courses as similar based on the ratings of Student ID 20162001, 20162002 and 20162003.

According to Schwarzkopf (2012), this technique determines recommendations by first exploring the items the active user rated. Then, for each item similar to the rated items, it calculates a score based on these similarities and ratings, with rated items that are more similar to the candidate item having a better influence on its final score. Finally, it generates top n items as best suggestions of items. This pseudocode, clearly explain this filtering technique.

```
User = C_User()
Item_User = itemsScoreValueBy(User)
for x in items_neigher(Item_User):
   y = FinalScore(i, User, Item_User)
   Suggestion += (x, y)
arrangedByFinalScore (Suggestion)
return Suggestion
```

4.1.1.2 Model-based technique

This comes on the drawback of memory-based filtering algorithms including speed and sparsity issues. Model based technique is better to generate suggestion if the dataset contains large amount of information. These methods rely on the matrix factorization and are better equipped to deal with sparsity. It works by extracting a model or sample data from database for effective suggestion rather than full dataset every time. Decision trees, Bayesian method, rule-based models and latent factor models are examples of this technique.

4.1.2 Content based recommendation system

Unlike collaborative filtering, content-based filtering have various approach for valuable suggestion and born from the idea of using each item's content for recommending purposes, and trying to address collaborative filtering's drawback such as cold to start for the new user, new item problem, scarcity, and transparency. Here, ratings are not necessary; instead, identifying similar content (Pujahari and Padmanabhan, 2014).

The demerit of this method over specialize search, a client once tells the interest for a specific item, the system explores to suggest similar content items. Content-based systems use metadata for instance course credit hour, course type, department, objective and lecturer to suggest courses. For instance, if a student has shown interest in one course, the system will analyze a description of a course to find other courses.

Basic concepts that used in content-based recommendation system are TF (Term Frequency) and IDF (Inverse Document Frequency) that used to determine a relative importance of an item like document, course, article, news item or movie etc.

$$TF_{ij} = \frac{f_{ij}}{\max k \,. f_{ij}} \tag{4.4}$$

$$IDF = log \frac{N}{ni} \tag{4.5}$$

$$TF - IDF = Tf_{ij} * IDF \tag{4.6}$$

Where: F_{ij} represents for TF occurrence of term or feature in the document or item,
f_{ij} stands for the frequency occurrence of feature or term in the document or item,
N stands for the total number of documents or items,
ni stands for the number of docs that mention in term or document.

Basic steps to suggest item through content-based recommendation system, we ought to follow those subsequent steps: -

1. Describe objects by calculating their vector
2. Create user preference profile.
3. Predicting interest of the client in objects.

The above steps state that taking items dataset to establish encoding characteristics of each item. Then produce mock objects for a client relies on user item interaction. Therefore, all clients presently encoded has the same alternatives and in dataset, the representations of item are the same.

4.1.3 Hybrid recommendation system

Content-based recommendation system have many benefits over collaborative filtering. There are various pros and cons for both pure filtering methods. As this reason, most of the systems begin to use the hybrid system to mix the benefits of these two techniques and check out to convey their customers for better accurate recommendations. According to Marutitech (2019), Hybrid recommendation system can give suggestion by combining of the two pure recommendation systems. In collaborative filtering, information deficiency of domain dependencies and in content-based filtering lack data about user's preferences, so these facts lead us to use the hybrid system for better smart suggestion of products.

There is no enough data about the domain dependencies in collaborative filtering and about the people's preferences in content-based system, so those facts lead us to use hybrid recommendation system for better effective suggestion of items. In addition, content-based capabilities combined to collaborative-based system or merging the methods into one for more effective and accurate recommendation.

Collaborative filtering makes predictions using incorporation between user ratings. The correlation crucial when clients more commonly rated items. However, this will not always be the situation in big datasets. Furthermore, absence of connection to the items content prohibits coupling of similar clients if the precise identical item is having rate. Rated items and their content are used for establishing user profile that provide detail explanation of items through content-based method. Content's weight expresses their significance for users.

4.1.4 Similarity matrix

The similarity measure is a subjective measure of two data objects, which express how much are they alike (Needham & Hodler, 2019). It mainly used in advising systems, which measure the similarity of two users or items. It relies on field and application. For example, if two students are similar, they failed the same course or they have interest on the same courses. When we are calculating a distance, we should care about unrelated dimensions. The similarity is range is [0,1] in which if users have high similarity, it will be 0 and if they have low level similarity, it will be 1. Recommendation systems are mainly used the following four distance similarity metrics:

i. ***Jaccard Similarity:*** Its procedure computes similarity between all pairs of items. This similarity computes the score for each pair of items once. It does not compute similarity of items to themselves (Needham & Hodler, 2019). It is typically use Boolean value rather than rating values. It relies on related users number having rated item A and B per the total users who have rated either of the two items. Equation 4.1 expressed the mathematical formula for Jaccard similarity.

$$(A, B) = \frac{A \cap B}{A \cup B} = \frac{A \cap B}{|A| + |B| - |A \cap B|} \tag{4.1}$$

ii. ***Cosine Similarity**:* It mainly uses and measures two non-empty vectors similarity, which computed through inner product space in order to measure the angle between them in terms of cosine (Needham & Hodler, 2019). Equation 4.2 explained the cosine similarity of two vectors.

$$\text{Cos}\,(\boldsymbol{\theta}) = \frac{A \cap B}{||A||\ ||B||} = \frac{\sum_{i=1}^{n} A_i \mathrm{B_i}}{\sqrt{\sum_{i=1}^{n} A_i{}^2}\sqrt{\sum_{i=1}^{n} B_i{}^2}} \tag{4.2}$$

iii. ***Pearson Similarity**:* It measures linear correlation of two variables A and B. The original equation of Pearson similarity based on raw data and mean of two variables A and B. The raw data are center by subtracting their means (μ) and re shared by a measure of standard deviation (σ) (Dalinina, 2017). This method mathematically expressed in Equation 4.3 as two *n*-dimensional vectors' covariance per their standard deviations' product. It relies on mean and standard deviation.

$$P(A, B) = \frac{\sum_{i=1}^{n}(A_i - \mu_A)(B_i - \mu_B)}{(n-1)\sigma_A \sigma_B} \tag{4.3}$$

iv. ***Euclidean Distance:*** It measures the gap within two points of Euclidean space, which is express as the square root of the sum of squared differences between corresponding items of two vectors. The general equation for Euclidean distance d(x, y) between two n-dimensional vectors x = (x1, x2, x3, …, xn) ^{T} and y = (y1, y2,y3, ..., yn) ^{T} is given by:

$$\mathrm{d}(\mathrm{x}, \mathrm{y}) = \sqrt{\sum_{i}^{n}(xi + yi)^2} \tag{4.4}$$

Note: Superscript T represents the transpose function, so we can write a column vector easily within a line $\binom{x}{y}$ as (x, y).

The above equation works with the values of X and Y values for variation in scale. For data that used the same scale is only beneficial from this measure. The Euclidean distance between standard data is not directly proportional to the correlation coefficient (Borgatti, 2017). Mainly this distance measure has a purpose to compare profiles of respondents across variables.

Euclidean distance as a family of the Minkowski metric is that huge-scale attribute would dominate the others (Borgatti, 2017). Even though Euclidean distance has drawbacks, the data clustering process is commonly using this distance measure. The best solution to tolerate and solve such problems of Euclidean distance is normalization of continuous features.

Generally, when we tend evaluate those three basic recommendations types, hybrid recommendation has better accuracy and effective performance than the pure collaboration and content-based recommendations.

4.2 Expert System

An expert system is a program, which classified under artificial intelligence. It uses professional level knowledge about a specific area and has clear understanding to use its expertise to react and provide a reliable service for user requests. Theoretically, the expert system should replace professionals in specific domain.

Expert system helps to replicate domain specific knowledge and skills of professionals in order to solve similar problems without human experts or participation. It is a computer program, which is able to provide a reasonable solution from a human experts' knowledge in specific sector and line of argument under that the user can communicate it. It is the house of experts' knowledge and a set of rules on a specific area to guide users. An expert system helps to make decisions by asking different questions and then, based on the user's respond and the knowledge that it holds, offering suggestions. It can even explain to the user how it reaches to a conclusion. It has known basic characteristics including high performance, symbolic reasoning, adequate response time, reliability, explanation capability and clearly understandable.

An expert system has three elements, these are knowledge base, user interface and inference engine. Figure 4.2 depicted the collaboration of all those components. A user interface helps a client to request a question and to gain a response back. User-interface shall apply ten fundamental user interface principles (Nielsen & Molich, 1990) for better quality of the system. A knowledge base stores various knowledge about specific domain in which it delivered to user as per the request.

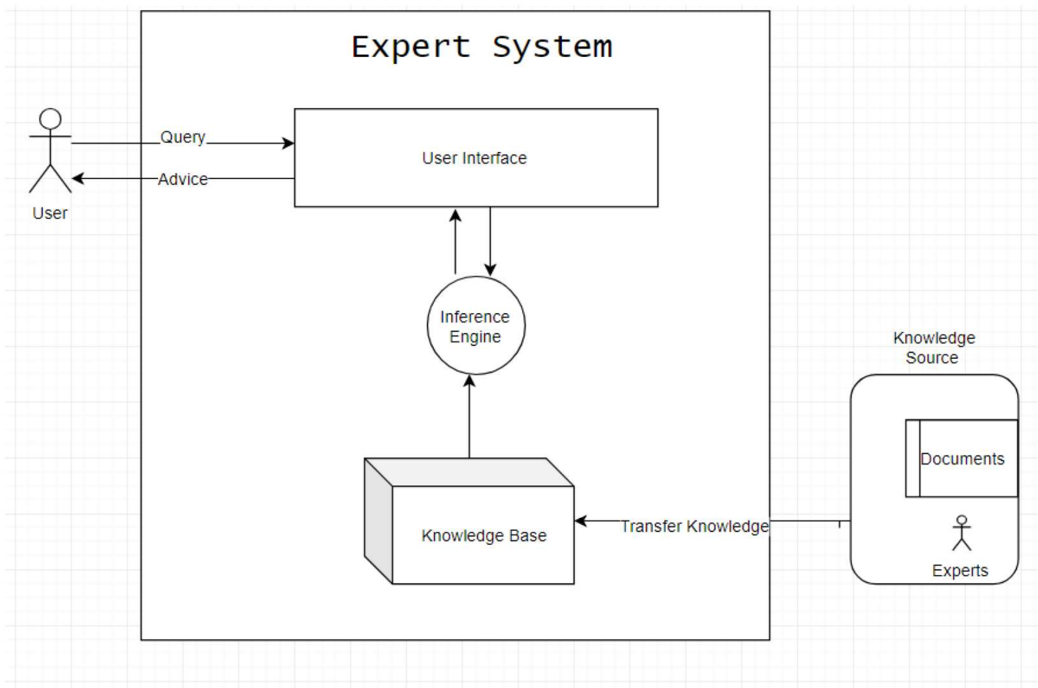


Figure 4.2: Expert system diagram

Inference engine is the internal intelligent machine of the system in which it explores knowledge form the warehouse to provide appropriate response to it. Expert system classified in to rule based and case based system.

4.2.1 Rule based expert system

It belongs to what we often called inductive approaches. Inductive approaches will try to induce a "compact" model from your past observations/experiences, and to generalize (make an induction) from those. Once this is done, you can forget your experiences and only use your learned model to make a prediction. In this system, rules represent more knowledge, in which they are if –then- else statements. Rule-based expert system's block diagram with its functional components is depicted in Figure 4.3.

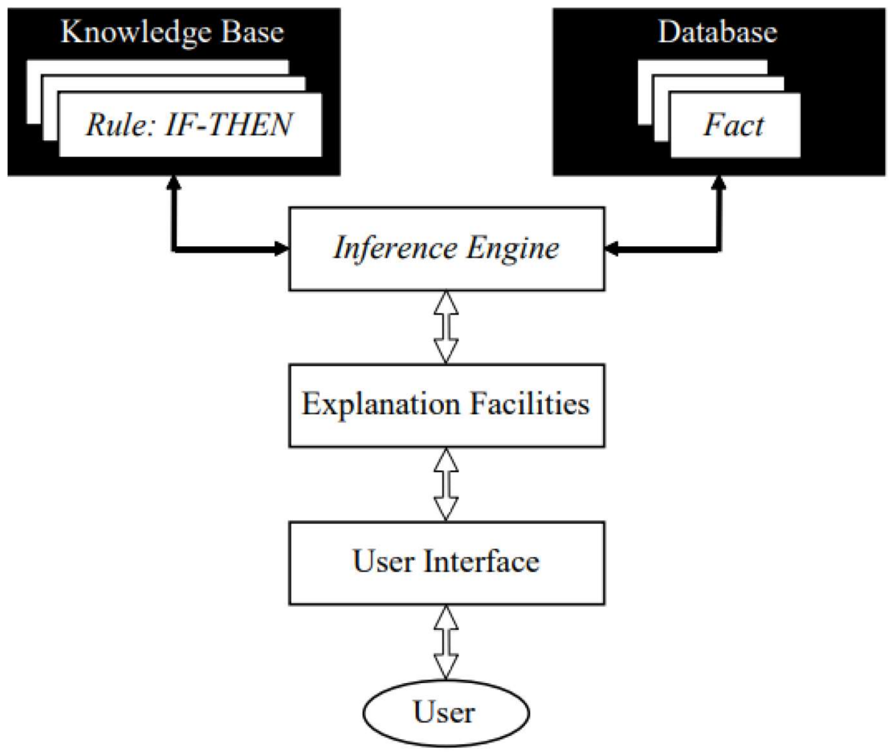


Figure 4.3: The complete structure of rule based expert system (Negnevitsky, 2011).

There are two types of inference chaining, such as forward and backward chaining (Negnevitsky,2012).

i. ***Forward Chaining*:** An argument begins from the known data and continue with that data in data-driven approach. It is a mechanism to collect knowledge and facts then inferring to disseminate response as per the request. In this chaining, the system executes many rules without their usage. Hence, this method can't effectively explore one particular evidence (Beemer & Gregg, 2006; Henderson & Goodridge, 2015).

ii. ***Backward Chaining*:** The expert system estimates a solution or goal and the inference engine claims to search the evidence to prove it. Mostly it refers to "goal-driven engine" which derived from its intended function. First, search rule in knowledge base that might have the desired solution. It is working with the rule and sets up a new goal, a sub goal, to prove the condition part of this rule. Then the knowledge base explore rules again, which can prove the sub goal (Henderson & Goodridge, 2015).

4.2.1 Case based expert system

This method will not try to induce a generic model from your past observations or experiences, but will simply try to explore the answer to a new problem by looking at experiences (Daramola et al., 2014). This means you have to maintain a memory of an experience, and then select the most adequate one to answer your new problem. In a case-based system, the composition of various issues with its corresponding response categories to represent knowledge. It focuses with providing a solution for new situations according to its experiences. When a machine is thinking to solve various issues, it vividly remembers previous histories that have high similarity to present scenario. If the present and past scenario is adequately the same, then the system provides an identical solution to both cases (Daramola et al., 2014).

As expressed in Figure 4.3, the basic steps in case-based expert system cycle are:

1. Retrieve matching case to a given problem
2. Adapt solution of a case to generate solution
3. Evaluate and review recommended solutions
4. Keep realistic and proven responses by incorporating to the request as case base for coming requests.

