Student Performance Predictive Model

Mitigating Students' Performance Gap In Outcome-Based Education Systems Using Mathematical Model

A Case Study

By

Vuong Tan Dat

Dissertation in partial fulfillment of
the requirement for the
Bachelor of Technology (Hons)
(Business Information Systems)

Jan 2012

Universiti Teknologi PETRONAS
Bandar Sri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

Student Performance Predictive Model

Mitigating Students' Performance Gap In Outcome-Based Education Systems Using Mathematical Model

A Case Study

By

Vuong Tan Dat

A Project dissertation submitted to the
Information Technology Programme
Universiti Teknologi PETRONAS
in partial fulfillment of the requirement for the
BACHELOR OF TECHNOLOGY (Hons)
(BUSINESS INFORMATION SYSTEMS)

Approved by,

(Ms. Chen Yoke Yie)

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK
Jan 2012

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own as specified in the references and acknowledgement, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

____

VUONG TAN DAT

ABSTRACT

Outcome-Based Education (OBE) model is a recurring modern means for education reform - a process of improving public education. It embodies the idea that best educational practice is to determine the end goals, or "outcomes", before the strategies, processes, techniques, and other means can be put into place to achieve them. While applications of OBE model have been continuously expanding and improving, "performance gap" - the gap between what students can do and what they are expected to do - still hinders its potential benefits. Mitigating this gap is among priority tasks of educators to achieve long-term goals of educational reform; and developing student performance predictive models is one way to approach this problem.

Most previous studies had targeted big scope of a long-term prediction and most had used various range of educational settings as their inputs, including students' demographic profiles and behavioral contents. They had applied different techniques in order to predict students' academic performance; however, due to the nature of these inputs, all had adopted complex data mining models. This project, instead, was purposely narrowed down to short-term programming courses at Universiti Teknologi PETRONAS (UTP), Malaysia. Main purpose was to design a functioning short-term predictive model which continuously assisted lecturers to analyze patterns and to accurately predict students' upcoming performance and final result in order to provide timely intervention and adjustment. The writer introduced a unique approach by focusing on a simplified set of inputs including (1) students' coursework breakdown and (2) users' dynamically subjective inputs. Instead of complex data mining models, a straightforward mathematical model was developed and was highly customized to best utilize those inputs, which resulted in a high level of accuracy for predictive outputs. A fully developed system from the testing prototype promises to serve as a relatively convenient tool for UTP lecturers to utilize simple yet richly informative coursework data into predicting students' performance, then mitigating the performance gap and ultimately achieving set objectives of UTP's OBE system.

ACKNOWLEDGEMENT

This project would not be a successful one without the help and support from many individuals and organizations. Thus, the writer would like to show his deepest appreciation to the people who were involved in ensuring the completion of this project.

First and foremost, the writer would like to express his appreciation and grateful thanks to his supervisor, Ms. Elaine Chen Yoke Yie, for taking time off her busy schedule to monitor his progress throughout the duration of the project. Her constructive guidance, advice and continuous supports had led to the final outcome of this project.

Second, the writer would like to extend the thanks to his internal examiner and also Head of Department, Dr. Mohd Fadzil Bin Hassan who had provided valuably fundamental ideas and background of study to the writer in carrying the project. The thought and knowledge shared by him was truly appreciated.

Third, the writer would like to express his gratitude toward his colleagues for supporting him in carrying and fulfilling the requirements of this project, especially for willingly participating in data gathering phase.

Finally, the writer would like to thank his family and friends for all the supports they gave when he faced problems during the project.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Background of Study

Outcome-Based Education (OBE) model is a recurring modern means for education reform - a process of improving public education. As compared to traditional model, OBE model drives its focus on student performance rather than on available resources to students in provided learning environment. Adopting student-centered learning philosophy, OBE involves restructuring of curriculum, assessment and reporting practices to reflect the achievement of high order learning and mastery rather than the accumulation of course credits [1]. Offering an opportunity for educators to set standards outside educational environment, OBE places its emphasis on expected skills set and knowledge gained out of the designed education system [2].

Data Mining Techniques have been continuously practiced to improve results of information and data processing. They exercise particular methods and mathematical algorithms to facilitate decision making processes by discovering hidden patterns and underlying information from large volumes of data [3]. With the help of data mining tools and applications, the techniques prove themselves useful in various dimensions and aspects of study [4]. One of them is Educational Data Mining.

Educational Data Mining (EDM) is an emerging discipline, targeted to improve learning environment by better understanding student models and the settings in which they learn. Educational settings, ranging from students' characteristics/states, learning environment to external influences, provide huge sources of potential data waiting to be processed. It is necessary to aware that educational data differentiates itself from average volumes of data by its multiple levels of meaningful hierarchy and non-independence. Faster access and broader usage of these valuable data are made available with development of educational data collection and data analysis tools, thanks to increasing uses of interactive learning environments, computer-supported collaborative learning, etc. [5]

Various works in the field of EDM are classified into few categories; among them is "Prediction". Predictive models have been developed to study individual learning, academic performance and the factors associated with student failure or non-retention rate in courses. Key areas of application of these methods are students modeling, domain's knowledge structure modeling, pedagogical support study and empirical evidence [6].

Common characteristic of all Data Mining models is that they treat systems as "black box". Their focus is on observed variables (system outputs) and on finding the patterns or regularities in the historical data in order to predict future behavior, without trying to explain these phenomena.

Differently, Mathematical models use mathematical concepts and languages to describe what happens inside that "black box" by proposing underlying mechanisms that cause those phenomena. They describe systems using a set of variables and a set of equations that establish relationship between the variables [7]. Besides, it is useful to incorporate subjective information as input for some mathematical models. Those information are based on intuition, experience, expert opinion, or based on convenience of mathematical forms [8].

1.2 Problem Statement

OBE models emphasize on the outcomes of education systems. In OBE framework, all the courses and assessment materials are structured to define "learning ends" for students, which are usually specific set of skills and knowledge [2]. Hence, student academic performance will be evaluated by credits given for which skills and knowledge they achieve out of the OBE system; and it is quantitatively measured in grade points.

However, a common dilemma faced by average educational institutions adopting OBE system is "performance gap", the gap between what the students can do and what they are expected to do. This gap tends to grow larger and larger over time, posing a serious threat to the education model. An observable consequence is discouragement and disengagement behaviors of affected individuals, especially at-risk students [9].

For years, educators have devoted many efforts seeking for applicable solutions to close this performance gap, or at least to mitigate it to a minimum extent. It has been among priority tasks of educators to achieve long-term goals of educational reform. Resources have been allocated to conceptualize and practicalize ways and means to improve students' academic performance by filling this gap [9]. One of the popular practices is application of EDM techniques. Adopting these EDM techniques to project student's academic performance or their grade points is a worth-noticed practice in the field. It is to help educators with informed corrective actions which aims to elevate student performance to their capabilities and to help them achieve the "learning ends" expected out of the OBE system.

For predicting performance in short-term courses, however, the complexity of indispensable data sources for EDM models such as students' demographic profiles and behavioral contents hinders their applicability. It causes a considerate burden for educators, as users of the models, to collect and manage those input data.

Therefore, in this project, the writer tried a new approach to the problem by simplifying the set of input data needed for the prediction system and by developing a customized mathematical model to process them. Final aim was still to successfully develop an applicable student performance predictive model as to facilitate educators in mitigating student performance gap currently existed in OBE environment.

1.3 Objectives

This project was aimed to develop a student performance predictive model which supported lecturers and instructors to mitigate student performance gap in their courses.

The objective was to develop, for lecturers as users, a simple computer application. It worked as a straightforward predictive model and it assisted lecturers to analyze patterns, to predict students' upcoming performance and final result, and to monitor their performance in order to provide timely intervention and adjustment. A high level of output accuracy and a sufficient level of system flexibility for users were expected.

Students' coursework marks and subjective inputs from users were the primary sources of educational data to be assessed. A highly-customized mathematical model was to be developed to perform an excessive breakdown of these coursework marks and to incorporate subjective information provided by users in order to predict student final performance at the end of the courses.

System functionality and accuracy testing were to be conducted to evaluate the developed model as well as the fundamental ideas underlying writer's new approach.