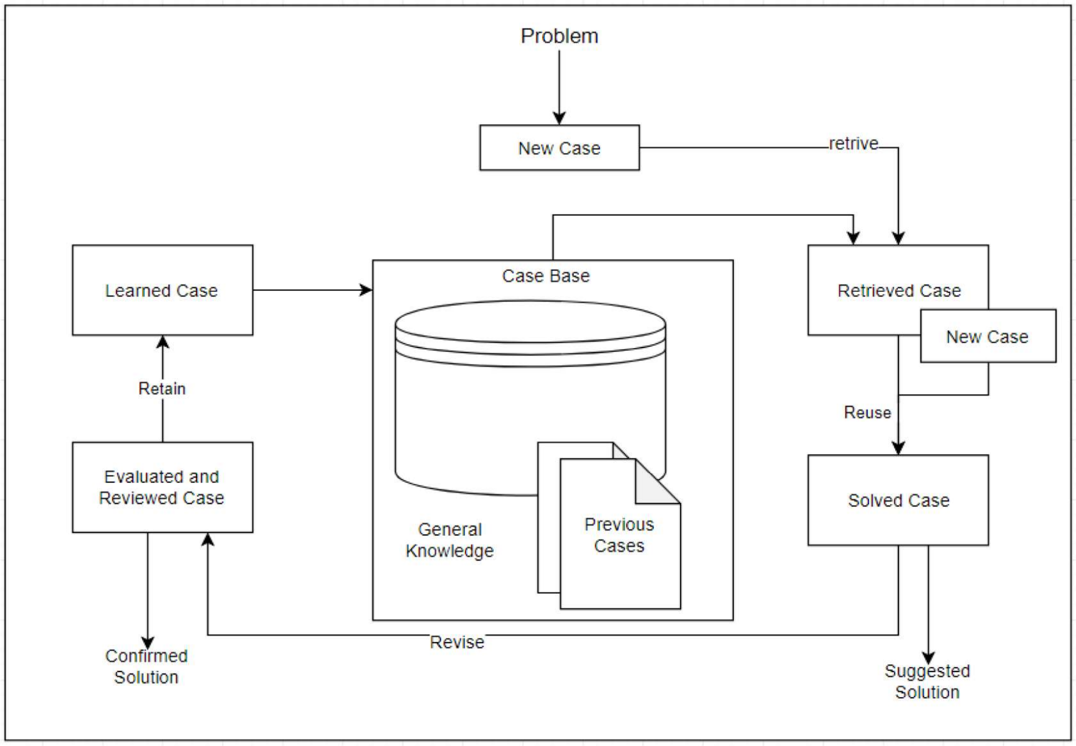


Figure 4.4: The case based reasoning cycle (Aamodt and Plaza, 1994)

4.3 Machine Learning Techniques

Machine learning is artificial intelligence's application in which it immediately learns from data in order to enhance its performance. It mainly targeted to develop a system that uses data to educate itself. Learning begins from access of data including experience and commands then explore the appropriate patterns to make further decisions for the coming issues. Dependency and correlations cannot be known instances in training and test set usually have large number of attributes. The algorithm needs to evaluate in changing environment.

As mentioned by Expert System Team (2017), the classification of machine learning in to supervised and unsupervised depends on the intended purpose and desired result of .the algorithm. Supervised machine learning algorithms is a technique for creating function from

trained data. It can apply the knowledge from its experience to the coming data through learning form labeled samples to put hypothesis about future events.
Unsupervised learning used to construct patterns form the datasets, which excluding labeled response for various input data. Usually, cluster analysis is belonging to supervised learning method in which observational analysis search unseen patterns or data clustering. Unsupervised learning deals about the process of inferring the operation using unlabeled data to clarify the unseen structure.

Semi-supervised machine is logically appeared between unsupervised and supervised learning since they are playing with tiny quantities of structured and enormous quantities of unstructured data for training. This activity makes the system significantly enhance an accuracy of learning.

Reinforcement machine learning is a mechanism the system and an environment interact each other through action and its response. The response might be failed or success. It makes the machines or software agents to identify the logical characteristic with a particular situation to enhance its performance. An agent receives success response in order to learn the best action so called “reinforcement signal”.

Machine learning can analyze an enormous quantity of data. The result should be measured in terms of speed and accuracy to decide whether it has a better opportunities or dangerous risk, (Dash & Vaidhehi, 2017).

4.3.1 K nearest neighbors (KNN) algorithm

K nearest neighbors holds all accessible instances to grouping fresh instances depends on a measure of resemblance. It makes a training data of grouped labels then filter K most comparable items. KNN is passive algorithm despite it can’t prepare any claims about fundamental data distribution and there is limitation of deduction.(Wu et al., 2008).
As Umamaheswaran (2018) mentioned, the algorithm for the development of KNN model had expressed as follow:

Read test and training dataset
Select K value

Till test data is null:

search Euclidean distance at individual value for all training data

save Euclidean distances to the list and make it in order

select starting of k points

test point of a class assigned depends on the highest number of classes appeared in the selected points.

End

KNN has some advantages including there is no any claims about data; ease of explanation, understandability and interpretability; relatively more accurate but less than supervised learning models; versatile which means it is better way for grouping and regression. In other ways, it has also demerits like, computationally fancy despite of an algorithm that store all training data; it must compute the distance and sort all the training data at each prediction, which can make it slow.

4.3.2 C4.5 Decision tree algorithm

Decision tree has a hierarchal shape in which it consists root node (the top of the tree), leaf (class label), internal node (test of attributes), branches (have the result of a test). Due to its structure, this algorithm has ease of interpretation and reliable, robust, simple to implement and understand. It classified under supervised learning in which it has easy way to predict information through tree rules.

C4.5, ID3 and J48 are decision tree algorithms in which they take agreed and top down method to construct a decision tree. C4.5 is relevant for the classification of new label data, according to the training data set Ruggeri (2002). Decision tree induction convinces with a training set in which every node resulting in smaller partitions, thus following a continuous divide and conquer strategy. Activities and attributes are the information related to the object or an event. A class label is associated with every tuple in the dataset in, which it determine the object to grouped a particular class. When the tuples put in various classes, the separation will be happened.

Dataset partition uses heuristics to select an attribute for best partition dataset. This refers to an attribute selection measure. It is responsible for a type of branching in which it applies on a node. Gini index and information gain are examples of attribute selection measure. Gini index partitioning a node into binary whereas information gain split anode in to multiway. If we need to convert one tree structure into others, we can use certain other ways. It has an advantage such as easy to interpret results and helps to visualize through a decision tree.

Generally, C4.5 algorithm generates decision trees for classification which also known as a statistical classifier. It uses both discrete and continuous data with the concept of information entropy Ruggeri (2002). If there is an incomplete data issue. It will easily handle it. It inherently employs single pass pruning process to mitigate overfitting.

4.3.3 Random forest algorithm

Random forest is also known as "random decision forests" which uses a collection of decision tree learning techniques for classification, regression and for various activities that build a model from many decision trees at training time to get more accurate and stable prediction(Donges, 2018). This algorithm combines weak classifiers to produce strong classifier. The randomness of this algorithm has two main concepts, the first is trading data point selection through random sampling during tree building process and the second one is during node splitting process, it considers features random subsets (Koehrsen, 2018). The expected outcome from this algorithm is classes or mean prediction of the individual trees.

According to Polamuri (2017), Random forests has biased towards information such as clustered variables with different number of stairs of attribute with higher stairs. Therefore, the variable significance values from this algorithm are not important to this type of data rather it mainly used for predicting missing data and if huge amount of data missed, it enhances accuracy. Hence, it can run efficiently on large databases and it has the ability to handle many input variables without deletion and gives a prediction the variables that are vital for classification. It produces internal free from bias prediction of deduction mistakes as the forest construction process and balancing error in class population biased dataset. Random forest algorithm pseudocode has two stages; those are random forest building and estimation generator pseudocodes.

4.3.4 Native Bayes algorithm

Naive Bayes classifiers are not a single algorithm rather they are a family of classification algorithms depends on "Bayes' Theorem" as a basic principle, which means all feature couples are grouped independently. Specifically, they are suited when the inputs have high-level dimensionality. They are probabilistic machine learning models used for task classification.

Bayes Theorem:

$$P(A\backslash B) = \frac{P(B\backslash A)P(A)}{P(B)} \tag{4.5}$$

Through Equation 4.5, it is easy to determine the A's probability (hypothesis) form known B (evidence) has occurred. Native Bayes algorithm have assumptions those are predicator or features are independent each other, and all these indicators equally affect the outcome. When we say naive, it means the presence of a specific feature does not affect the other. To compute a probability of observing features form f_1 up to f_n, and c is given as a class Then under the Naive Bayes thought Equation 4.6 and 4.7 holds:

$$P(f_1,\ldots,f_n\backslash c) = \prod_{i=1}^{n} P(fi\backslash c) \tag{4.6}$$

$$P(f_1,\ldots,f_n\backslash c)\ \alpha\ \ p(c)p(f_1\backslash c)\ldots p(f_n\backslash c) \tag{4.7}$$

According to Ray, (2019) Naïve Bayer algorithms are simple and fast estimate the test data set's class especially, in multiclass estimation. If this algorithm has autonomous assumption, it will have better performance because it requires less training data. It has better performance in grouped input variables compared to the numerical variables. In the other hand, if there is a group of classified variables within test data set, so it is not shown in the training dataset, then it gives zero probability due to this a model can't predict. Smoothing technique has ability to solve this problem. Naïve Bayes model have three classifications; these are Multinomial, Bernoulli and Gaussian models. The main difference are on their predicator such that they uses frequency of words, Boolean variables and neither continuous nor discrete values (derived from Gaussian distribution) as predicator respectively.

Those algorithms are applicable for spam filtering, real-time prediction, sentiment analysis, multi-class prediction and advising systems, etc. The implementation process is fast and easy. The feedback of those algorithms is that the predictor's requirement to be independent but in fact, they have dependent predicators.

4.4 Association Rule Mining

Mostly people assume data mining and knowledge discovery from repository, as similar concept even though some academicians state data mining is necessary element or steps for knowledge discovery. Nowadays information and data storage technology have rapid growth and broad implementation. This leads to emerge knowledge discovery to and data mining as a new technology. Data mining are mainly focus on the development of methodologies and resources for making automation of data analysis and creation of vital knowledge to support decision. Data mining used by intelligent methods for exploration of data patterns.

Association rule data mining is the way to explore patterns in data to find features that occur together and correlated. It mainly involves on machine learning for analyzing various data from different patterns or mutual existence in the repository. It determines conditional cooperatives. We can use such rule in any dataset where features take either zero or one.

For Example,
Let Course = {CourseCode_1, CourseCode_2, CourseCode_3, CourseCode_4} are items that defined as group binary courses.
Let Enroll= {Trans_1, Trans_2} are the group of transactions known as database. Every transaction included in Enroll set contains a subset of items in Course set.
Trans_1 ⇒ Trans_2 where $A, B \subseteq X$ and $A \cap B = \Phi$ (Here, Trans_1 is a course or a group of courses)
*Trans_1: Antecedent, Trans_2: Consequent

Table 4.1 indicates the presence of course in that transaction, and 0 shows the absence.

Rule: {CourseCode_1, CourseCode_2} ⇒ {CourseCode_4}, meaning that if Course_1 and Course_2 are enrolled, a student also select course_4.

Threshold used for relation are support, confidence, lift, conviction all confidence and collective strength. It used for the exploration of frequent patterns, causal structures, associations or correlations in various dataset from diversified repositories including transactional and relational data repositories (Jain, 2019). In general, the ultimate aim of association rule mining is making hypothesis to a particular item occurrence depends on other items within the transaction through exploring rules.

Table 4.1: Course enrollment transactions

Transaction ID	CourseCode_1	CourseCode_2	CourseCode_3	CourseCode_4
1	1	0	0	0
2	1	1	0	1
3	1	0	1	0

4.4.1 Priori algorithm

A priori algorithm is an influential algorithm, which computes on data repository consisting various transactions. It produces a set of candidate items only form historical large set of items. This algorithm considers any frequent item subset as a frequent item set. Hence, the algorithm searching the item sets who have greater support count in order to reduce the number of considered candidates. A priori property is helps to reduce the search space, which leads to improve the efficiency of frequent item sets generation process (Wu et al., 2008).

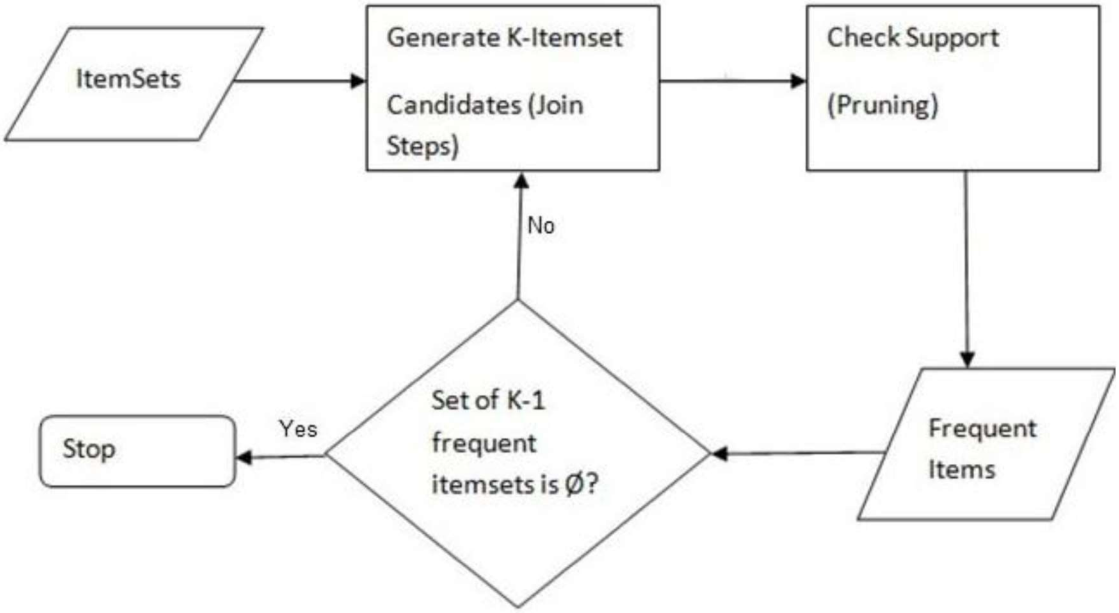


Figure 4.5: General process of a priori algorithm (Kavyashri, 2018)

Priori algorithm explores frequent item sets, which consists of an item with minimum support. It applies "bottom-up" approach, where it extends one item of frequent subsets at a time. The entire algorithm has two basic phases (Jain, 2019):

- ***Phase 1:*** Explore all the frequent sets with k items in a data repository by applying a minimum support.
- ***Phase 2:*** By using self-join rule find the frequent sets with k+1 item with the guidance of frequent k-item sets. Do continuously such process from k=1 to the point when we cannot apply the self-join rule.

A priori algorithm has limitations, those are:

- It has very slow computing speed due to candidate generating process;
- The repository needs (n+1) times scan, where n is the longest pattern's length.

CHAPTER 5
INTELLIGENT AND PERSONALIZED COURSE ADVISORY MODEL DEVELOPMENT

5.1 Overview of Course Advisory System

Course advisory system is the process of giving suggestions about courses depends on student's preference and ability in order to support decision-making process during course registration period. It involves an academician giving counsel to their assigned students during course registration period in order to satisfy established academic requirements. This system enables students to get the right information at the right time to make right decision in courses to achieve their educational goal. Educational institutes practice course advising during course registration timeline. because this system suggests different courses that are to be added or dropped depends on the rule and regulation of the institute, courses nature, students' academic history.

Mostly, course advising processes in many educational institutes held by contacting with an advisor through mail system, telephone or in person, which consume more time, and when many students have assigned for one academic advisor, it has many problems to provide right advising for every students because the process by itself is the most fed up activity.

Students in HEIs expected to fulfill some performance criteria in order to pass from one stage to another, with a specified number of credit units to be passed among a set of compulsories (core), electives, and common courses. The responsibility of course adviser is to guide and ensure that a student makes good decisions on courses that should be registered relative to the student's current level and academic history to fulfil graduation requirements. The course advisory activity is a domain for the application of the expert system, artificial intelligence, recommendations system due to domain-specific knowledge, voluminous data, inaccurate characterization, curriculum flexibility (changing quickly through time), and decisions which have to be made based on the specific rules of the University concerned. Many of the decisions made by a human advisor during the process of advising a student depend on reasoning drawn from previous episodes and experiences that the advisor had gained over time, and known rules of the department as well as the institution that relates to course registration.

Human course advisors have skilled in counseling and interpersonal relationships, able to listen, able to be directive and non-directive, able to demonstrate patience and tolerance and careful about details such as record keeping, follow-through, and follow-up. Course advisory system should be student oriented which means having an interest in and concerned for students as individual. If there is a change in the knowledge about the requirements and policies of the institution should be considerable. However, many HEIs has been following different rule and regulation of course advising. Even departments in one HEIs have different curriculum, this makes students advising system more complex.