1.4 Significant of The Project

As mentioned, OBE model focuses primarily on student performance, as "outcomes" for a successful educational system. However, this model has been threatening with one common problem which is student failure to acquire skills and strategies at the rate that their normal-achieving peers do, resulting in their inability to successfully respond to grade-level curriculum demands.

Consequently, performance gap grows larger over time, causing lower-than-expected performance from students, leading them to discouragement and disengagement behaviors against the education system [9]. As final damage, this gap hinders the realization of full potential benefits from an OBE system.

This project, with its ultimate goal to help mitigate this gap, contributed to worldwide continuous efforts to improve the OBE model and its position in a debating progress towards an optimum education reform.

The development of this system also gave beneficial contribution to the fast growing field of student modeling. Simplicity of input data for short-term prediction models had not yet been properly valued before. This project's new approach, using students' coursework marks as primary point of assessed data, along with uniquely developed mathematical model, promised a valuable knowledge to the field.

For lecturers and instructors, while most of the currently available predictive tools were either over-power or too complex for them, the uses of a straightforward software would facilitate their job to achieve targeted outcomes which are desirable student performance, as set in the university's OBE goals.

1.5 Scope of Study

This study was narrowed to short-term courses, which last around one academic semester of 3-4 months, at Universiti Teknologi Petronas (UTP), undergraduate level. Due to limitedly attainable data for this study, the scope stopped at Programming courses only, not considering other disciplines such as Business, Finance, etc.

Since student's coursework marks were the major input for the predictive model, only those courses with coursework-final_exam grading structure were taken into account. Those courses such as Final Year Project which had no final exam were out of scope.

All survey and prototype testing activities were conducted within UTP campus, with participation of lecturers and students involved in those programming courses.

1.6 Feasibility of The Project

1.6.1 Technical & Scope feasibility

Technically, the writer (also system developer) was equipped with a moderate level of technology familiarity. Both hardware and software requirements were simple with not much burden for the developer. Intermediate uses of VB.NET coding was sufficient to develop a functioning prototype of the system.

The size and scope of this project was medium and suitable for a Final Year Project at undergraduate level. By limiting the scope to Programming courses in UTP only, data gathering, data analysis as well as prototype testing acitivities were convenient to the writer.

1.6.2 Time constraints

As a Final Year Project in the programme structure, the writer had been given a standard eight(8) months (two academic semesters) to complete the project. Consider the narrowed scope and its technical feasibility, the risk level of timely completion is low.

A detailed Gantt Chart was prepared by the writer to dynamically monitor project progress throughout its duration. More details can be found in 'Methodology' chapter.

CHAPTER 2

LITERATURE REVIEW

2.1 Related Works

Ayesha et al. [3] proposed a prediction model which used coursework components including class quizzes, assignments, tests, etc. as internal assessment materials. Additional information such as attendance, previous performance and extra-curriculum involvement were also concerned. Also, external assessment, based on students' performance consistency level throughout recent final exam scores, was incorporated.

Their designed predictive model aimed to provide lecturers with relevant information about student's performance before the conduction of final exam, which would help to improve overall learning practice in the course and to reduce withdrawal ratio. Also, the prediction of students fail ratio in an on-going course provided major help to lecturers when designing course structure, teaching methods, and frequent assessment materials. At-risk students with low performance were spotted in order to be saved from serious academic position. Appropriate subsequent steps were then taken by the lecturers to improve their performances and to save them from failure. At a bigger scope, the proposed model also helped compare students' success rate throughout their four-year course of undergraduate degree.

For their data mining model, the authors employed clustering technique, one of the most basic techniques used in analyzing and processing huge data volumes, and K-means clustering algorithm, to segment ate student groups based on their characteristics and behavioral contents.

In another study, Ogor [10] focused on monitoring the performance of students' continuous academic results, based on tests and exam scores, and how it played a crucial role in providing educators with relevant and valuable information which helped to improve interactively changing learning environment for students.

Ogor emphasized the needs for effective and efficient performance monitoring systems in order to offset the implicitly unobserved knowledge and information hidden inside huge amounts of available educational data. Various data mining techniques were developed and utilized to react upon the quest to improve educational institutions' student performance monitoring system. Classification of students was facilitated with application of machine learning processes.

The author stated that mere value of entry-level assessment of students was not sufficient in giving an efficient monitoring in long term. Therefore it raised a need for dynamic follow-up monitoring of students' performance throughout the course of study. Only then the suitability of students before admission could reveal itself. His objective was to design a measurable student progress monitoring model with rapid processing and quick result in order to facilitate educational system. Factual and partly behavioral factors of students' profile were taken into account in performance profiling. That included factual contents such as gender, race, previous test results records, etc. and behavioral contents such as attitudes, motivation, curriculum involvement and peer influence. A simple rapid response system was developed to spot out students who needed special attentions and reinforcements upon.

With fairly large input volume of operational data of 1,360 students in two consecutive academic years and five different courses, Ogor came to a conclusion that data mining techniques proved their usefulness in educational environment with a 94% success rating from his functioning student monitoring tool.

Merceron and Yacef [11] questioned the application of data mining techniques in educational settings and their usefulness in improving teaching and learning experience for all stakeholders involved. A number of studies following this direction were mentioned in the paper. They also proposed a future trend of ideas merging in which simple statistics, queries and visualization algorithms were together employed to predict student performance. They suggested a simple pedagogical policy utilizing clustering and cluster visualization methods to identify shared characteristics and behavioral state of failing students. It aimed to provide a timely intervention to prevent at-risk students from serious harm before final exam.

While online learning environment for educational settings was emerging itself as a potential and expanding trend among institutions, White and Larusson [12] conducted a study to examine possibilities and limitations of online systems where available data namely transmission of information, evaluation of teacher, learner performance and online interaction were recorded and ready to be processed. These Learning Management System (LMS) showed their capabilities as a crucial supplementary, even worthy substitute, to conventional face-to-face communication environment.

Different data mining techniques namely logistic regression, artificial neural network (ANN) and neuro-fuzzy were used by Rusli, Ibrahim, and Janor in their study [13] . They took students' cumulative grade point average (CGPA) upon graduating as a success measure of their academic performance. Demographic profiles and first semester result were all the necessary inputs for the three developed predictive models. Also using ANN model, Oladokun, Adebanjo and Charles-Owaba[14] together proposed another academic performance predictive system with a correct prediction rate of 70%.

Another study [15] conducted at Universiti Teknologi Mara analyzed a wide range of factors including students' demographic profiles, active learning, attendance, extra curriculum involvement and course assessment frequency. It concluded that all mentioned settings were directly related to students' academic achievement.

2.2 Critical Analysis of Related Works

All related works adopted educational data mining techniques into their predictive models and concluded that predicting students' academic performance is crucial for educational institutions as the information collected can be critically important for immediate and future improvement of the educational system, specifically the mitigation of performance gap. Strategic programs can also be planned from those information to maintain students' performance throughout their course of study [13].

With similar measurement as in [3], in this project, a student's final grade in a particular course is still adopted as the single indicator of his or her overall academic performance. Student with poor performance raises a potential threat toward unsatisfied final result at the end of the course. This leads to the student being objectively classified as low-academic-performance group.

In all related works, the scope of studies can be categorized into TWO(2) main groups: long-term prediction and short-term prediction.

2.2.1 Long-term prediction

Most of the previous papers fall into this group. In these papers, educational data mining techniques were all applied into long-term prediction and long-term assessment of students' academic performance.

Ogor [10] proposed a predictive model, which took into account students' entry-level background at the beginning and a dynamic follow-up database of student performance throughout the program, to predict their results upon completion of the program, which would be three or four years later. Rusli, Ibrahim, and Janor [13] took demographic profiles and merely first semester result of undergraduate students to calculate a projected result of their academic positions at the end of a four-year program, without any follow-up information. Similarly, Oladokun, Adebanjo and Charles-Owaba [14] and Ali et al. [15] attempted to produce most accurate and consistent prediction for a several-year study program by taking different sets of data at the beginning of the period.

All these studies were different from each other only in terms of selected input, data mining methods and algorithms used; yet they all showed efforts to project student's academic position few years ahead of time, with less concern to dynamic movements, changes and immediate external influences on students during the course of study.

2.2.2 Short-term prediction

The second group is short-term prediction with only few other works involved. Ayesha et al. [3] put their focus of study on a narrowed scope of particular courses which lasts averagely few months each. Instead of predicting years-later performance of students, the paper aimed to excessively assess coursework breakdown, along with other external assessment variables, in one particular course at a time, in order to project students' outcome at the end. Similarly, Merceron and Yacef [11] proposed a pedagogical policy with clustering and cluster classification methods to spot out group of students with high potential of failing the final exam before it was conducted.

In terms of prediction scope and timely monitoring, this group of short-term prediction models apparently bring less value to the educational data mining field of study; however, the key advantage of this short-term scope is to produce a much more accurate prediction which are more dynamically responded to the changing variables during the course of study, and to provide valuably additional follow-up information for other long-term prediction models.

As for the data mining techniques adopted, all previously related works shared one common input: they all directly took into account non-coursework related data such as demographic profile (gender, family background, etc.) and behavioral contents (student model of characteristics, attendance, extra curriculum involvement, motivation, etc.). As an instance, in [15], attendance became one of the major indicators in the predictive model as each unit of students' time spent in the class was proven to be one of the most valuable and important determinants of student success [16].

2.3 New approach to the problem

With a fairly different approach from previous related works, in this study, the writer selected coursework breakdown as the single direct input for the proposed predictive model. Within the scope of this study, among four factors namely (1)coursework marks, (2)psychological questionnaire result, (3)total number of materials download and (4)total number of times online in E-learning platform, there was only one factor which has strong relationship with student's final grade. That was coursework marks; other three factors showed weak and unreliable relationship with student's final grade. UTP's E-learning platform is one instance of Learning Management System (LMS). As stated in [12], transmission of information and learners' interaction on these platforms were direct factors influencing students' academic performance. However, in the case of UTP E-learning system, an unpublished study by Che Sarah, C.N., Elaine, C.Y.Y. in 2011 had shown weak relationships between these factors and students' actual performance.

One crucial criterion of this new approach is that the input data, mainly students' coursework marks, must continuously develop itself throughout the life of each conducting course. Pursuing this, the author bares limitation of this proposed system in terms of timely prediction and application scope. Also, subjective information from users such as exam paper's difficulty level are necessary to improve accuracy of the outputs. Coursework breakdown data were started to be recorded and analyzed only after the first coursework component's result is published (E.g. Test 1 result). From this initial input, users (which are lecturers or instructors) will start to enter their subjective evaluations in order to improve prediction outputs. This process was repeated for each of the next major inputs (such as Test 2 result, assignments, lab exercises and quizzes, etc.) until final coursework is completed. More details on this structure will be described in later chapters.

2.4 Advantages of This Predictive Model

As compared to other students' performance predictive models in related works, FOUR(4) major advantages of the model in this study are:

- The forecasted outputs are continuously refined and re-evaluated to be more accurate throughout the short-term courses.

- By using mainly coursework marks, it omits the burden of collecting complex and abundant type of inputs such as demographic profiles and behavioral contents.

- Users, which are lecturers, are provided with flexibility to decide which component(s) of students' courseworks is a good predictor of their final exam scores and course outcomes, also to decide how final exam paper's difficulty level will quantitatively affect those end results.

- Users can observe how each coursework component (both published and to-be-conducted components) in one particular course expectedly affect students' final grade, which facilitates timely intervention and assessment materials re-evaluation to achieve OBE objectives.

CHAPTER 3

METHODOLOGY

3.1 Overview

This project adopts Rapid Application Development (RAD) prototyping methodology. It involves system construction with repeatedly spiraling through the phases and relies on rapid prototyping rather than thorough planning and analysis phase. The analysis, design and implementation phases are performed concurrently and repeatedly until completed. The first prototype is the first part of the system that user will use. The prototype then evolves into the final system. With this approach, the prototypes are utilized to their fullest potential [17].

Figure 3.1 below shows the framework of the prototyping system development methodology:

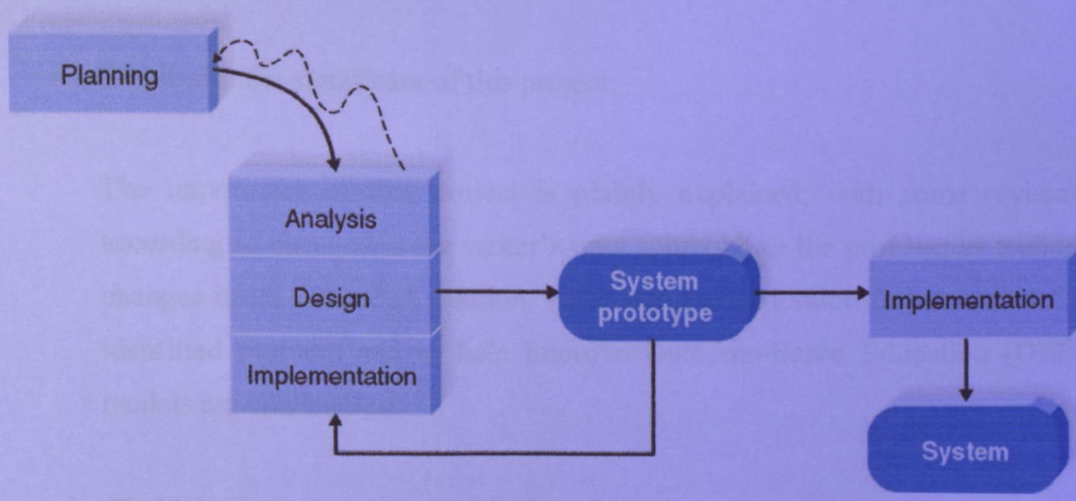


Figure 3.1 Framework of RAD prototyping methodology

Various levels of completeness and complexity of the proposed system, as well as ease in changing requirements throughout the course of system development are of main advantages of this methodology. An iterative construction approach is employed to accelerate the requirement analysis and design phases and to also detect errors, programming and time constraints earlier in prototypes rather than later in complete system model. As change is an expected factor during development, this approach is at most suitable usage.

3.2 Project Activities

3.2.1 Planning phase

- Research on the background of the preceding study and related works.

- Identify the problem and propose the solution.

  The problem statement of previous paper is reevaluated based on new findings from the first activity. Then, a solution is accordingly proposed for the revised problem.

- Emphasize the significant of this project.

  The importance of this project is plainly explained, with some revision according to changes in the writer's new approach to the problem as well as changes in the proposed solution. Its valuable contribution to help solve the identified problem and to help improve Outcome-Based Education (OBE) models are emphasized.

- Clarify project scope, goals and objectives.

Goals and objectives of the project are clarified to guide subsequent activities. Project scope is narrowed down specifically to suit project's needs, requirements and constraints.

- Analyze project feasibility.

Given the standard 8-month period for Final Year Project in UTP, as well as other technical and scope constraints, the writer conducted a feasibility analysis to examine the project's overall chance of success.

- Identify milestones and Gantt chart.

Project milestones and Gantt chart are developed to support monitoring project activities. See Appendix C for more details.

3.2.2 Analysis Phase

- Clarify analysis objectives.

As mentioned in the earlier chapters, the author's approach in developing this students' performance predictive model is fairly unique and different from previous ones. No students' demographic profile or behavioral contents will be directly taken into account. Instead, coursework marks is the primary source of input, along with subjective inputs from users/lecturers.

This analysis phase is to discuss and evaluate the rationale and the justification behind this concept, the idea and its unique approach. The goal is to explain the authenticity and the cogency of the writer's research, based on the validity of research data, measures and time taken to conduct the study.

The objectives of this phase are listed as below:

- ✓ Analyze related works against the proposed solution to evaluate it
- ✓ Develop system requirements definition
- ✓ Analyze fundamental ideas underlying the writer's unique approach
- ✓ Gather and process necessary data to evaluate the ideas
- ✓ Design multiple analysis models to support system development process
- ✓ Develop a mathematical model for the system

- Analyze related works critically: scopes, data mining techniques, algorithms used, and relative application to the scope of this study.

- Identify advantages of the proposed system and its unique approach.