5.2 Intelligent and Personalized Course Advisory Model Analysis

One advising technique may effectively work for some HEIs but not for others, so this leads many problems in students, advisors, academicians, and departments as well as HEIs especially, it does directly affect transfer students who adapts different advising methods. HEIs have different SIS as per their requirement or needs including grading system, course advising method as well as rule and regulation. Consistently changing those methods, rules and curriculums leads to make expensive SIS maintenance cost. Course advisory system does retrieve student's academic history, course description and department rules to use, rather than manipulate of academic records from database. Hence, the proposed model overcomes on those drawbacks of traditional advising systems to use the same academic advising structure for all HEIs as framework, which enables flexibly change the rule, regulation, curriculum and grading system without affecting persistent academic data. The model is highly compatible to all HEI's SIS components, because it uses intermediate CSV file dataset, which import and export from the system so there is no direct contact between course recommendation and SIS database.
The emergence of IPCAM helps to tolerate the gap between HEIs and reduce the time spend in course advising through regularly assisting both the advisees (students) and academic advisor to accomplish their goals.

5.2.1 Characteristics of IPCAM

The course advisory model is special designed model for the unique needs of each student for providing a suggestion or helping decision-making process during his/her course registration period at the beginning of every semester. The key advantages of the customized model are that it is custom-built, which means it is specifically built according to student's individual requirement, so using this system model students can make a decision with fully informed manner without any human advisory or other guides.

It can easily merge with the current SIS system and integrate without any discrepancy to HEIs SIS. By using this, students get course suggestion service based on their favorite area of interest, their department's curriculum, their academic history, and rule and regulation of the university while the rest can be.

- The model cannot use original database as a common with SIS to make any change on it rather it uses the copy of it as a training set. It makes to protect against malicious attack and other hacker risks that attempt in the recommendation system so that the SIS continues to advise students correctly under such potential risks. Security is necessary to provide integrity, authentication, and availability of the full system.

- It also has the capability to perceiving and interpreting data sensed from its environment, reflecting events in its environment, and taking actions to provide an effective course suggestion without permanent guidance from its student and an advisor. It has the intrinsic ability to communicate, cooperate, coordinate, negotiate, and learn, as well as have the capability to evolve through interactions with other components such as course registration system that can be standalone or part of SIS.

Generally, this model shall be more flexible, personalized, intelligent, compatible, efficient, and secure and has high performance.

5.2.2 Requirements analysis

The following steps are used for students to register necessary courses:

1. Authenticate to determine whether he/she fulfill minimum requirement to be registered in to course based on the previous academic performance.
2. Choose opened courses in order to take for this specific semester.
3. Add or drop courses after initial registration held on step 2 and validate course prerequisite and elective courses.
4. Validate the rules that guide total number of courses to register and the combination of courses to register.

To avoid violation, students and advisors must have adequate knowledge and understanding about the rules and regulation of HEI, and the procedures that explained above. Once students enrolled, they have to discuss with their academic advisor to select the most preferable courses according to student's academic history to register for specific course. However, Due to an intelligence and personalized course advisory model, students automatically get course suggestion in a customized manner without any intervention academic advisor.

The detail analysis of this model capture in Figure 5.3 to reveal a number of specific requirements that a course advisory system achieves.

1. The system shall receive student information from SIS registration system to analysis and filter student academic history.
2. The system shall analysis and filter appropriate courses based on student academic history.
3. The system shall validate prerequisite, minimum and maximum number of courses per semester.
4. The system shall provide interface to add special interest if and only if the student has to take elective courses.
5. The system shall suggest appropriate courses depends on student's previous academic history and their special interest.

6. If there is, any change in SIS that are necessary to course advising, Course registration system shall continuously update information in .csv format.
7. System Administrator easily change the rule, regulation and grading system based on institutions SIS.
8. Academic experts shall give vote according to their knowledge and skill for professions or referring the current trends of each course (specialized area) respective to courses in academician portal of SIS.

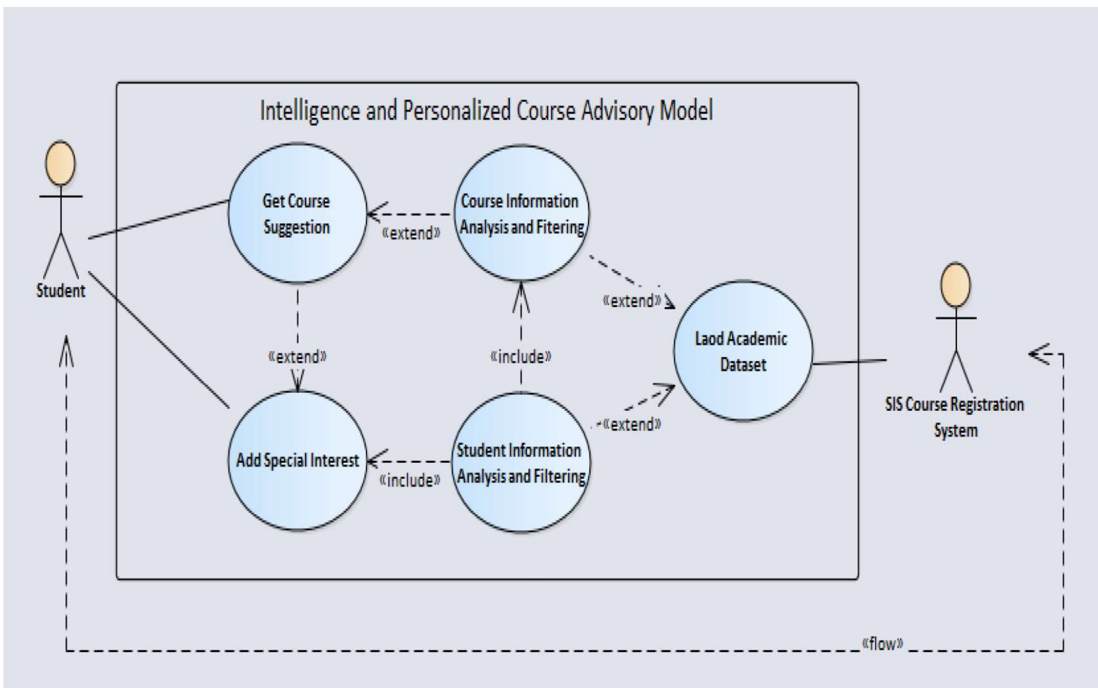


Figure 5.1: IPCAM use case diagram

5.2.3 IPCAM processing

Figure 5.2 tells, IPCAM implicitly performs many basic activates to achieve its main objective. It also describes the flow of information form one activity to another as follow:

1. Students authenticate SIS using their own unique ID (user name) and password.

2. HEI database automatically academic data set, so if there is any change in database this dataset is also automatically update itself.
3. System Administrator assign and modify priority score according to the rule and regulation of HEIs and the department head may change the curriculum as per the decision of the committee members.
4. If student is new for the institute, the system shall suggest appropriate courses based on the default priority assigned by the department head and institute's system administrator.
5. If the student is not a new for the institute and he/she has failed courses, the system specify failed courses.
6. If the student is not a newcomer and he/she has not failed courses the system give lowest priority (score = 0) for passed courses.
7. If not in number 6, dependent courses on failed prerequisite course are get lowest priority (score = 0).
8. If the student eligible to take elective course according to his/her academic history so, he/ she select a special interest (favorite profession that his / her department may achieve). If not the system suggest appropriate courses.
9. The academic advisory system prioritizes elective courses based on the mean expert rating that collected from a SIS.
10. The system automatically suggest appropriate courses as per student's academic history.

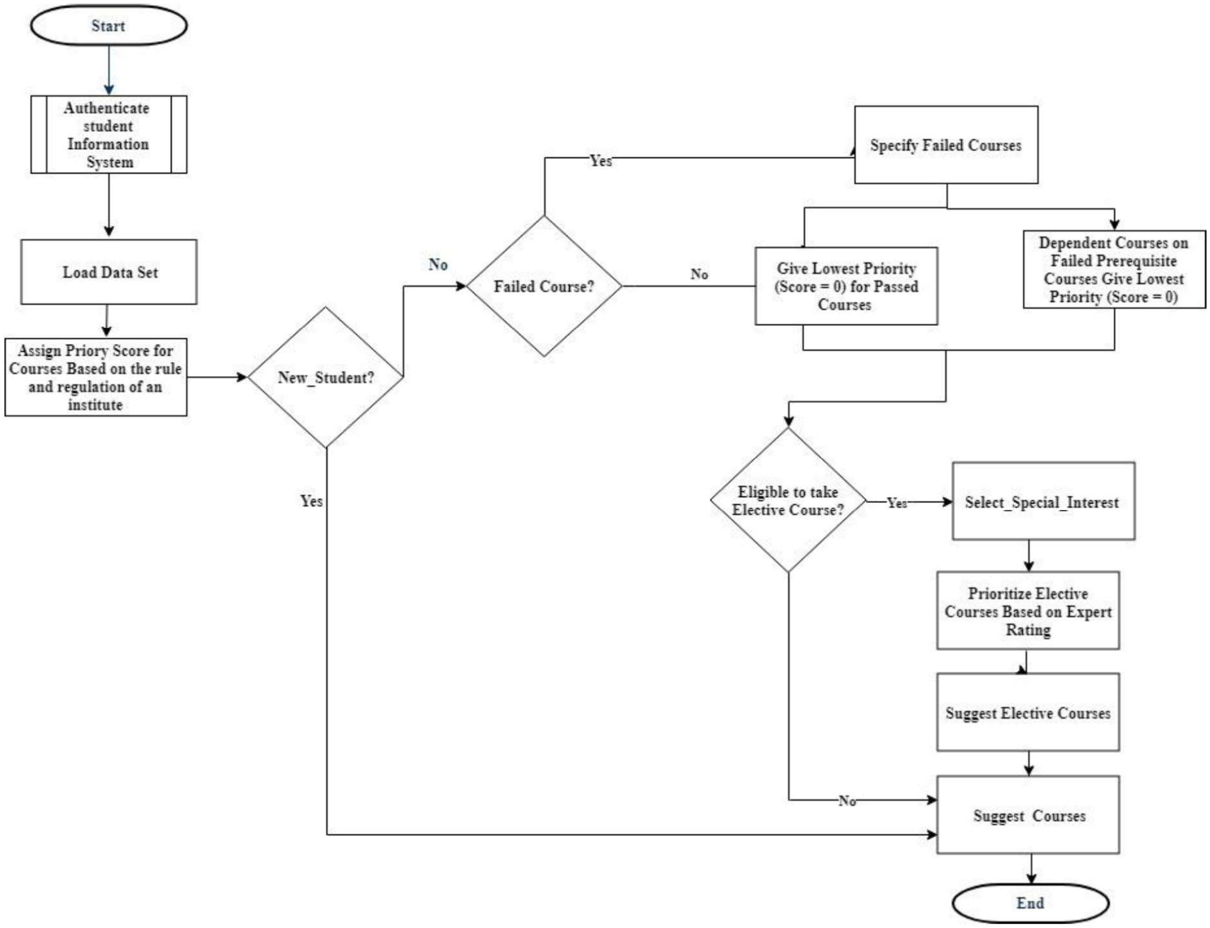


Figure 5.2: IPCAM processing

The above process working with seven components in which they modulated in low coupling and high cohesive processes with specified input and output. Those components are student information analysis, taken courses analysis, prerequisite courses analysis, elective course analysis, priority configuration, special interest analysis and finally course suggestion.

Figure 5.3 depicts about the activity diagram of student information component, which is a well-defined procedure that allows a computer to operate with student information for personalization.

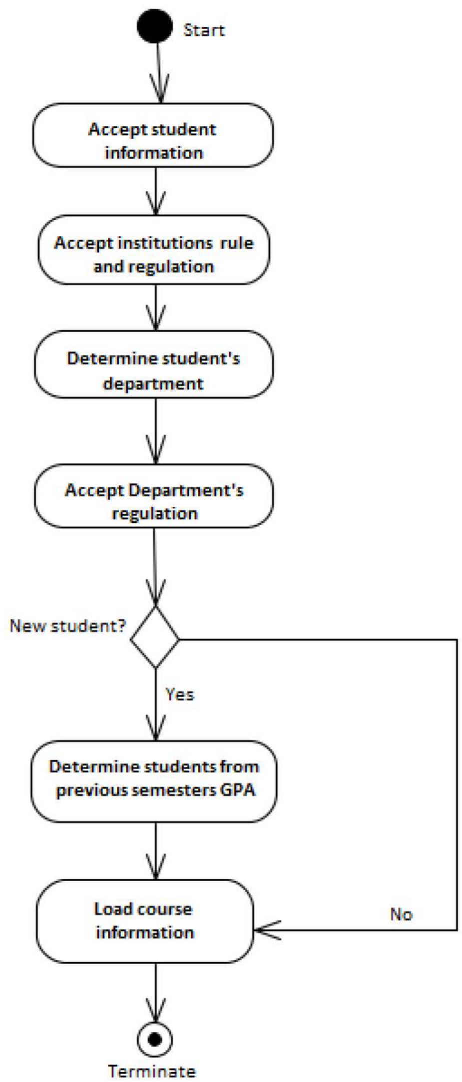


Figure 5.3: Activity diagram for student information analysis

As Figure 5.4, Taken course analysis component have well-defined procedures that allow a computer to analyze registered courses of a particular student.

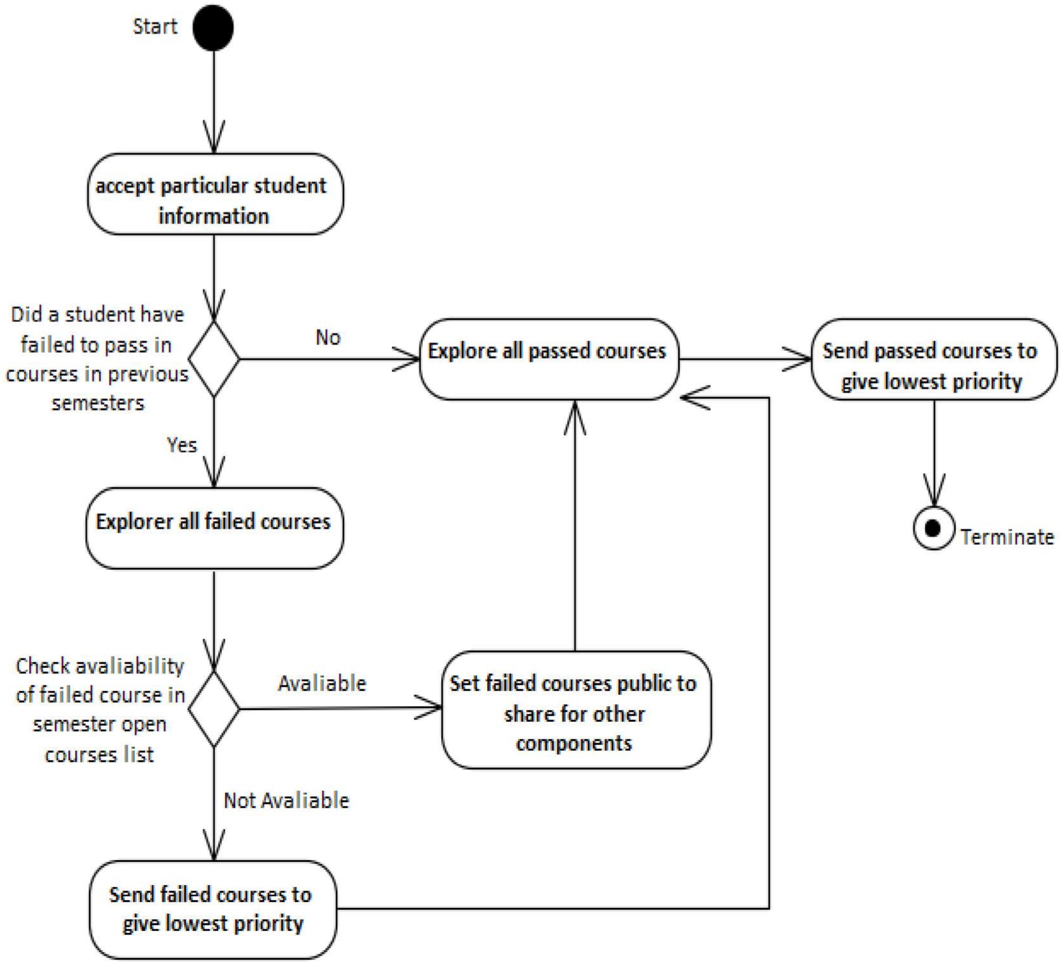


Figure 5.4: Activity diagram for taken course analysis component

Prerequisite course analysis component allows a computer to analyze prerequisite courses and its corresponding dependent courses of a particular student. Figure 5.5 briefly depicts the activity diagram of this component.