  The writer's unique approach to the identified problem leads to certain crucial advantages when applying the proposed model into the scope of this study, as compared to previous works.

- Summarize an overview of assessment materials in Programming courses at UTP and prepare system requirements definition.

  ➢ <u>Overview of assessment materials in Programming courses at UTP</u>

  All programming courses at UTP employ coursework-final exam grading structure. The weightage between these two parts are usually 60-40 (which means 60% coursework and 40% final exam).

Coursework(CW) components usually include : test 1, test 2, lab exercises, assignments, quizzes, and group/individual project.

Average course lasts 14-week (excluding final exam), and the coursework is usually completed within the last two weeks before final exam.

Final exam question paper of each course is often prepared in advance at around week 3 or week 4, before most of the coursework components (CWCs) are assessed. By then, the lecturer has had full knowledge of the questions in the paper when conducting the course as well as when preparing CWCs such as tests, assignments, quizzes, etc.

- <u>System requirements definition</u>

a. *Functional requirements*

Projecting performance:

- The user can edit list of coursework components (CWCs) and their weightages.
- The user can change timely order of the CWCs conducted.
- The user can insert/edit/delete students' results for each corresponding CWC.
- The user can edit status(published/to-be-conducted) of each CWC.
- The user can edit whether or not a CWC is a good predictor of final exam score.
- The user can generate projected Total CW Score for each student.
- The user can decide whether or not the final exam paper's level of difficulty does affect students' final exam scores. If yes, the user

can edit how this difficulty level quantitatively affects the scores of different groups of students in the course.

- The user can generate projected exam score and their course's final score and final grade accordingly.
- The user can generate a final grade range (such as "B+ to A-"," D+ to C", etc.) in which a student' final score may fall into.

Monitoring performance:

- The user can sort any data column (CW components, Total CW, Exam Score, Final Score, etc.) alphabetically or smallest to largest value or vice versa, etc. to view 'at-risk' or 'well-performed' student groups.
- The user can view summary table and summary charts of students' projected performance, after generating projected final score and final grade.
- The user can test how each subsequent CW components, their relationship to final exam paper, or the paper's difficulty level, etc. affect students' expected scores.

b. Non-functional requirements

- The system will operate in Windows (XP and above) environment as an offline standalone application.
- The system must be fully functioning, yet straightforward enough for average users (lecturers/instructors)
- The system will be able to connect with Microsoft Access database files.

- The data must be able to be saved or updated upon users' requests.
- No special security requirements are anticipated.
- No special cultural and political requirements are anticipated.

- Critically analyze fundamental ideas which form the foundation for the writer's approach.

➢ Clarify underlying ideas

Following points are to be discussed and evaluated:

within the scope of this study,

- **P1**: Excessive coursework breakdown analysis is SUFFICIENT for acceptable predictive outputs.
- **P2**: Lecturers'/users' SUBJECTIVE INPUTS (such as which coursework components are good predictors of final exam score and how final exam paper's difficulty level affects students' scores), are helpful to improve the accuracy of outputs.
- **P3**: Students' demographic profiles and behavioral contents are NOT necessary to be included into the data sources.

Figure 3.2 Three underlying ideas of writer's approach

➢ Develop hypotheses to evaluate the ideas

Following are the five(5) hypotheses (namely H1, H2, etc.) developed by the writer to form the skeleton for later data collection and analysis stages.

They are divided into three(3) categories: coursework marks, final exam paper and other factors.

1. *Coursework marks*

Coursework (CW) marks is a major assessment criterion in most courses at UTP. CW marks usually carry a percentage of 40% to 60% out of the overall final result of 100%. As mentioned in Chapter 2, an overall final grade in particular course always comes in direct relation with CW marks. For instance, one who scores 80-90%, out of total CW marks allocated, is most expected to also score 80-90% in his/her final exam.

- **H1:** One's *Total Coursework* marks is propotional to his/her *Exam Score*.

However, for each specific *CW Component* (CWC), the relationship between it and *Exam Score* is at different levels from one to another. For instance, *test papers* with similar type of questions as in *exam paper* would carry a relevant relationship between *test scores* and *Exam Scores*; whereas *group projects* usually would not.

- **H2:** Only for those *CW Components* with SIMILAR type of questions to exam paper's, one's score is proportional to his/her *Exam Score*.

Additionally, given the standard A-F grading structure, one's *Total CW Lost score* DOES affect his/her target scores for the final exam. In other words, the difference between *Total CW Score* and nearest potential ranges of grade in the A-F grading system, plays an important role in predicting exam score. The A-F grading structure at UTP is: A (85-100), A- (80-84.9), B+ (75-79.9), B (65-74.9), C+ (55-64,9), C (50-54.9), D+ (45-49.9), D (40-44.9), F (0-39.9). For instance, one who lost 20-25% over 60% coursework is most expected to be satisfied with a B in final result, given him/her much less pressure preparing for the final exam, since he can afford to lose up to 15% out of 40%. Similarly, one who lost 10-13% out of total coursework would most probably set his/her target for A- (primary) or A (secondary), which allows him/her to lose 7%, at max, over 40% allocated for the exam paper.

- **H3:** Given the standard A-F grading structure, one's *Total CW Lost Score* DOES affect his/her target scores for final exam.

2. *Final exam paper*

Final exam paper is a crucial element forming the final score and final grade of a student in specific courses. *Difficulty level* of the questions plays an important role in determining which grade in the A-F grading system the student may get.

Complexity (the quality of each question to be compounded in terms of multiple learning concepts involved), originality (the quality of being new in the way lecturers apply taught concepts to the questions), covered scope (the broadness of learning concepts such as number of chapters, references, etc. covered in the exam) and time requirements (average time to complete the paper as compared to the standard allocated 2-3 hours per paper) [19] together indicates the overall level of difficulty.

- **H4:** Overall *Difficulty level* of the final exam paper is negatively related to students' *Exam Score*.

3. *Other factors*

For *short-term* courses, the impact of students' *demographic profiles* (gender, race, family background, education background,etc.) and *behavioral contents* (attendance, involvement in extra-curriculum activities, etc.) can be ignored.

- **H5:** Students' *demographic profiles* and *behavioral contents* are NOT necessary to be included as a data input for the predictive model.

- Collect data to evaluate developed hypotheses

➢ Design survey questions

All of the 5 hypotheses, which later form the skeleton of author's mathematical model, are evaluated against results data collected from a survey.

All 5 hypotheses, though are generated from sufficient researches, are still of the writer's subjective opinion. Hence, it is crucial to evaluate these opinion against "public opinion"; in this case it is UTP students in programming courses. With that specific scope being set, a concise survey aimed for a selected sample of the population is of best interests among various data gathering methods.

This survey's goal is to gain insights into students' perspective about major factors that indicate their expected performance in final exam and course's final result accordingly. The survey's scope is maintained to be the same as the overall scope explained in Chapter 1; it is also explained to all survey participants.

The survey consists of ten(10) questions, which are segregated into four(4) sections : Total CW mark, CW components, Final exam paper and Other factors.

For each question, the participants are required to select one out of five(5) options from the Likert 5-point scale : strongly disagree, disagree, neutral, agree and strongly agree [20].

See Appendix D for a complete version of the survey.

Each question's responses are, then, analyzed to evaluate a corresponding hypothesis. Table 3.1 below summarizes this structure:

Table 3.1 Survey design structure

Hypothesis	Question(s)
H1	1
H2	3,4
H3	2
H4	5,6,7,8
H5	9,10

➢ Conduct the survey

A survey with 48 participants was conducted within UTP campus. The selected participants are programming courses' students ranged from year 1 (first year) to year 4 (final year) at undergraduate level. Their backgrounds are also spread in multiple disciplines, with majority (28/48) are from Computer and Information Science (CIS) programme; others include Petroleum Engineering (PE), Electrical & Electronics Engineering (EE) and Civil Engineering (CV).

➢ Design an analytical procedure to process survey data

Table 3.2 below shows how each participants' response will be quantitatively measured by assigning different weights for each response.

Table 3.2 Survey response-weight structure

Response	Weights
Strongly disagree	-10
Disagree	-5
Neutral	0
Agree	+5
Strongly agree	+10

For each question, *total weight* accumulated from all 48 participants' response will be calculated. If it is *positive*, the corresponding hypothesis is *approved valid* and it will be directly reflected into system structure and the algorithm. For instance, if question no. 1 received 3 strongly disagree, 6 disagree, 5 neutral, 27 agree and 7 strongly agree, its total weight is **+145** (*positive*). As a result, hypothesis H1 would be approved valid.

For hypotheses that involves more than one question such as H2, H4 and H5, the *average total weights of all related questions* will be calculated and evaluated.

- Analyze survey data

  - Generate survey result summary table and summary chart

    Using "Pivot Table" function in Microsoft Excel, survey data is recorded and analyzed to generate a summary table and visual charts displayed the overall results. See Chapter 4 for more details.

  - Adopt the designed analytical procedure to evaluate developed hypotheses

    The developed *response-weight structure* is adopted to evaluate the five(5) hypotheses: whether each of them is approved valid or not. See Chapter 4 for more details.

  - Re-evaluate the fundamental ideas underlying the writer's new approach

    The proposed hypotheses is then assessed to re-evaluate writer's fundamental ideas P1, P2 and P3: whether each of these three(3) fundamental ideas is approved valid or not, within the study scope. See Chapter 4 for more details.

- Develop analysis models for the system

  - Develop Activity diagram (functional model)
  - Develop Class diagram (structural model)

See Chapter 4 for complete diagrams.

- Develop a **mathematical model** to support the unique approach

  - Form an overview structure for the model

    As mentioned in system requirements definition, Total CW Score, Exam score, Final score/grade are projected based on *published CW Components* (results are already out and available for processing) and *subjective inputs* from users.

    This model structure, which is derived from the approved underlying ideas, will help to provide a general view about the mathematical model works: how one variables can be projected/derived from the others. See Chapter 4 for more details.

  - Develop detailed formulas for each component.

    This explains in details, using mathematical formulas, how the model works throughout the whole system, from initial inputs (which are *published CWCs*) till end results (which are the projected *Final grade*, *Grade range* and *summary tables/charts*). See Chapter 4 for more details.

3.2.3 Design Phase

- Clarify design objectives
  - ✓ Adopt a simple architecture on which the system is built.
  - ✓ Design the system with straightforward functions and user-friendly interfaces using Object-Oriented Systems Analysis and Design techniques.
  - ✓ Use VB.NET software to code the designed system.
  - ✓ Develop a functioning prototype for testing purpose.

- Develop system architecture

- Design Graphical User Interfaces (GUIs) and built-in system functions accordingly.

3.2.4 Testing and Implementation Phase

- Clarify testing and implementation objectives *(prototype)*
  - ✓ Test the designed functions and the performance of the prototype.
  - ✓ Test accuracy level of the predictive outputs using past data.
  - ✓ Conduct change management if needed
  - ✓ Finalize the prototype and put it on hold for future full system development for actual implementation if needed.

- Conduct functionality test

- Conduct accuracy test

- Finalize the prototype

As for the scope and the initial requirements of this Final Year Project, a functioning prototype (available for testing and demonstration) is sufficient. A fully developed system is not necessary at present, yet can be feasibly evolved from the prototype.

3.3 Tools Requirements

3.3.1 Hardware

One computer with average specifications (e.g. Intel Core 2 Duo T7500, 160GB HDD, 2GB DDR2, etc.) is sufficient.

3.3.2 Software

The prototype is developed using:

- Windows Vista/7 operating system, for running environment platform.
- Visual Basic Express 2010 window application programming software, for VB.NET coding.
- Microsoft Access 2007, for database storage, access and management.
- Microsoft Excel 2007, for survey data analysis and summary reporting.
- Microsoft Word 2007, for survey design and reporting.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Data Analysis

4.1.1 Survey results summary table and chart

Table 4.1 below shows the summarized result of all participants' responses to each question:

Table 4.1 Survey result summary table

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
1-Strongly Disagree	4%	2%	2%	2%	13%	6%	4%	2%	8%	2%
2-Disagree	13%	2%	2%	6%	17%	19%	19%	19%	19%	21%
3-Neutral	23%	21%	29%	35%	46%	31%	40%	46%	31%	35%
4-Agree	44%	38%	54%	46%	21%	40%	31%	27%	38%	38%
5-Strongly Agree	17%	38%	13%	10%	4%	4%	6%	6%	4%	4%

The underlined figures are those of majority responses.

Figure 4.1 below shows the summary chart with data from the summary table:

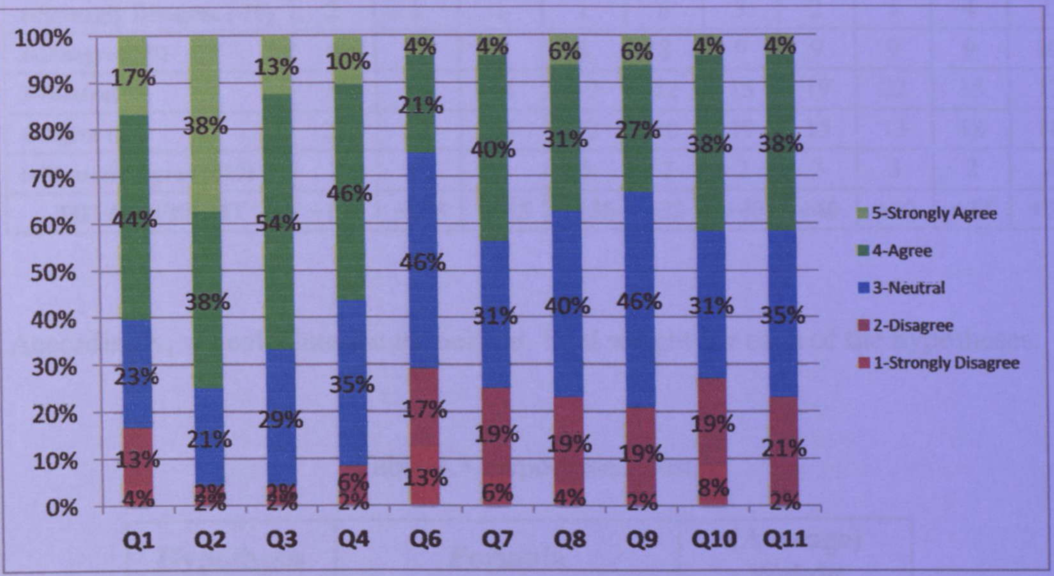


Figure 4.1 Survey result summary chart

The chart shows that majority of participants are either "agree" or "neutral" with the proposed statements.

4.1.2 Evaluation of the proposed hypotheses

Next, in order to quantitatively measure these collected figures, the writer adopted the response-weight system (Table 3.2) as mentioned in the "Analytical procedure" section of the previous chapter.

Table 4.2 below summarize total weight of each question:

Table 4.2 Survey result analysis using response-weight system

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
1-Strongly Disagree (-10)	2	1	1	1	6	3	2	1	4	1
2-Disagree (-5)	6	1	1	3	8	9	9	9	9	10
3-Neutral (0)	11	10	14	17	22	15	19	22	15	17
4-Agree (+5)	21	18	26	22	10	19	15	13	18	18
5-Strongly Agree (+10)	8	18	6	5	2	2	3	3	2	2
TOTAL WEIGHT	**+135**	**+255**	**+175**	**+135**	**-30**	**+40**	**+40**	**+40**	**+25**	**+50**

Accordingly, we calculate the associated total weight for each of the hypotheses:

Table 4.3 Hypotheses' result

Hypothesis	Formula	(Average) Weight
H1	= Q1	**+135**
H2	= (Q3+Q4)/2	**+155**
H3	= Q2	**+255**
H4	= (Q5+Q6+Q7+Q8)/4	**+22.5**
H5	= (Q9+Q10)/2	**+37.5**

As derived from the table above, the average weights for every hypothesis are positive, meaning that all proposed hypotheses are, to certain extent, approved valid. However, the degree of 'positiveness' is different from one to another; therefore, Table 4.4 below shows a number of caveats to be noted:

Table 4.4 Evaluation of hypotheses

Hypothesis	Points	Degree of 'Positiveness'	Notes
H1	+135	Positive	The hypothesis is approved valid.
H2	+155	Positive	The hypothesis is approved valid.
H3	+255	Highly positive	The hypothesis is strongly approved valid.
H4	+22.5	Moderately positive	The hypothesis is approved valid. Yet the impact of the final exam paper's level of difficulty is perceived as low. Hence, this difficulty level is included into the model as an optional variable; user can choose whether or not to incorporate it into the projection.
H5	+37.5	Moderately positive	The hypothesis is approved valid. Yet, at the point when the survey was conducted, there may be still concerns about the accuracy level of the predictive model if 'demographic profile' and 'behavioral contents' are NOT considered. One suggested reason is that the accuracy test had not been completed to justify system outputs.