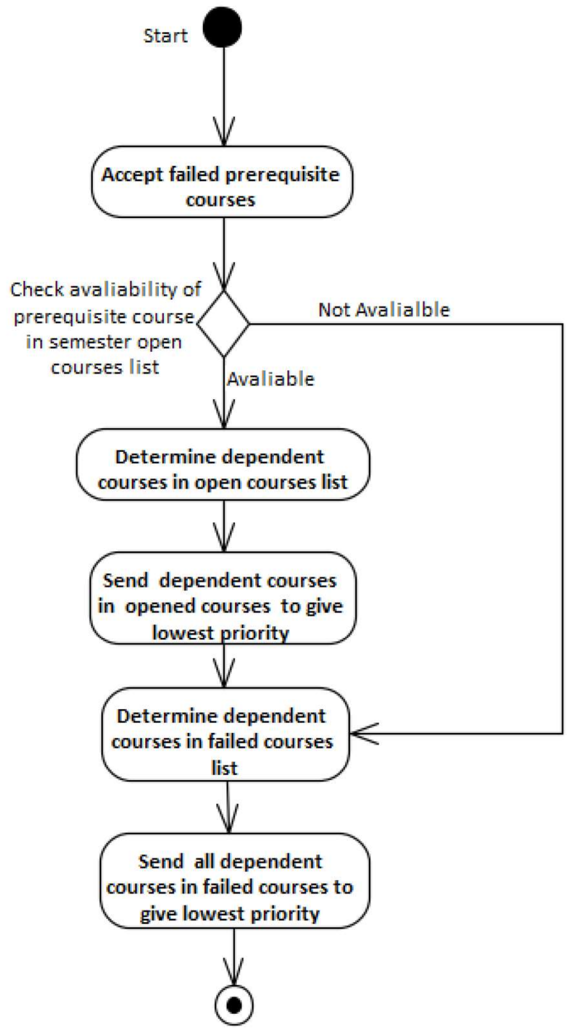


Figure 5.5: Activity diagram for prerequisite course analysis component

Elective course analysis process allows advisory machine to analyze elective courses depends on student's special field of interest. Its overall process depicted in Figure 5.6.

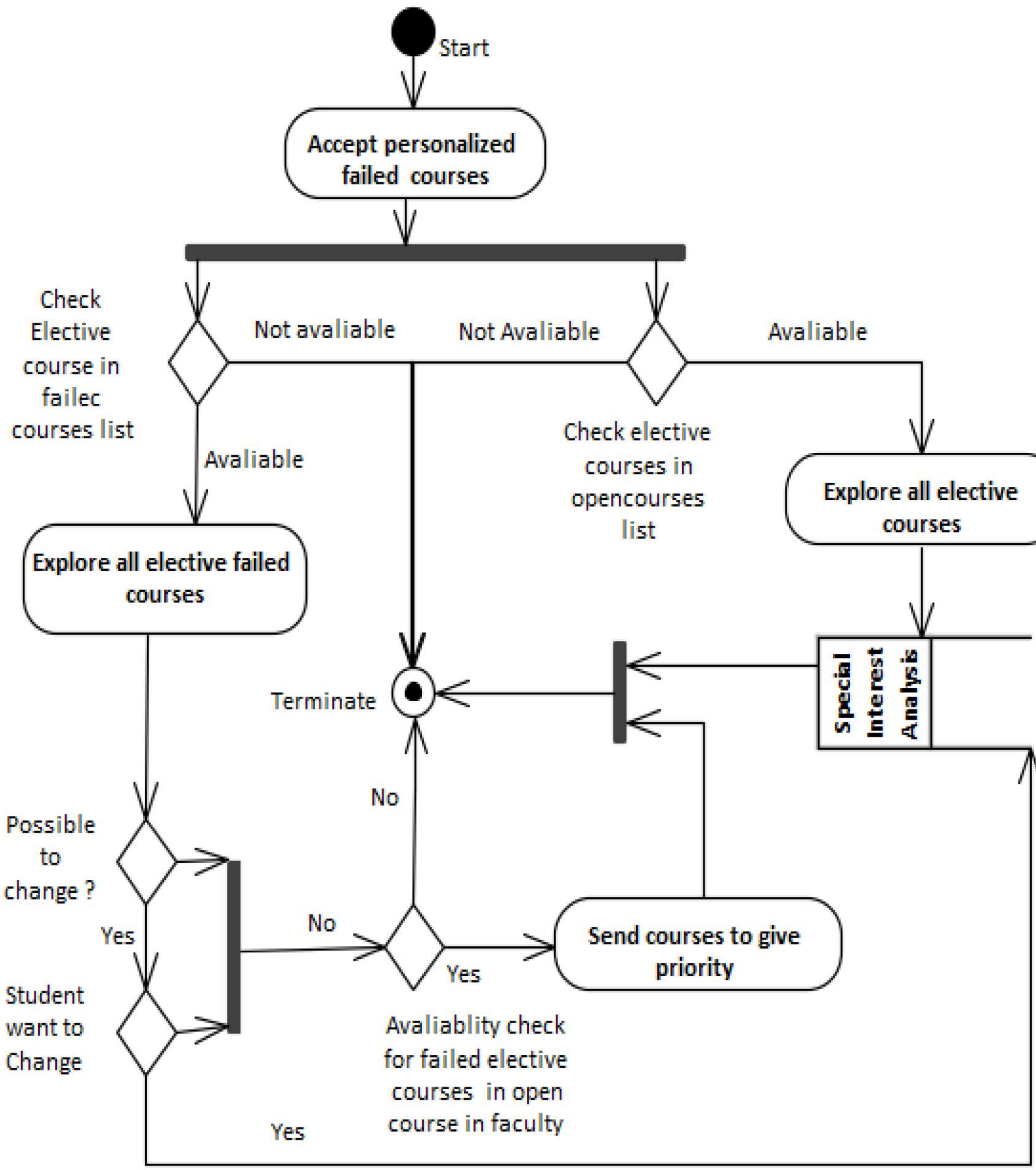


Figure 5.6: Activity diagram for elective course analysis component

Special interest analysis process allows advisory machine to analyze personalized student's special interest as shown in Figure 5.7.

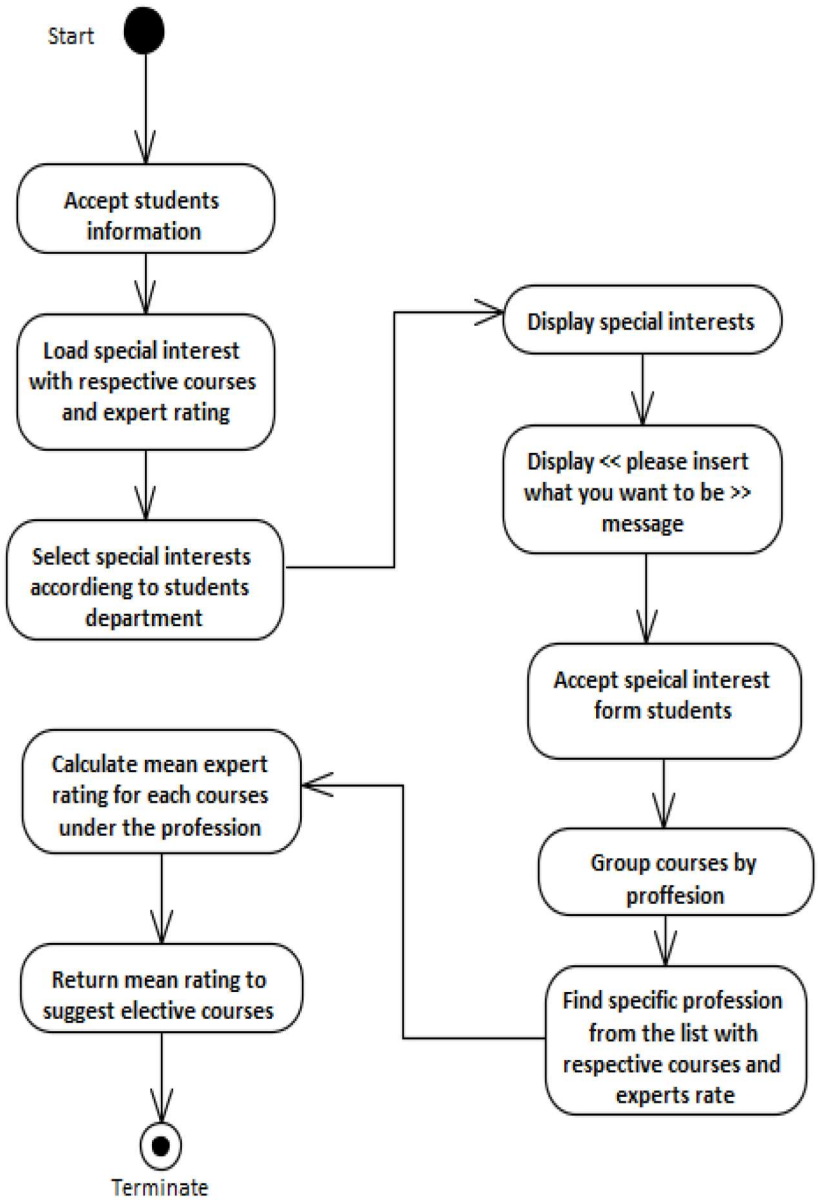


Figure 5.7: Activity diagram for special interest analysis component

Course priority configuration allows a computer to configure and adjust course priority score according to the information delivered from other components, department's curriculum and regulations of particular HEI. It depicted in Figure 5.8.

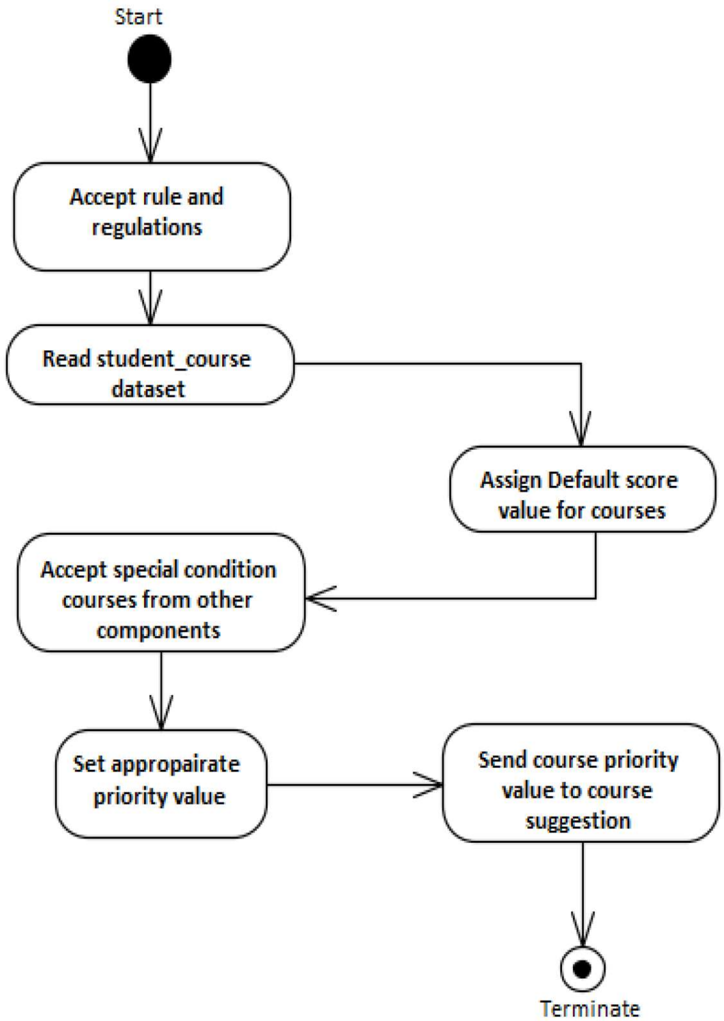


Figure 5.8: Activity diagram for course priority configuration

Course suggestion process has well-defined sub procedures that allow advisory machine to recommend courses in a personalized manner. Its overall process analyzed in activity diagram as expressed in Figure 5.9.

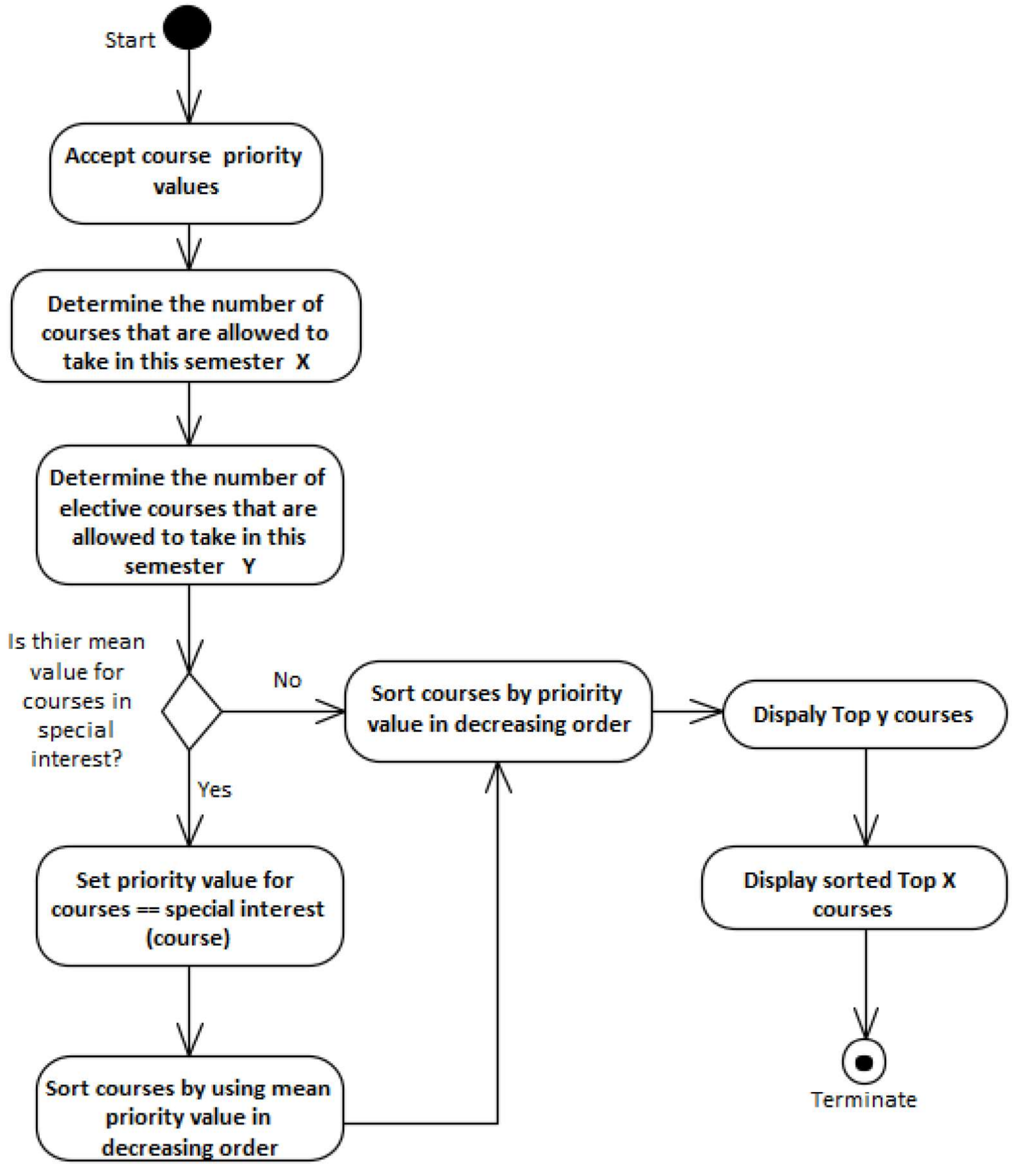


Figure 5.9: Activity diagram for course suggestion component

5.2.4 Data management

Data management is a process for ensuring the accessibility, reliability and timelines of data for students or advice, which include acquiring, validating, storing, processing and controlling data. While some software development companies are good in data collection, they are not managing well enough to make sense of it. Acquiring of data is not enough for the development of effective software model but also data representation, data analysis, data management are necessary.