4.1.3 Evaluation of the proposed fundamental ideas based on hypotheses' results

Below is a summary list of the five(5) hypotheses:

- **H1:** One's *Total Coursework* marks is proportional to his/her *Exam Score.*
- **H2:** Only for those *CW Components* with SIMILAR type of questions to exam paper's, one's score is proportional to his/her *Exam Score*.
- **H3:** Given the standard A-F grading structure, one's *Total CW Lost Score* DOES affect his/her target scores for final exam.
- **H4:** Overall *Difficulty level* of the final exam paper is negatively related to students' *Exam Score*.
- **H5:** Students' *demographic profiles* and *behavioral contents* are NOT necessary to be included as a data input for the predictive model.

The positive result of hypotheses H1, H2, and H3 shows that students' detailed courseworks carry two potentially useful pieces of information for the developing predictive model.

Firstly, the approved hypotheses H1 and H3 indicate a direct relationship between *Total Coursework Score* and *Exam Score*.

Secondly, the approved hypothesis H2 indicates a direct relationship between *certain Coursework Components* with *Exam Score*. Hence, a coursework breakdown analysis is necessary to be included in the model in order to improve accuracy level of predictive outputs.

As a result, the first fundamental idea is justified:

- ***P1:** Excessive coursework breakdown analysis is SUFFICIENT for acceptable predictive outputs.*

Next, the approved hypothesis H2 also indicates that NOT all components are necessarily good predictors of final *Exam Score*. Hence, *subjective inputs* from the users, about which ones are, will improve the accuracy of predictive outputs.

Similarly, the approved hypothesis H4 indicates that overall difficulty level of the exam paper does, in fact, influence students' exam scores. Hence, it needs to be included into the list of main factors that influence students' performance.

Lecturers are also expected to provide their subjective inputs on this difficulty level assessment. The inputs are from their own perspective, yet are based on their knowledge about students' recent performance in their conducting courses. For example, the same final paper may be difficult with this year's students, yet be easy for next year's students; in this case, the lecturers are supposed to input "difficult" for this year, yet input "easy" for next year, even though it is still the same paper.

As a result, the second fundamental idea is justified:

- ***P2:** Lecturers'/Users' SUBJECTIVE INPUTS, such as which ones among the coursework components are good predictors of final exam score, are helpful to improve the accuracy of outputs.*

Lastly, the approved hypothesis H5 indicates that *demographic profiles* and *behavioral contents* are NOT necessary to be included into the data sources.

As a result, the last fundamental idea is justified:

- ***P3:*** *Students' demographic profiles and behavioral contents are NOT necessary to be included into the data sources.*

4.2 Framework of The System

4.2.1 Analysis models

- ***Activity diagram (functional model)***

Figure 4.2 below shows the functional activity diagram which illustrates activity flows of the system:

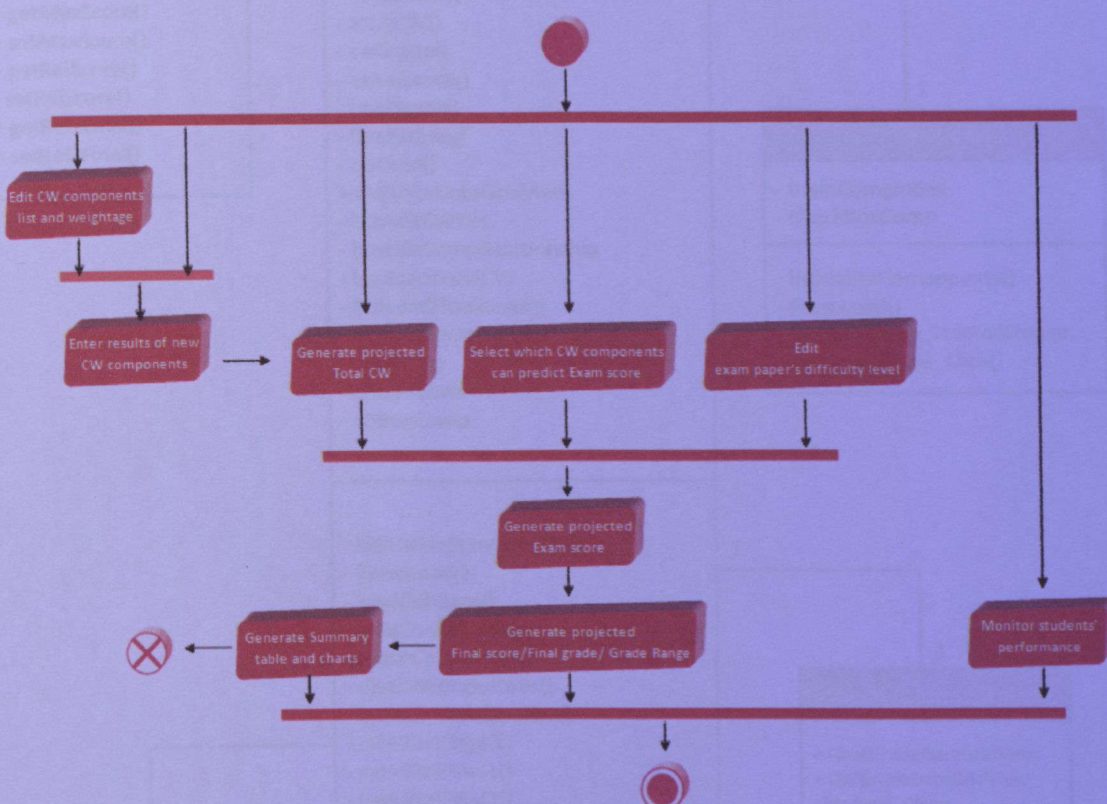


Figure 4.2: Functional activity diagram

- ***Class diagram (structural model)***

Figure 4.3 below shows the structural class diagram which illustrates logical organization and structure of data that supports the functional model.