Many educational institutions have different database management tools and technologies. Their database design also developed according to various rules and regulations. We have to manage those various data through data preparation and data analysis to develop effective software model.

i. *Data collection and preprocessing*

According to the data that collected from different universities through the methods that explained in Section 3.1 is quite complex in order to develop a common framework for course advising system so we need to make a compensation. SIS, rule and regulation of courses and course advisory system in many universities including NEU, Wollo University, Cyprus International University, University of Kyrenia, Addis Ababa University, Gomel University of Pakistan, Black University of South Carolina, King Abdulaziz University of Jeddah, and American University of Beirut are different.(Muhammad & Abdur, 2011; Moogan et al., 2003; Feghali et al., 2011;).

However, we should use common framework that have direct impact on the course advising such as student, course, instructor attributes, season and academic year. The rule and regulation of HEIs, department curriculum and grading system must be compensated to develop a best framework

.

ii. *Data analysis*

Many HEI’s SIS uses different database tools, technologies, structure and data representation format. In IPCAM development, various institutes’ data should be collected and organized as dependent and independent data. Dependent data are academic data, which are common for all institutions, and they can directly integrate to IPCAM, whereas, Independent data are different from institute to institute in which they organize and operate in different interacting components of HEI SIS. Comma separated values format can gather information form HEI SIS's database by integrating students, courses and instructor's information in query report form. Figure 5.11 briefly explain about course attributes in different perspective.

It is difficult to change basic information without maintaining the system in order to make it a framework for all HEIs. However, so to develop common framework, we have to identify common rule and regulation of different institutes before the actual development begin. During integration of this system to existing SIS, System administrator can change necessary information as per the institute's grade system and the rules and regulations.

The transformation mapping method is helps to exhibit distinct boundaries between incoming and outgoing data. According to Figure 5.10, the system have a serious of steps in which to deliver appropriate course advice service to individual students through analyzing academic data from academic dataset which exported or updated form SIS database or generated form academic experts.

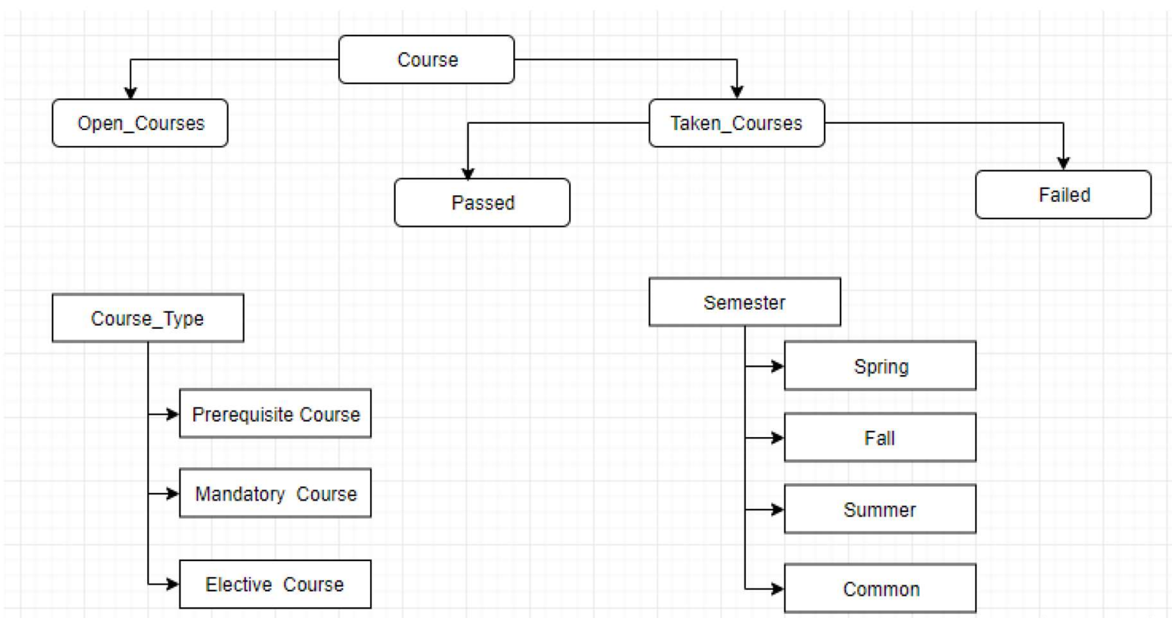


Figure 5.10: HEI common academic data hierarchy

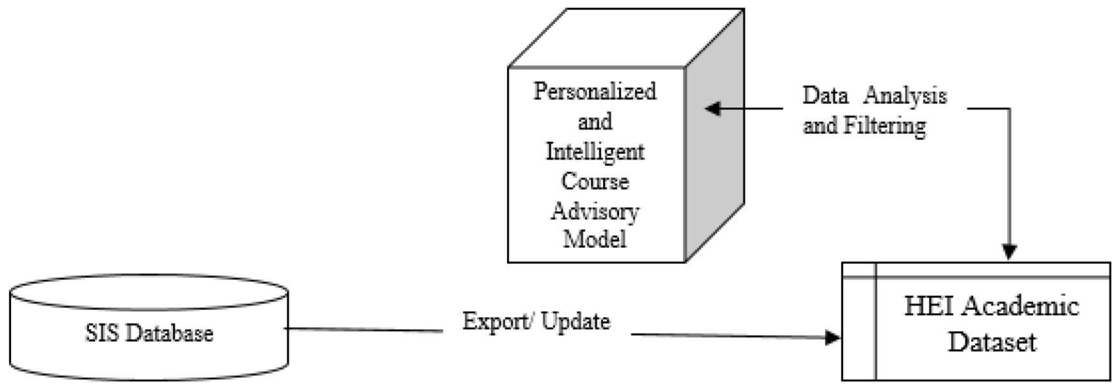


Figure 5.11: Academic data storage integration with course advisory model

5.3 IPCAM Design

A design uses analyzed requirements to transform into some suitable and structured form, which guides the coder in system implementation and maintenance phases. This topic describes the model design using architectural design and detail design:

5.3.1 Architectural design

Architecture of academic advisory model is the fundamental structure, which shows all its elements with their functions and relations. It also describes how to address the key system objectives including the performance and reliability. Academic advisory model architecture vital to maintain or make modification, to help the developers by explaining the fundamental structure, to address key risks, and to increase reusability and adaptability of the system.

IPCAM is the sub component of SIS, which is directly and indirectly interact with another component such as SIS authentication, course registration, SIS persistent database and academic dataset. It automatically gets some data about student, course, and academic semester from course registration component. It uses those few data in order to filter, analyze, generate transient data in academic dataset then finally intelligently suggest courses according to student's academic history and interest.

SIS persistent data storage/database use database technology to store educational information. It retrieves students, courses, instructors and different information about an institute as per the query of different systems or SIS components. It can also generate academic dataset for academic advisory model.

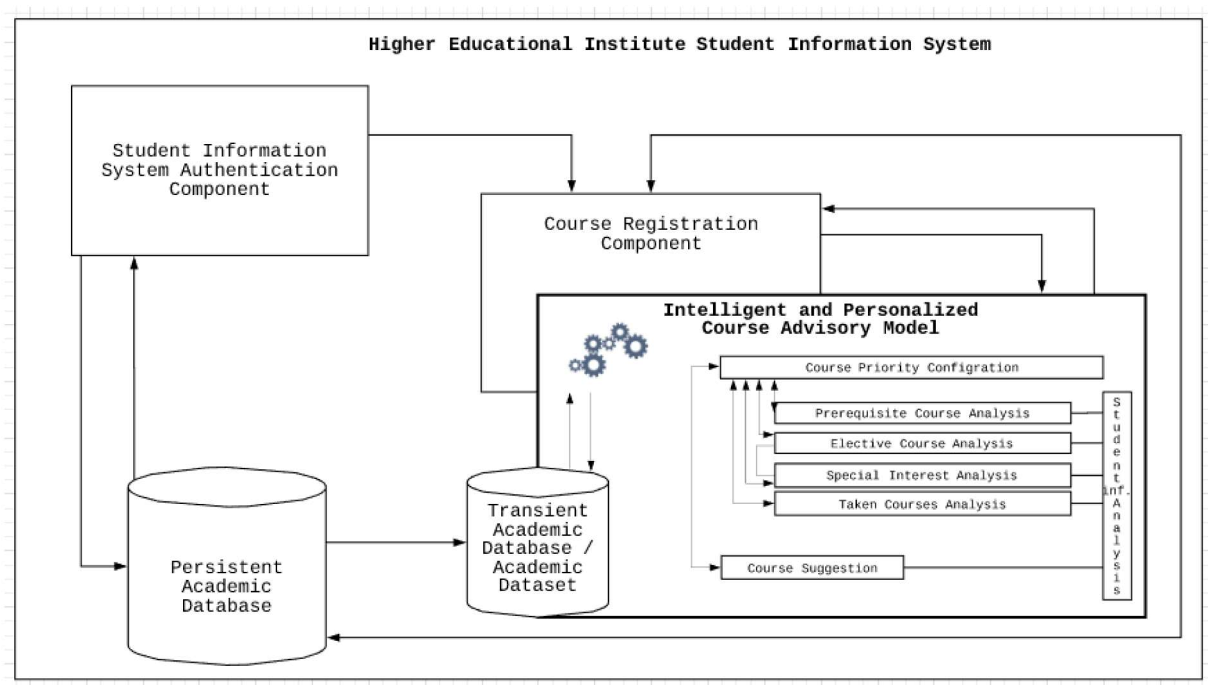


Figure 5.12: Architectural design of IPCAM

Academic dataset store transient data for course advisory model to auto generate courses as per the student's academic profile, which filtered form this data set. It is a collective information from SIS persistent database in which if there is any change in respective data in the main SIS database, it will automatically update itself.

SIS authentication component gives an interface for students to check their information in SIS persistent database. In many HEI SIS, students get their username and password while they register in to the system at the first semester. The student can access any SIS component through this component, so the student authentication result and retrieved student information send to course registration and course advisory components.

Course registration component expected to show us all course and lecturer detail information from SIS persistent database through different information retrieval technique. All those data should pass to course advisory model in order to filter information in intelligent manner. In this component academician, provide vote for field of interest (professional work in specific department) respective to the course. It provides an interface for course registration for specific semester in which students register to specific course after getting advising according to their academic history in course advisory component.

As we discussed in Chapter 4, we should select most appropriate algorithm for course advising requirements. Based on the discussion, the model development uses content-based recommendation system without distance measures, simple association rule and expert system with higher degree modification according to the course advisory model requirements.

5.3.2 Detail design

This section deals with the implementation part of a model and its sub-components regarding to architectural design. It has special focus towards components and their implementations through defining a logical structure of each components and their interfaces to communicate with other component. The output of this model design process is pseudo codes, detailed logic diagrams, process diagrams, and detailed description of requirements. The implementation of this model depends on all outputs mentioned above and HEIs flexibly integrate to their SISs.

IPCAM is has small size components in which they collaborate in structured manner in order to achieve the ultimate goal the model. As described in Figure 5.12, course priority configuration, taken course analysis, prerequisite course analysis, elective course analysis, course suggestion, personalization and academic term analysis components have their own specific purpose. Component diagram in Figure 5.13 provides a simplified, high-order view of an IPCAM in which all components support the usability and interchangeability. It also depicts how these components are organized and how they interact each other.

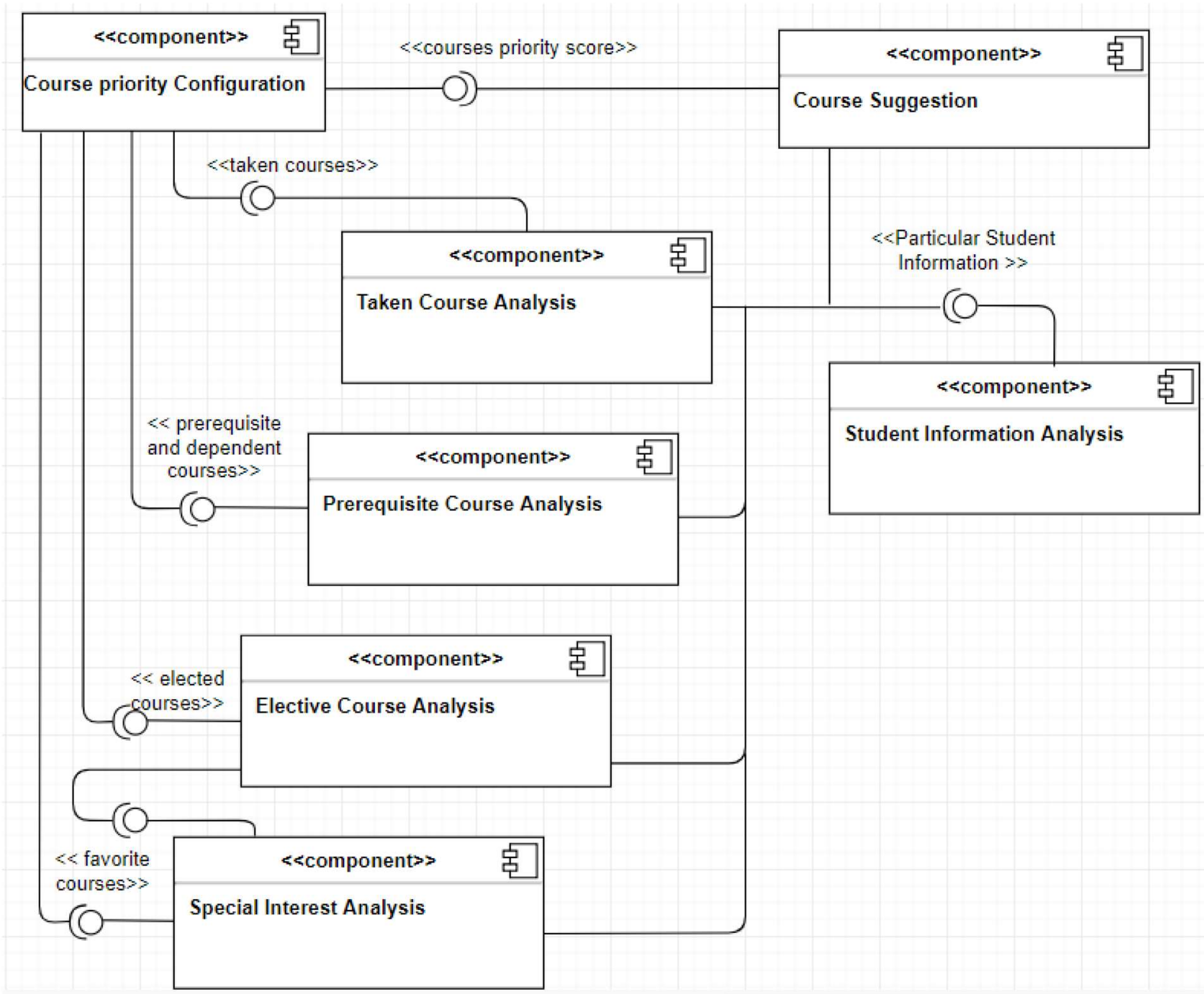


Figure 5.13: Component diagram for IPCAM

i. ***Student Information Analysis:*** *It* is a fundamental component to personalize course advising through analyzing and determining student-course relation depends on the data that get from student registration component of HEI's SIS. The outcome is a piece of core information for all other components, in which directly or indirectly interacted with this component in different contexts. It has ability to get the current term and student's batch from SIS component in customized manner, which helps to focus on the open courses and consider taken courses.

The following statements are a pseudocode for student information analysis, which is the implementation of an algorithm in the form of annotations and informative text written in plain English.

```
Start
declare Stud_info[], Inst_rule, stud_depart, depart_rule, stud_Semester, GPA
assign  Stud_info[] <-- registration.stud_info[]
assign Inst_rule <-- admin.HEI_rule
assign stud_depart = stud_info[indexOf(stud_depart)]
assign depart_rule <-- department.depart_rule
if Stud_info[indexOf(stud_semester)] == 1
   GPA = Stud_info[indexOf(GPA)]
data_frame = read_csv('student_course_dataset_name.csv')
End
```

ii. ***Taken course analysis:*** It enables to use different student information for filtering and analyzing registered course in the previous semesters. The student has to take grade for those courses according to his performance, which helps to determine whether the student passed, failed, withdrew, in completed, satisfied or unsatisfied courses based on HEI rule that set by the system administrator during system configuration. The outcome of this component identifies courses based on course status for a particular student then send to course priority configuration component for priority scoring.

The following statements are a pseudocode to analyze taken courses, which is the implementation of an algorithm in the form of annotations and informative text written in plain English.

```
Start
data_frame = read_csv('student_course_data_set_name.csv')
declare failed_course, passed_course, this_semester_open_course
failed_course = get(dataframe.course_status == [Inst_rule.failed_course])
passed_course = get(dataframe.course_status == [Inst_rule.passed_course])
this_semester_open_course = data_frame[(dataframe.Stud_ID == Stud_ID) &
```

```
(dataframe.status = open_course)]
if failed_course in data_frame
   failed_course = data_frame[(dataframe.Stud_ID == Stud_ID) &
(dataframe.Status = failed_course)
   if failed_course in this_semester_open_course
     send_to_all(failed_course)
   else
     send_to_course_priority_config(failed_course)
else
   passed_course = data_frame[(dataframe.Stud_ID == Stud_ID) &
(dataframe.Status = passed_course)
send_to_course_priority_config(passed_course)
End
```

iii. ***Prerequisite Analysis:*** It can easily identify prerequisite courses from other courses in order to determine courses that are dependent to the prerequisite and send to course priority configuration for setting priority score. This component especially focusses in two cases such as taken courses and unregistered courses. If a student failed to passed in prerequisite courses, other dependent courses in the department will send to course priority configuration to get the list priority score.

The following statements are a pseudocode for prerequisite course analysis, which is the implementation of an algorithm in the form of annotations and informative text written in plain English.

```
Start
declare prerequisite
assign prerequisite = get([course.type]== prerequisite)
get(failed_course)
assign this_semester_open_course = data_frame[(dataframe.Stud_ID == Stud_ID)
& (dataframe.status = open_course)]
assign failed_course_prerequisite = failed_course[(failed_course_type ==
prerequisite)]
```

```
assign dependent_course = this_semester_open_course.prerequisite_for ==
course_code
if prerequisite in this_semester_open_course
  send_to_course_priority_config(prerequisite)
send_to_course_priority_config(dependent_course)
End
```

iv. ***Elective Course Analysis:*** It enables to filter and analyze all elective courses under their job positions with respective rating in which academic experts gave the score in their own portal. Those academic experts shall have the way in SIS to give their vote for course and job position relation according to the current trends of technology, his knowledge, skill and experience. This component requires particular student information, open elective courses and field of special interest from student information analysis in order to deliver dependent courses for priority score configuration component as an output.

The following statements are a pseudocode for elective course analysis, which is the implementation of an algorithm in the form of annotations and informative text written in plain English.

```
Start
declare change_possible, change_interest,
assign elective_course = data_frame[(dataframe.type == elective)]
data_frame = read_csv('student_course_data_set_name.csv')
get(failed_course)
if elective_course in this_semester_open_course
  get(elective_course)
  goto Special_Interest_Analysis
if elective_course in failed_course
  get(failed_elective_course)
  if change_possible == "True"
    if change_interest == "True"
       goto Special_Interest_Analysis
```

```
    else goto FCEO
  else goto FCEO
FCEO: if failed_elective_course in open_course
      send_to_course_priority_config(failed_elective_course)
End
```

v. ***Special Interest Analysis:*** It is the smallest component of IPCAM, which receive currently open elective courses from elective course analysis component and students special interest (what do they want to be or What are they going to do when you graduate) and it delivered filtered information to use rating score that collected from academic experts. The following statements are a pseudocode for special interest analysis, which is the implementation of an algorithm in the form of annotations and informative text written in plain English.

```
Start
declare faculty,related_profession, student_interest, course_rating, mean_rating,
profession
assign SI_data_frame = read_csv('Special_Interest_data_set_name.csv')
assign related_profession = SI_data_frame[(SI_dataframe.faculty == faculty)]
display related_profession
display message << please select what you want  to be?>>
accept student_interest
group_by_profession(courses)
assign  profession = search(student_interest)
foreach course_code in profession
   for course_code = profession.course_code
       calculate mean_rate(course_rate)
       course = group_by_course()
       course.set(mean_rate)
return course
End
```

vi. ***Course priority configuration:*** It is one small component of the model, which mainly focus on giving priority order for courses depend on both the rule and regulations of HEI, and taken /registered courses. As mentioned in Section 5.3, Most HEIs have not the same rule and regulations, grading system as well as curriculum. This component compensates their differences through gathering specific and HEI wise information from other SIS components. Therefore, SIS administrator has an authority to set all the basic information that mentioned above in his SIS portal. This component receives necessary information from SIS in order to give general priority for prerequisite, mandatory and common courses Some course specially, those courses that student did not registered courses have default priority score according to his/her institute. There are other courses in which a particular student already taken in previous academic semesters. Based on the result of other IPCAM components, it set a priority score for each course.

The following statements are a pseudocode for course priority configuration, which is the implementation of an algorithm in the form of annotations.

```
Start
dataframe.set_priority_score(set(default_value))
assign data = get(course,stud_ID,inst_rule,depart_rule)
for  course in data
  data_frame.search(course)
  dataframe.set_priority_score(course,priority_value)
send_to_course(data_frame)
send_to_course_priority_config(data_frame)
End
```

vii. ***Course Suggestion:*** It comes after a series of process that explained above. In this component, there are two sub tasks prioritize and generate course suggestion for a particular student. Courses are prioritizing through arranging them in order of importance relative to each other. Then according to department plan and curriculum generate necessary and eligible course suggestion to guide a student to make a decision on courses.

The following statements are a pseudocode for course suggestion, which is the implementation of an algorithm in the form of annotations and informative text written in plain English.

```
Start
declare X, Y, sorted_elective_course, sorted_elective_course, sorted_course
get(course,stud_ID,inst_rule,depart_rule)
get(data_frame,)
goto Special_Interest_Analysis
assign x = depart_rule.num_course_semester
if mean_value in Special_Interest_Analysis.column is "True"
   for course in
      if Special_Interest_Analysis.course == data_frame.course
         dataframe.set_priority_score(course,0)
         sorted_elective_course = data_frame.sort(mean_rate, dec)
sorted_course = data_frame.sort(priority_score, dec)
display sorted_course
display sorted_elective_course
End
```

Information filtering deals with the delivery of information that the students is likely to explore the necessary courses depends on their academic history. Academic information filtering system assists students by filtering the data source and deliver necessary information to the students. When the delivered information comes in the form of suggestions, the system is called “a recommender system”. Due to student’s different interests and academic history, this system must filter information in personalized manner to accommodate the student’s interests.

CHAPTER 6
EXPERIMENT AND EVALUATION

6.1 Overview

Software evaluation can easily determine technical and developmental issues in addition to this, it also assesses the general usability and sustainability of a software. It works wisely to determine advantage, worth, and necessity of a subject using requirements regulated by a set of standards. The most vital benefit of software evaluation is to gain understanding into existing or previous software and to help identify future changes.

6.2 Software Evaluation Method

To know how far a software quality is measurement holds importance, the following concepts are briefly explained it.

1. The increasing of software customer's demand, the expectation of qualifying software product has grown concerning to a reliability.
2. A software application has complex structure, so software inspection and testing tasks become quite difficult. Therefore, software developers often classify software tasks in to deliverables and defining a reference point to mark the completion of one task.
3. If the tester or reviewer detect software errors in previous phases, developers should rectified them. Unless those bugs lead to error in the next consecutive phases which have impacts on the overall software product.

Software quality models act as a scheme for software attribute evaluation, which contributes to the quality of a software. They are adjust assessing methods or apprising a software product. Software designers are planning and developing new applications to meet the exponential changing requirements. High quality products lead to the contentment of users and signifies success of a software project. The potential to which the system meets various stockholders stated and intended needs. Quality model represents those stakeholders' requirements.

After understanding all characteristics of IPCAM, many software evaluation models view their internal behavior to get the best match for evaluating and measuring the model. Therefore, ISO-25010 software evaluation model chosen as the best fit for the inspection of IPCAM.

ISO / ICE 25010 describes a software quality in quality of product and operating quality. Operating quality contains five quality attributes that focus on the output of interaction when a software is active to do in a specific context of use (Miguel et al., 2014). ISO 25010 is mainly concerned with the quality model's definition, which used to clearly explain the desired quality of the software product, for software construction and software assessment. The product quality model for software assessment has eight exhaustive quality features. Each characteristic has sets of sub-attributes because of its wide scope. Those all the eight quality attributes are recapitulated in Appendix 1.

6.3 Experiment

NEU student registration process takes place in every fall, spring and sometimes summer semester with greater than 4,500 international and native students. NEU's young developers team discuss specifications and develop SIS system for the administrative purpose of the university. They had developed Einstein SIS for this university. Now, it is under development by Near East Technology org. maintaining the system through extending some functionalities and enhance the system quality. The Einstein SIS must provide many services to all NEU communities including students, academic personals, management staffs and development team from maintenance teams to senior managers. Einstein SIS also has ability to overcome academic administration difficulties, such as student admission, academic, dormitory and cafeteria management, financial management, etc. According to statistical data obtained from the Einstein system administrator, the system is accessed approximately 6,000 times per day (Almassri, 2003).

The experiment performed in course advising system of NEU's Einstein to guide student for selecting suitable courses during the starting of every semester. IPCAM helps students to select appropriate, suitable courses according to department's curriculum and institution's rule and regulation. IPCAM is used NEU Einstein's course registration system to take student and course information then send recommended courses for student approval or decisions.

The system requires information from the department head in which they should provide by using their portal. This system also accesses information from the SIS database and student portal as CSV file and plain text format respectively.

NEU uses the same grading scale to Turkey universities, which depicted in Table 6.1. Status is depending on the grade what he/she got in the previous semester. If the previous semester GPA is smaller than 2.0, then the student should readmit courses which have grade CC, DC, and FD (Amassri, 2003).

Einstein's course registration system provides an interface for IPCAM and collect various data from different stack holders. Those are:

- Department head specifies the highest possible course and elective courses that allowed for registration and list of open courses that students are going to register.
- Course registration system give various information about term (spring, fall or summer), current academic year, ID, Student's GPA.
- Students should enter a special interest to get elective courses recommendation.
- Academicians rate for profession and course relationship, 5 for highest and 0 for lowest priority.

This Einstein's course registration system also should accept IPCAM's output for supporting student's process to make course choice.

Table 6.1: NEU grading system and failed courses specifications

Grade	Scale (In Percentage)	GPA	Status
AA	90 – 100	4	Pass
BA	85 – 89	3.5	Pass
BB	80 – 84	3	Pass
CB	75 - 79	2.5	Pass
CC	70 - 74	2	Pass (if GPA < 2.0 fail)
DC	65 - 69	1.5	Pass (if GPA < 2.0 fail)
DD	60 - 64	1	Pass (if GPA < 2.0 fail)
FD	50 – 59	0.5	Fail
FF	49 and below	0	Fail
W	-	0	Withdrawal / Fail
U	-	-	Unsatisfactory / Fail
S	-	-	Satisfactory / Fail
I	-	0	Incomplete / Fail
P	-	-	Progress / Fail

6.3.1 Course Priority Hierarchy

Courses priority is the backbone for IPCAM, this system without priority score and rating could not suggest courses. IPCAM automatically set the configuration of course priority setting based on course type, student's grade status and term. The failed courses have higher priority score than unregistered courses which means if a student fail to pass in a specific course, he/she have to take failed courses first so the system prioritize them according to the previous grade.

Below Figure 6.1 depicted a priority score configuration hierarchy for registered courses. When we compare in the major classification, the prerequisite courses have highest priority than the others. The range for priority score is between zero and ten. Example: If the course's priority score is low, the course will not be recommended. If the course's priority score is high, it is more relevant than the other courses.

Below Figure 6.2 depicted a priority score configuration hierarchy for semester open courses and unregistered courses. The range for priority score is between zero and ten. Example: If the course's priority score is low, the course will not be recommended. If the course's priority score is high means it is more relevant than the other courses.

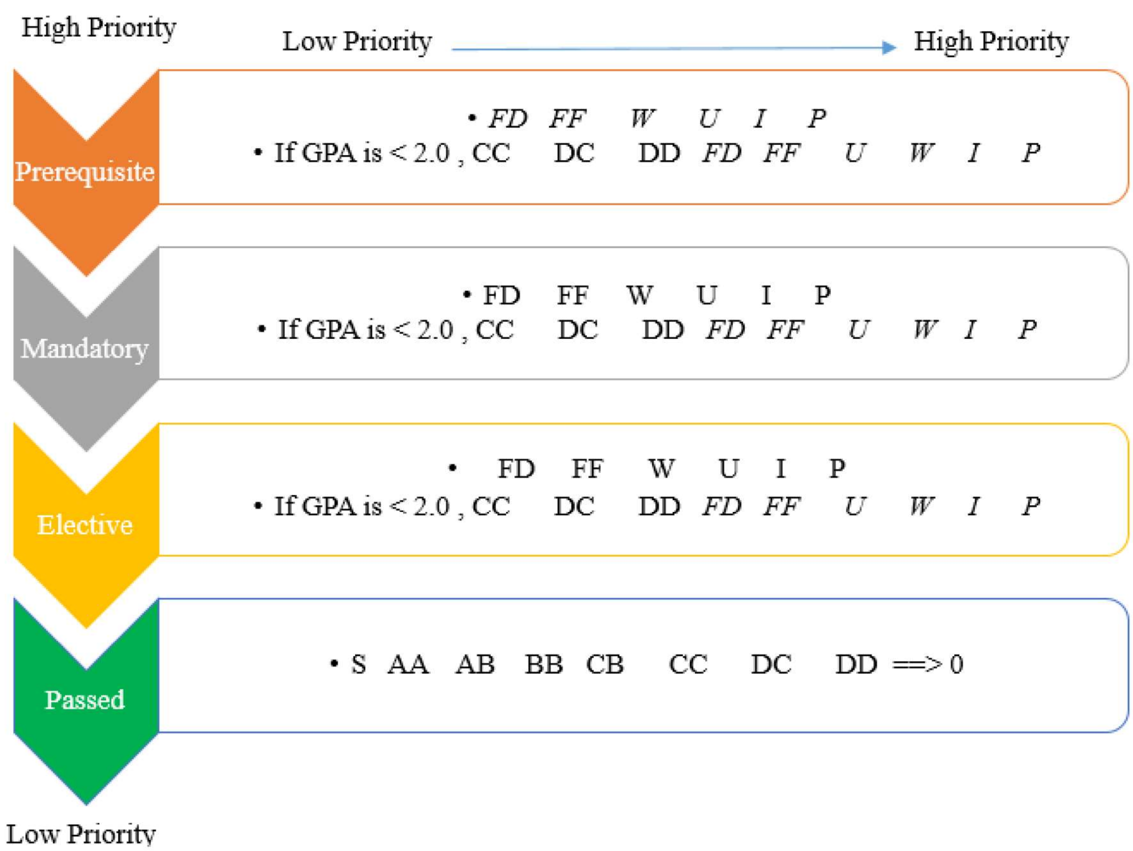


Figure 6.1: Course priory hierarchy for registered course

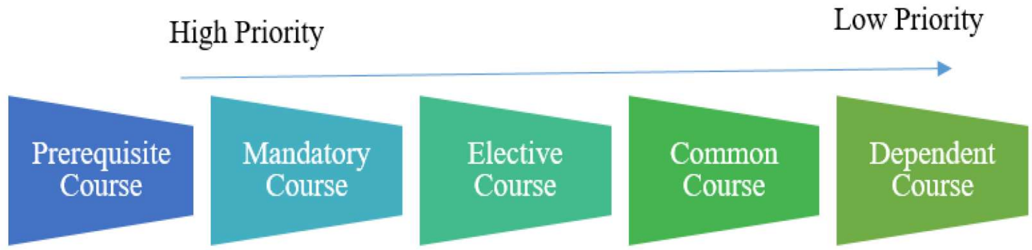


Figure 6.2: Priority score configuration hierarchy for unregistered courses

6.3.2 Experimental dataset

IPCAM uses datasets to avoid directly communication with the SIS database and to hold multiple data tables.

After a serious discussion and analysis of Einstein system database structure, we establish two datasets for course filtering and recommendation according to student's academic history and analyzing and suggesting elective course rating based on student's special interest.

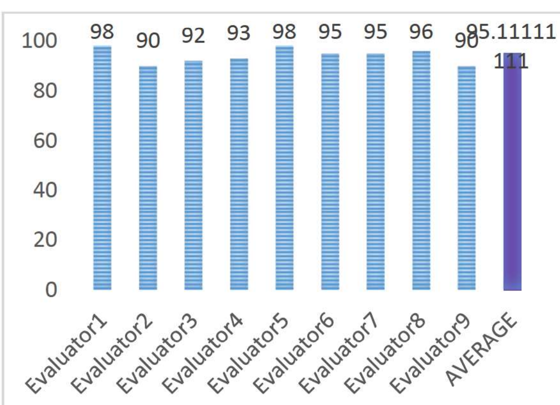
i. ***General course dataset*:** It is an academic dataset, which consist of three selected departments of applied science faculty students to provide an intelligence and personalized course suggestion according to their interest. Those randomly selected departments are software engineering, information science and computer science departments. The dataset consists 1221*3 rows with 11 visible and 1 invisible column to represent 5 students from each batch with their respective courses for fall semester.

ii. ***Elective course dataset*:** It includes elective courses with the rating of their corresponding specialist. Professional experts or academicians have different skills and knowledge to decide which courses are good in specific area. Even the rating is not statics because it depends on the current technology trends. Therefore, they have to give rating score on course – profession relation by using their portal in Einstein system in order to suggest electives courses to achieve student's specialization goal. This experimental dataset consists of 9 professions, 20 courses and 1000 rating values.

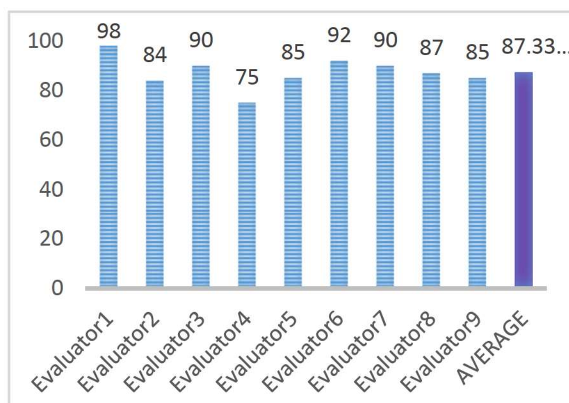
To ensure overall acceptance of the model, students from software engineering, information science and computer science department of Near East University were selected as evaluator and allowed to assess the system by either volunteering their own transcript or using sample transcript of IPCAM.

6.3.3 Results

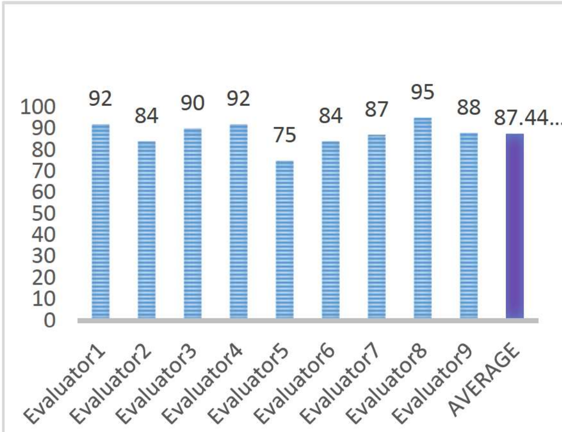
According to the obtained results from the evaluators show that IPCAM has satisfied 87.33% performance efficiency, 95.11% functional suitability, 87.44 % usability, 90.11% compatibility, 92.89% reliability, 82.78% portability, 92.56% security and 86% maintainability. It generalizes that IPCAM has high functional suitability, reliability and security. Figure 6.3 depicted the individual evaluation result of NEU course advising model under ISO 25010 quality characteristics per evaluators and Figure 6.4 represented the quality features that characterized the overall evaluation result in a percentage.



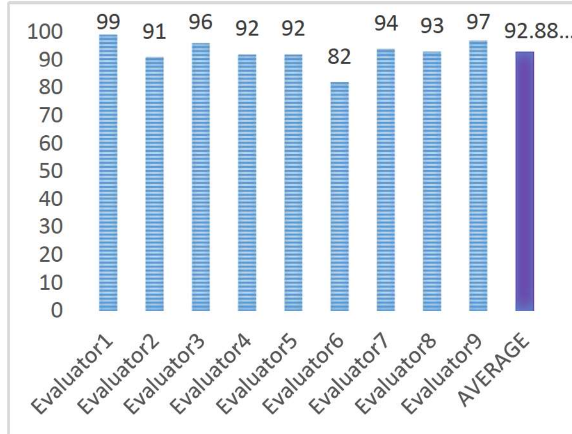
(a) Functional suitability



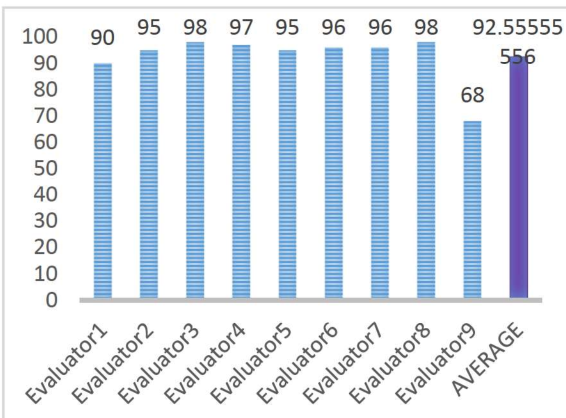
(b) Performance efficiency



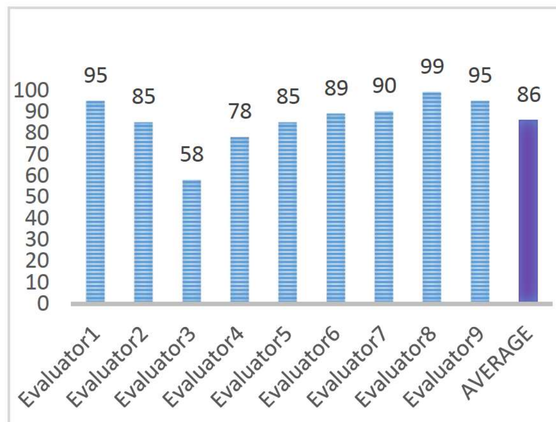
(c) Usability



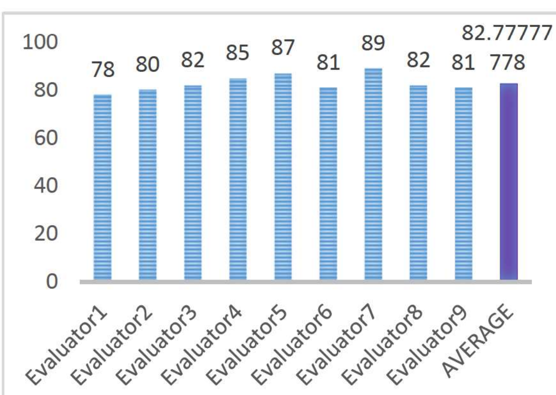
(d) Reliability

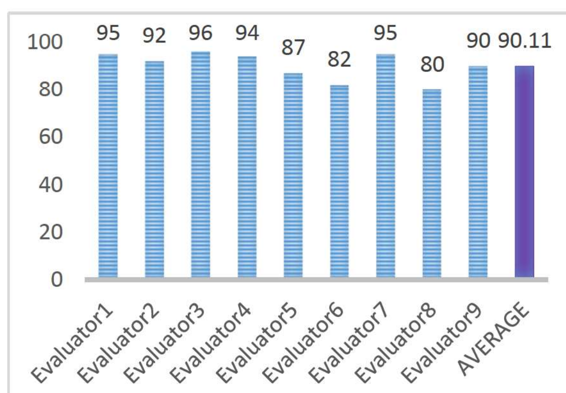


(e) Security



(f) Maintainability





(g) Portability

(h) Compatibility

Figure 6.3: General evaluation result of IPCAM in three departments of NEU using ISO-25010 software quality attributes per evaluators

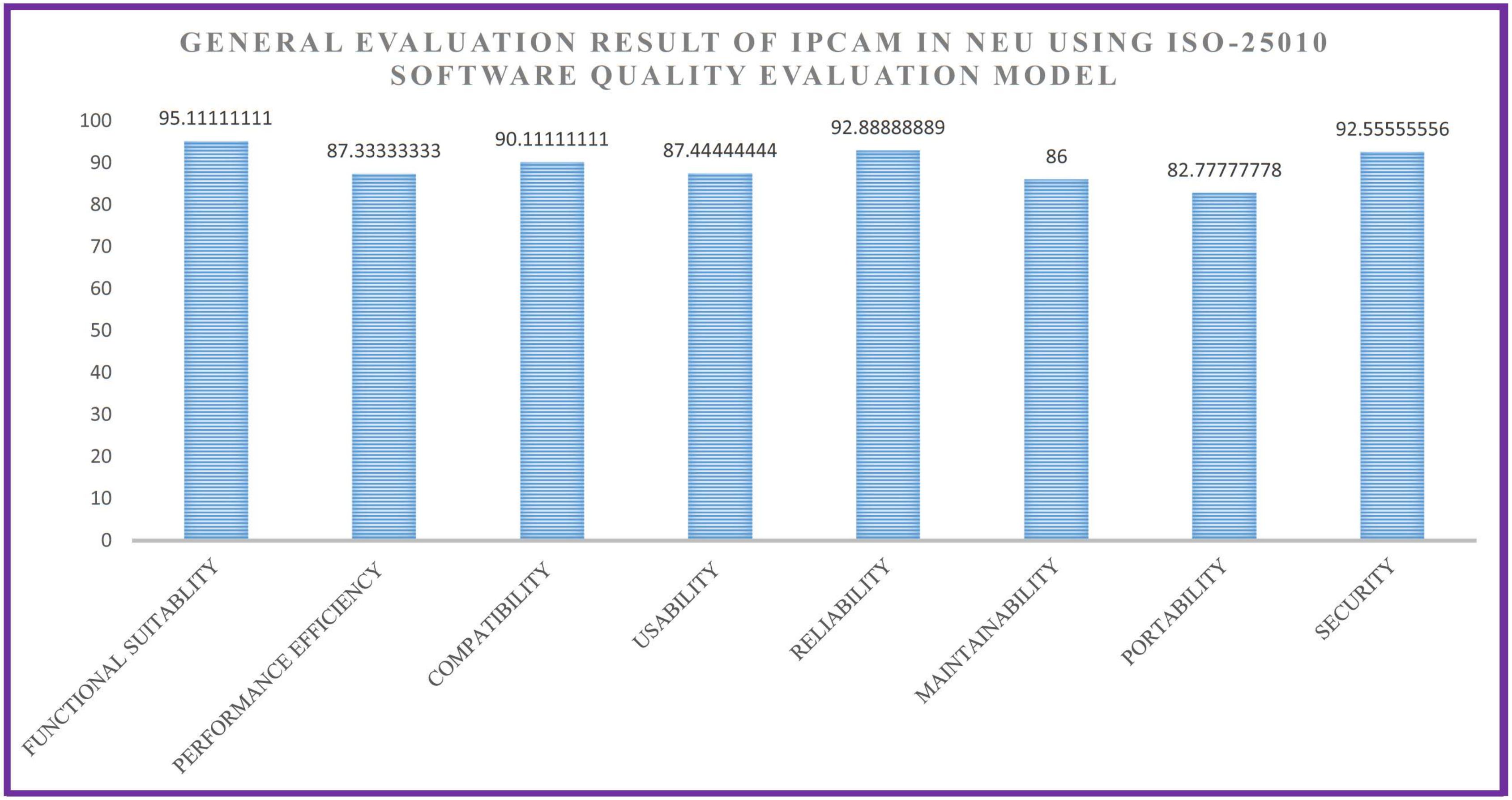


Figure 6.4: Grand evaluation result of IPCAM in NEU using ISO-25010 Quality Model

CHAPTER 7
CONCLUSIONS AND FUTURE WORKS

7.1 Conclusions

An Advisory system supports decision making through providing guide and help different users with different real-life problems in which human experts normally do it. Those experts do not make a decision rather they give advice as a guide in order to supply the best solution for decision makers or advisees. IPCAM is a type of advisory system which guide students during course registration time by suggesting appropriate courses based on their academic history and special interest that what they want to be.

The drawbacks of existing course advising system and limitations in its development including reusability, maintainability, usability, personalization and intelligence were force to develop IPCAM. This paper mainly targeted on existing academic advising systems in various HEIs, related works on course advising problems, clarification of possible and popular advising algorithms, development of effective and highly portable IPCAM, evaluation of the model through ISO 25010 quality-model in NEU. IPCAM plays principal role for students, academicians, software development organizations, universities and countries to suggest appropriate courses for students based on their academic history, special interest profession and regarding HEI's rules and regulations. This model is highly suited to the SIS specifically, student course registration module. It makes the advisor and advisee to use their knowledge effectively and efficiently by delivering the best solution that saves their time and money.

7.2 Future Works

The future works shall focus on two basic targets. The first one is the evaluation of IPCAM by using various effective, standard and appropriate software evaluation and testing techniques via integrating it with various HEI's SIS. The second one is IPCAM enhancement through updating the entire design and algorithm as per the emerging technologies and various HEI's requirements. It shall also focus on adding new functionalities and developing user interface as an add-on for HEIs' SIS.

REFERENCES

Aamodt, A., & Plaza, E. (1994). Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI communications*, 7(1), 39-59.

Abdelhamid, Y., Ayoub, A., & Alhawiti, M. (2015). Agent-based intelligent academic advisor system. *International Journal of nAdvanced Computer Technology*, 4(2), 1-6.

Alhir, S. A. (2003). *Learning UML*. Sebastopol, CA: OReilly.

Al-Ahmar, M. A. (2011). A prototype student advising expert system supported with an object-oriented database. *International Journal of Advanced Computer Science and Applications (IJACSA), Special Issue on Artificial Intelligence*, 1(3), 100-105.

Al-Ghamdi, A., Al-Ghuribi, S., Fadel, A., & AL-Ruhaili, F. A. A. T. (2012). An expert system for advising postgraduate students. *International Journal of Computer Science and Information Technologies,* 3(3), 4529-4532.

Al-Nory, M. T. (2012, August). Simple Decision Support Tool for university academic advising. *In Information Technology in Medicine and Education (ITME), 2012 International Symposium* on (Vol. 1, pp. 53-57). IEEE.

Almassri, M. (2003). Development of facultys' students information system. *Graduate Project,Near East University,* TRNC Mersin 10, Turkey.

Aslam, M. Z., & Khan, A. R. (2011). A proposed decision support system/expert system for guiding fresh students in selecting a faculty in Gomal university, Pakistan. *arXiv preprintarXiv:1104.1678.*

Beemer, B. A., & Gregg, D. G. (2008). *Advisory Systems to Support Decision Making.* Handbook on Decision Support Systems 1, 511-527.doi:10.1007/978-3-540-487135_24

Borgatti, S. (2007). Distance and Correlation Retrieved April 03, 2019, from http://www.analytictech.com/mb876/handouts/distance_and_correlatio.htm

Brubacher, J. S. & Rudy, W. (1997). *Higher Education in Transition: A History of American Colleges and Universities* (4th ed.). New Brunswick, N.J. : Transaction.

Dalinina, R. (2017, January 31). Introduction to Correlation. Retrieved April 02, 2019, from https://www.datascience.com/blog/introduction-to-correlation-learn-data-science-tutorials.

Daramola, O., Emebo, O., Afolabi, I. and Ayo, C. (2014). Implementation of an Intelligent Course Advisory Expert System. *International Journal of Advanced Research in Artificial Intelligence*, 3(5).

Dash, P., & Vaidhehi, V. (2017). Enhanced Elective Subject Selection for ICSE School Students using Machine Learning Algorithms. *Indian Journal of Science and Technology*, *10*(21), 1–10. https://doi.org/10.17485/ijst/2017/v10i21/109551

Deorah, S., Sridharan, S., & Goel, S. (2010, February). SAES-expert system for advising academic major. *In Advance Computing Conference (IACC), 2010 IEEE 2nd International* (pp. 331-336). IEEE.

Donges, N. (2018, February 22). The Random Forest Algorithm. Retrieved April 22, 2019 from https://towardsdatascience.com/the-random-forest-algorithm-d457d499ffcd

Engin, G., Aksoyer, B., Avdagic, M., Bozanlı, D., Hanay, U., Maden, D., & Ertek, G. (2014). Rule-based expert systems for supporting university students. *Procedia Computer Science*, 31, 22-31.

Feghali, T., Zbib, I., & Hallal, S. (2011). A Web-based Decision Support Tool for Academic Advising. *Journal of Educational Technology & Society*, 14(1).

Feigenbaum, E. A. (1981). *Expert systems in the 1980s. State of the art report on machine intelligence*. Maidenhead: Pergamon-Infotech.

Gallagher, P. J. & Demos, G. D. (Eds.). (1983). *Handbook of Counseling in Higher Education*. New York : Praeger.

Gordon, V. N. (1992). *Handbook of Academic Advising*. Westport , CT : Greenwood Press.

Gordon, V. N., Habley, W. R. & Associates. (2000). *Academic advising. A comprehensive handbook*. San Francisco: Jossey-Bass.

Harris, T. A. (2018). Prescriptive vs. Developmental: Academic Advising at a Historically Black University in South Carolina. *The Journal of the National Academic Advising Association*, 38(1), 36-46. https://doi.org/10.12930/nacada-15-010

Harrison, M., & Prentiss, M. (2016). *Learning the Pandas Library: Python Tools for Data Munging, Analysis, and Visual*. Create Space Independent Publishing Platform.

Henderson, L. K., & Goodridge, W. (2015). AdviseMe: An intelligent web-based application for academic advising. *International Journal of Advanced Computer Science and Applications*, 6(8), 233-243.

Hingorani, K., & Askari-Danesh, N. (2014). Design and development of an academic advising system for improving retention and graduation. *Issues in Information Systems*, 15(2), 344-349.

How do Recommendation Engines work? and What are the Benefits? (2019, March 27). Retrieved May 29, 2019, from https://www.marutitech.com/recommendation-engine-benefits/

Huang, T., Zhan, G., Zhang, H., & Yang, H. (2016). MCRS: A Course Recommendation System for MOOCs. *Proceedings of 2017 International Conference on Advanced Technologies Enhancing Education (ICAT2E 2017)*. doi:10.2991/icat2e-17.2016.20

Hwang, G. J., & Chang, H. F. (2011). A formative assessment-based mobile learning approach to improving the learning attitudes and achievements of students. *Computers & Education,* 56(4), 1023-1031

ISO/IEC 25010. (n.d.). Retrieved April 03, 19, from https://iso25000.com/index.php/en/iso-25000-standards/iso-25010

Ishak, I. B., & Lehat, M. L. B. (2012, June). A conceptual framework of web-based academic advisory information system. *In 2012 IEEE Symposium on Humanities, Science and Engineering Research* (pp. 957-961). IEEE.

Jain, R. (2019, May 20). A beginner's tutorial on the apriori algorithm in data mining with R implementation. Retrieved from https://www.hackerearth.com/blog/developers /beginners -tutorial-apriori-algorithm-data-mining-r-implementation/

Kavyashri. D.(2018, March 15). Mining Frequent itemsets - Apriori Algorithm. Retrieved from http://dwgeek.com/mining-frequent-itemsets-apriori-algorithm.html/

Koehrsen, W. (2018, August 30). An Implementation and Explanation of the Random Forest in Python. Retrieved from https://towardsdatascience.com/an-implementation-and explanation-of-the-random-forest-in-python-77bf308a9b76

Komives, S. R., Woodard, Jr., D. B., & Associates. (1996). *Student Services: A Handbook for the Profession* (3rd ed.) San Francisco: Jossey-Bass.

Laghari, M. S. (2014). Automated course advising system. *International Journal of Machine Learning and Computing*, 4(1), 47.

Laghari, M. S., Al Habsi, S. A. A., Maaz, N. A., & Al Naqbi, M. A. A. (2015). A one-semester course planner for EE students. *International Research Journal of Electronics and Computer Engineering,* 1(1), 13-18.

Levinas, C. A. (2014). An analysis of memory based collaborative filtering recommender systems with improvement proposals Master's thesis, Universitat Politècnica de Catalunya.

Luo, S. (2018, December 10). Intro to Recommender System: Collaborative Filtering. Retrieved from https://towardsdatascience.com/intro-to-recommender-system collaborative-filtering-64a238194a26

Miguel, J. P., Mauricio, D., & Rodríguez, G. (2014). A review of software quality models for the evaluation of software products. *arXiv preprint arXiv:1412.2977*.

Moogan, Y. J., Baron, S., & Scribner, E. (2003). The Journal of Higher Education,.*Journal of Further and Higher Education*, *27*(3), 26–27 https://doi.org/10.1080/03098770320 0098699

Muhammad, Z., & Abdur, K. (2011). A proposed decision support system/expert system for guiding fresh students in selecting a faculty in gomal university, pakistan. *arXiv preprint arXiv:1104.1678, 1(4), 33–40.*

NACADA. Concepts of academic advising (2018). Retrieved March 25, 2019, from the Kansas State University website: https://www.nacada.ksu.edu/Resources/Pillars/Concept.aspx

Negnevitsky, M. (2011). *Artificial intelligence: A guide to intelligent systems*. Harlow, England: Addison Wesley.

Nambiar, A. N., & Dutta, A. K. (2010, September). Expert system for student advising using JESS. *In 2010 International Conference on Educational and Information Technology* (Vol. 1, pp. V1-312). IEEE

Needham, M., & Hodler, A. E. (2019). *Graph Algorithms: Practical Examples in Apache Spark and Neo4j*. O'Reilly Media.

Nielsen, J., & Molich, R. (1990, March). Heuristic evaluation of user interfaces. *In Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 249-256). ACM.

Ray, S. (2019). Six easy steps to learn Naive Bayes algorithm (with code in python). Retrieved April 12, 2019 from https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/

Ricci, F., Rokach, L., & Shapira, B. (2011). Introduction to recommender systems handbook. *In Recommender systems handbook* (pp. 1-35). Springer, Boston, MA.

Rudolph, F. (1990). *The American College and University:* A History. Athens, GA : University of Georgia Press.

Shatnawi, R., Althebyan, Q., Ghalib, B., & Al-Maolegi, M. (2014). Building A Smart Academic Advising System Using Association Rule Mining. *arXiv preprint-arXiv:1407.1807*.

Schenider, D., Sasso, P., & Puchner, L. (2017). Adviser and faculty perceptions of the benefit and feasibility of intrusive advising. *The Mentor: Innovative Scholarship on Academic Advising, 19*.

Schwarzkopf, E. (2012, June 12). User-Based and Item-Based Collaborative Filtering Retrieved May 28, 2019, from http://ericschwarzkopf.de/?p=12

Varney, J. (2012). Proactive (intrusive) advising. *Academic Advising Today*, 35(3),13.

Umamaheswaran, V. (2018, November 14). Comprehending K-means and KNN Algorithms Retrieved March 2, 2019, from https://becominghuman.ai/comprehending-k-means-and-knn-algorithms-c791be90883d?gi=9ac0e4e64ee8

Ruggieri, S. (2002). Efficient C4. 5 [classification algorithm]. *IEEE transactions on knowledge and data engineering*, 14(2), 438-444.

What is Machine Learning (2017, October 05). Retrieved April 23, 2019 from https://www.expertsystem.com/machine-learning-definition/

Wu, X., Kumar, V., Ross, Q. J., Ghosh, J., Yang, Q., Motoda, H., … Steinberg, D. (2008). *Top 10 algorithms in data mining*. *Knowledge and Information Systems* (Vol. 14). https://doi.org/10.1007/s10115-007-0114-2

APPENDICES

APPENDIX 1

ISO-25010 Quality Characteristics Form

Give evaluation score 5 for Very high, 4 for high, 3 for moderate, 2 for low and 1 for very low.

Quality Attribute	Score	Description
Functional Suitability		Degree to which the set of functions works all the expected tasks and user objectives
		Degree to which a product or system provides the accurate results with the needed degree of precision.
		Degree to which the functions facilitate the accomplishment of specified tasks and objectives.
Performance Efficiency		Degree to which the response and processing times and throughput rates of a product or system, when performing its functions, meet requirements.
		Degree to which the amounts and types of resources used by a product or system, when performing its functions, meet requirements.
		Degree to which the maximum limits of a product or system parameter meet requirements.
Compatibility		Degree to which a product can perform its required functions efficiently while sharing a common environment and resources with other products, without detrimental impact on any other product.
		Degree to which two or more systems, products or components can exchange information and use the information that has been exchanged.
Usability		Degree to which users can recognize whether a product or system is appropriate for their needs.
		Degree to which a product or system can be used by specified users to achieve specified goals of learning to use the product or system with effectiveness, efficiency, freedom from risk and satisfaction in a specified context of use.
		Degree to which a product or system has attributes that make it easy to operate and control.

Table continued

Quality Attribute	Number	Description
Usability		Degree to which a system protects users against making errors.
		Degree to which a user interface enables pleasing and satisfying interaction for the user.
		Degree to which a product or system can be used by people with the widest range of characteristics and capabilities to achieve a specified goal in a specified context of use.
Reliability		Degree to which a system, product or component meets needs for reliability under normal operation.
		Degree to which a system, product or component is operational and accessible when required for use
		Degree to which a system, product or component operates as intended despite the presence of hardware or software faults.
		Degree to which, in the event of an interruption or a failure, a product or system can recover the data directly affected and re-establish the desired state of the system.
Maintainability		Degree to which a system or computer program is composed of discrete components such that a change to one component has minimal impact on other components.
		Degree to which an asset can be used in more than one system, or in building other assets
		Degree of effectiveness and efficiency with which it is possible to assess the impact on a product or system of an intended change to one or more of its parts, or to diagnose a product for deficiencies or causes of failures, or to identify parts to be modified.
		Degree to which a product or system can be effectively and efficiently modified without introducing defects or degrading existing product quality.
		Degree of effectiveness and efficiency with which test criteria can be established for a system, product or component and tests can be performed to determine whether those criteria have been met.

Table continued

Quality Attribute	Number	Description
Maintainability		Degree to which a product or system can effectively and efficiently be adapted for different or evolving hardware, software or other operational or usage environments.
Portability		Degree of effectiveness and efficiency with which a product or system can be successfully installed and/or uninstalled in a specified environment.
		Degree to which a product can replace another specified software product for the same purpose in the same environment.
Security		Degree to which a product or system ensures that data are accessible only to those authorized to have access.
		Degree to which a system, product or component prevents unauthorized access to, or modification of, computer programs or data.
		Degree to which actions or events can be proven to have taken place, so that the events or actions cannot be repudiated later.
		Degree to which the actions of an entity can be traced uniquely to the entity.
		Degree to which the identity of a subject or resource can be proved to be the one claimed.

APPENDIX 2

RANDOMLY SELECTED NEU ACADEMIC HISTORY DATASET

Stud_ID	Department	Course_Code	CrHr	ECTS	Course_Type	Status	Year	Semester	Prerequisite_For
20191111	Software_Engineering	Math_001	3	6	Common	Open_Course	1st_year	fall	None
20191112	Software_Engineering	SEng003	3	6	Prerequisite	Open_Course	1st_year	fall	SEng114
20171403	Software_Engineering	SEng300	4	7	Prerequisite	DD	2nd_year	fall	SEng410
20191107	Software_Engineering	SEng001	3	6	Mandatory	AA	1st_year	spring	None
20191111	Software_Engineering	Stat_040	3	6	Common	Open_Course	1st_year	fall	None
20171303	Software_Engineering	SEng312	3	6	Mandatory	BA	2nd_year	spring	None
20191110	Software_Engineering	Eng_002	3	7	Prerequisite	Open_Course	1st_year	fall	None
20161605	Software_Engineering	Stat_040	3	6	Common	BA	1st_year	spring	None
20161605	Software_Engineering	SEng001	3	6	Mandatory	BA	1st_year	spring	None
20161605	Software_Engineering	SEng900	4	7	Prerequisite	FF	4th_year	spring	SEng901
20161602	Software_Engineering	SEng505	3	5	Elective	BC	3rd_year	fall	None
20170303	Computer_Engineering	Ele	-1	-1	Elective	Open_Course	3rd_year	fall	None
20160504	Computer_Engineering	Co_020	3	5	Elective	CC	3rd_year	fall	None
20160505	Computer_Engineering	Co_205	3	6	Mandatory	CC	2nd_year	fall	None
20160504	Computer_Engineering	Co_020	3	5	Elective	CC	3rd_year	fall	None
20170304	Computer_Engineering	Co_312	3	6	Mandatory	DD	2nd_year	spring	None
20170305	Computer_Engineering	Co_102	3	6	Mandatory	DD	2nd_year	fall	None
20170305	Computer_Engineering	Co_300	4	7	Prerequisite	DD	2nd_year	spring	Co_410

Table Continued

Stud_ID	Department	Course_Code	CrHr	ECTS	Course_Type	Status	Year	Semester	Prerequisite_For
20192207	Information_System_ Engineering	ISE004	3	6	Prerequisite	AA	1st_year	spring	ISE114
20192207	Information_System_ Engineering	ISE001	3	6	Mandatory	AA	1st_year	spring	None
20172305	Information_System_ Engineering	ISE102	3	5	Mandatory	FF	1st_year	spring	None
20172401	Information_System_ Engineering	ISE003	3	6	Mandatory	FF	1st_year	spring	None
20192213	Information_System_ Engineering	Eng_001	3	6	Common	Open_Course	1st_year	fall	None
20192214	Information_System_ Engineering	Math_001	3	6	Common	Open_Course	1st_year	fall	None
20182103	Information_System_ Engineering	ISE212	3	6	Mandatory	Open_Course	2nd_year	fall	None
20182104	Information_System_ Engineering	ISE101	3	5	Mandatory	Open_Course	2nd_year	fall	None
20182104	Information_System_ Engineering	ISE202	4	6	Prerequisite	Open_Course	2nd_year	fall	ISE312
20192215	Information_System_ Engineering	ISE003	3	6	Mandatory	Open_Course	1st_year	fall	None

APPENDIX 3

RANDOMLY SELECTED NEU ELECTIVE COURSES TO PROFESSIONS RATING DATASET

Inst_ID	Course_Code	Special_Interest	Rate
Inst_0010	Co_001	Computer and Information Research Scientists	4
Inst_0004	Co_001	Information Security Analyst	2
Inst_0001	Co_001	Computer Network Architect	3
Inst_0008	Co_001	System_Designer	2
Inst_0004	Co_001	Software Developer	3
Inst_0014	Co_002	Computer Hardware Engineer	0
Inst_0009	Co_013	Software Designer	3
Inst_0003	Co_013	Computer Hardware Engineer	4
Inst_0010	Co_013	Web_Programmer	2
Inst_0017	Co_014	Computer and Information Research Scientists	0
Inst_0006	Co_014	Information Security Analyst	5
Inst_0012	Co_001	Web_Programmer	4
Inst_0001	Co_002	Computer and Information Research Scientists	1
Inst_0003	Co_002	Information Security Analyst	5
Inst_0008	Co_002	Computer Network Architect	4
Inst_0020	Co_002	Software Developer	4
Inst_0003	Co_002	Robotics	3
Inst_0020	Co_002	Information Security Analyst	4
Inst_0011	Co_002	System_Designer	3
Inst_0020	Co_002	Information Security Analyst	4
Inst_0017	Co_002	Software Designer	3