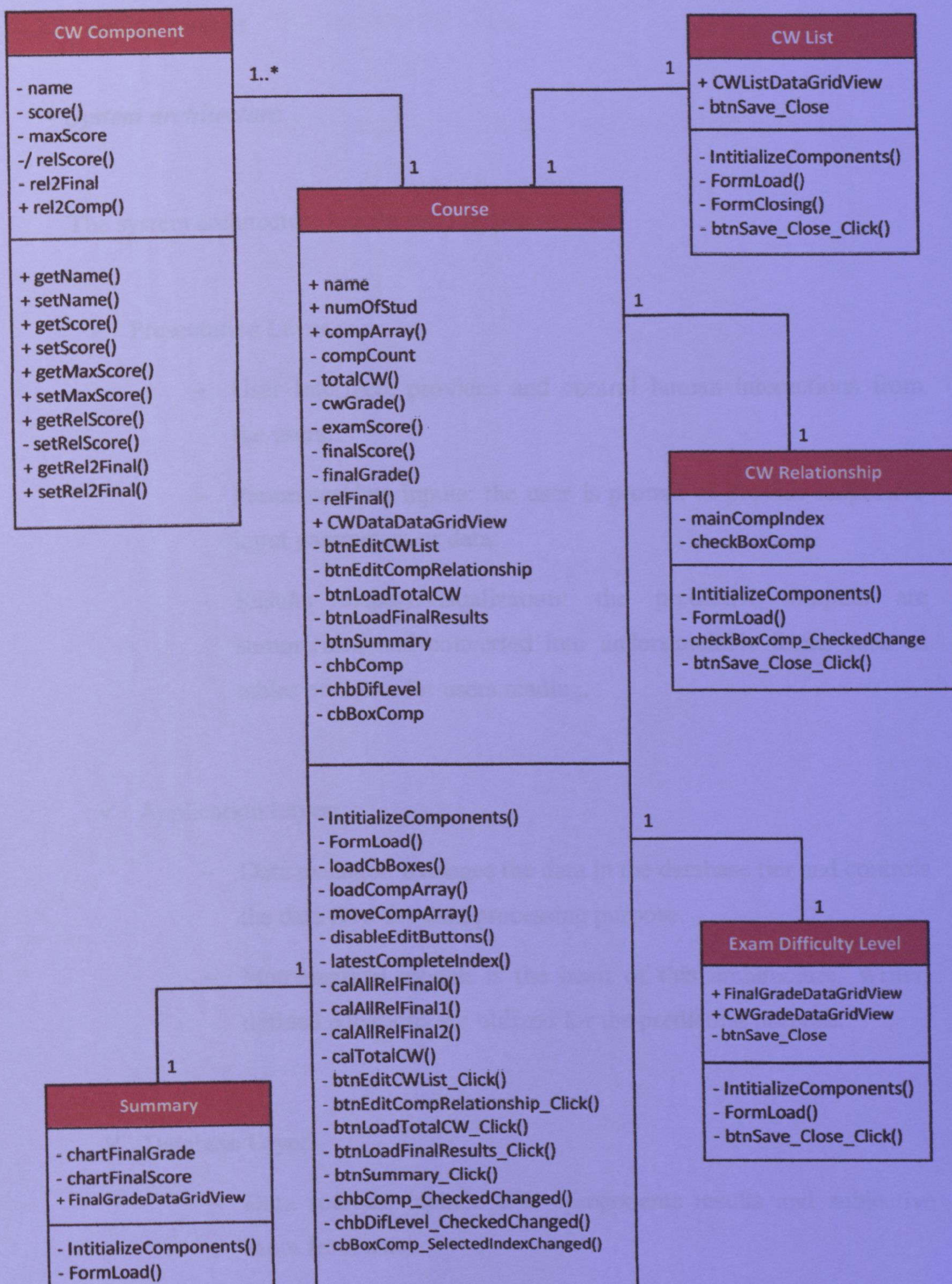


Figure 4.3 Structural class diagram

4.2.2 Design models

- ***System architecture***

The system architecture has three(3) layers:

✓ Presentation Layer :

- User Interface: provides and control human interactions from the users.
- Parameter/data inputs: the user is prompt to provide respective input parameters or data.
- Results display/visualization: the predictive outputs are summarized and converted into understandable forms such as tables or charts for users reading.

✓ Application Layer:

- Data manager: manages the data in the database tier and controls the data flow for data processing purpose.
- Mathematical model: is the heart of this architecture. Writer-defined equations are utilized for the predicting purpose.

✓ Database Layer:

- Data sources: include CW components results and subjective input from users.
- Data outputs: store the results from application layer.

Figure 4.4 below illustrates how the system works:

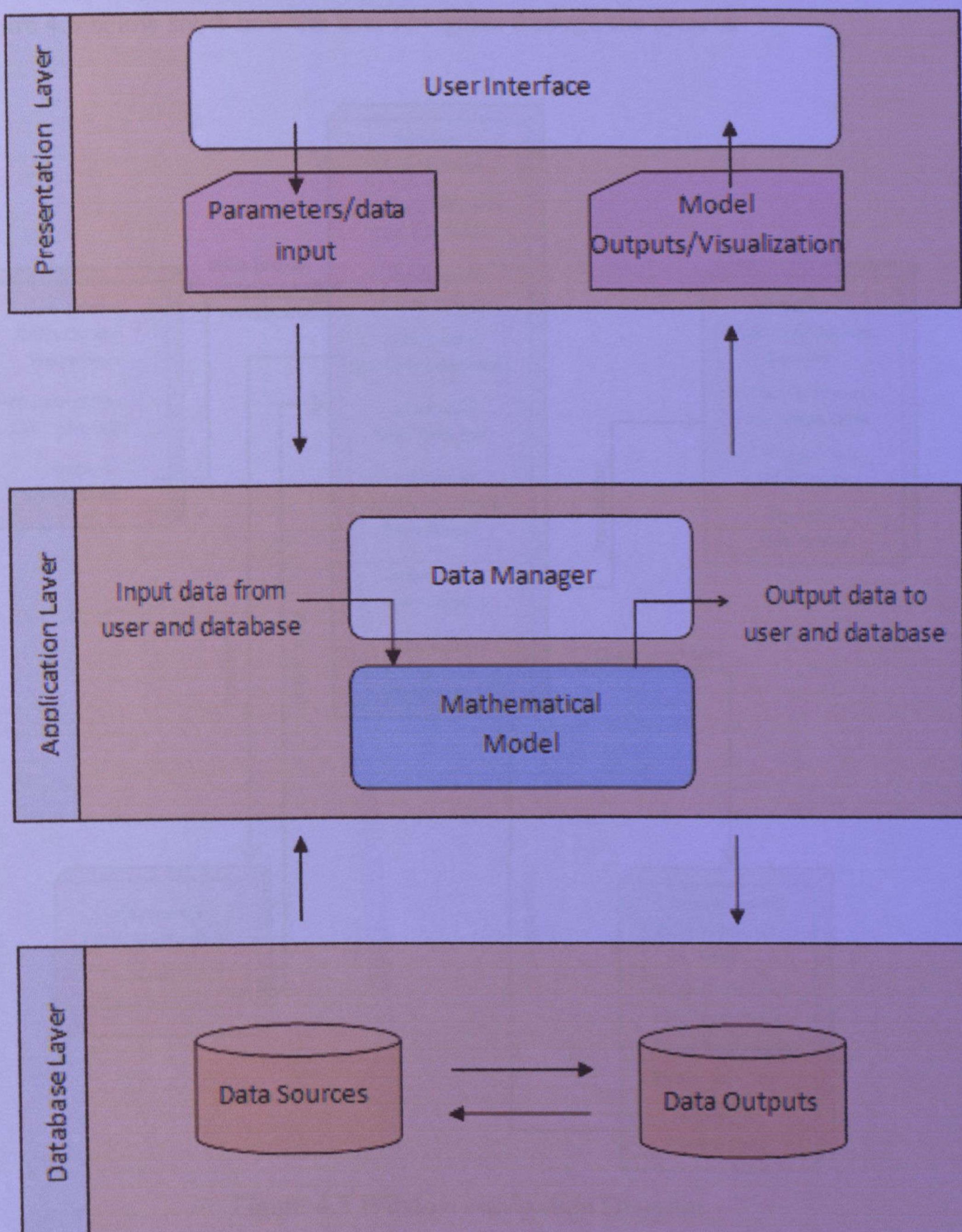


Figure 4.4 System architecture

▪ ***GUIs design***

Figure 4.5 below show how the user navigates through the system:

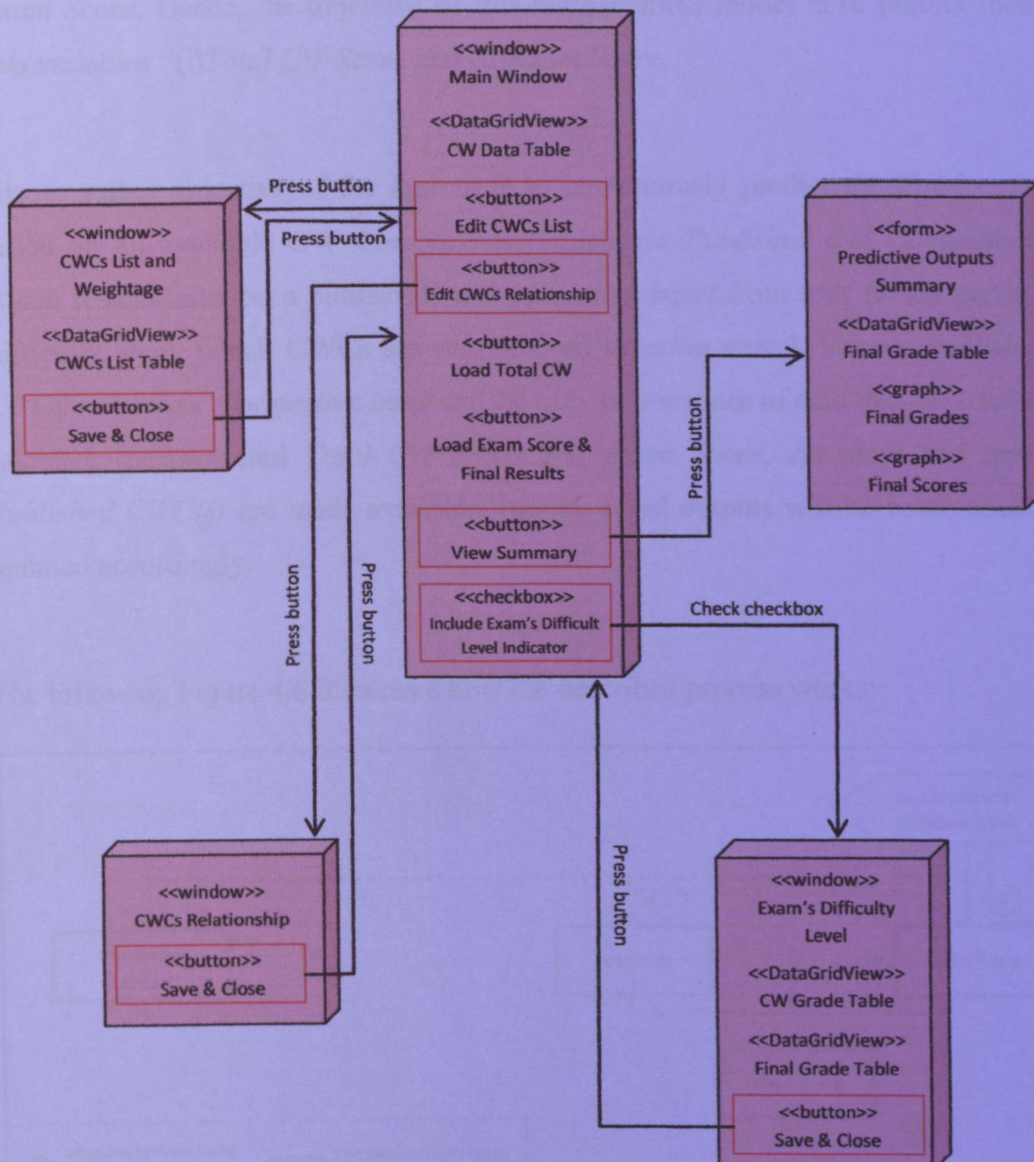


Figure 4.5 Window Navigation Diagram

See Appendix A for snapshots of main interfaces.

4.2.3 Mathematical model

The objective of the system is to predict students' *Final score* (Final grade) at the end of a particular course. This Final score is an addition of *Total CW Score* and *Exam Score*. Hence, the objective of this mathematical model is to predict these two variables : (i)*Total CW Score* and (ii)*Exam Score*.

Also, another objective of the system is to continuously predict the *Final score* based on all available CW Components (which are *Published CW Components* which results have been published) and *subjective input* from user (exam paper's difficulty level, which CWCs are proportional to exam score). Hence, *Published CWC(s)* and *user's subjective input* are the only two sources of data that are used to calculate the projected *Total CW Score* and *Exam Score*. As more and more *Published CWC(s)* are made available, the projected outputs will be continuously updated accordingly.

The following Figure 4.6 illustrates how the described process works:

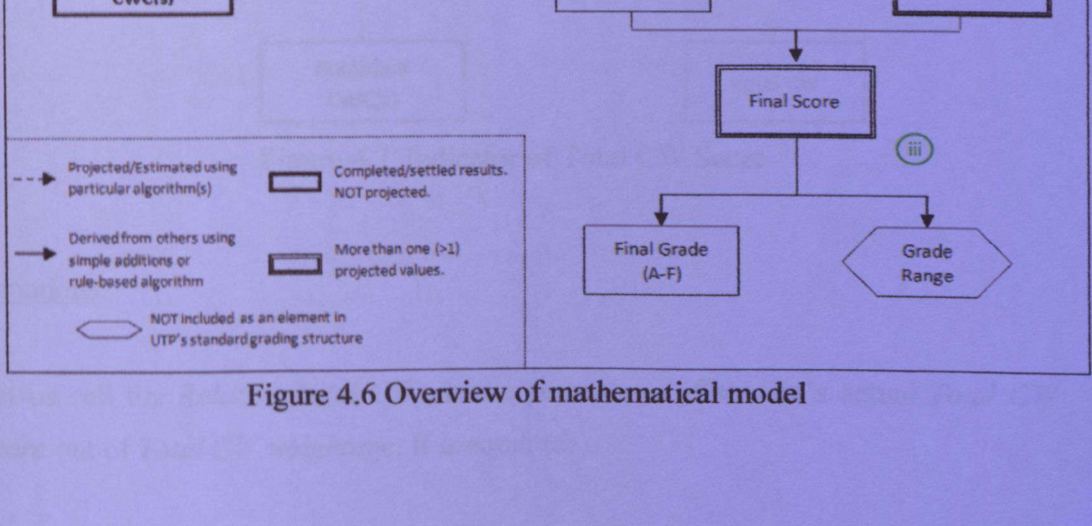


Figure 4.6 Overview of mathematical model

More details on mathematical equations used at each node are to be plainly explained in the subsequent sections, with reference to the following scenario.

Scenario:

"Structured Programming" is a 14-week programming course at UTP. The grading structure is 60% Coursework (CW) and 40% Final Exam.

The Total CW consists of four (4) components: Test 1, Assignment, Test 2, and Project in that exact order of time conducted. The percent weightage for these components are 10%, 20%, 10% and 20% respectively.

i. *Projecting Total CW Score*

Objective: to project *Total CW Score* based on the *Published CWCs*.

Figure 4.7 below shows main indicator of *Total CW Score*. It is extracted from Figure 4.6.

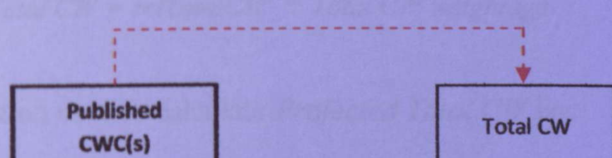


Figure 4.7 Indicator of Total CW Score

Equations:

Let us call the *Relative Total CW* is the percentage of student's actual *Total CW Score* out of *Total CW weightage*. It is equal to:

$$relTotalCW = \frac{Total\ CW}{Total\ CW\ weightage}$$

As a result, to calculate the *Projected Total CW,* we only need to calculate the *Projected Relative Total CW.*

Hence, first we calculate the *Projected Relative Total CW* from the *Published CWC(s)* using following formula:

$$\begin{matrix}Projected\\ relTotalCW\end{matrix} = \frac{Total\ scores\ of\ all\ Published\ CWC(s)}{Total\ weightages\ of\ those\ CWC(s)}$$

Given the scenario, when 'Test 1' and 'Assignment' results are published, the *Projected Relative Total CW* can be calculated as follow:

$$\begin{matrix}Projected\\ relTotalCW\end{matrix} = \frac{score(Test\ 1) + score(Assignment)}{10 + 20}$$

In which, 10 is the weightage of 'Test 1' and 20 is the weightage of 'Assignment'.

Then we calculate the *Projected Total CW* from the *Projected Relative Total CW* using the following formula:

$$Total\ CW = relTotalCW\ *\ Total\ CW\ weightage$$

In summary, the equation used to calculate *Projected Total CW* is :

$$\boxed{\begin{matrix}Projected\\ Total\ CW\end{matrix} = \frac{Total\ scores\ of\ all\ Published\ CWC(s)}{Total\ weightages\ of\ those\ CWC(s)}\ *\ Total\ CW\ weightage}$$

ii. Projecting Exam Score

<u>Objective</u>: to project *Exam Score* based on the *Projected Total CW*, *Difficulty level* of the exam paper and the user-selected *Published CW Components* which results are expected to be proportional with Exam Score.

Figure 4.8 below shows main indicators of *Exam score*. It is extracted from Figure 4.6.

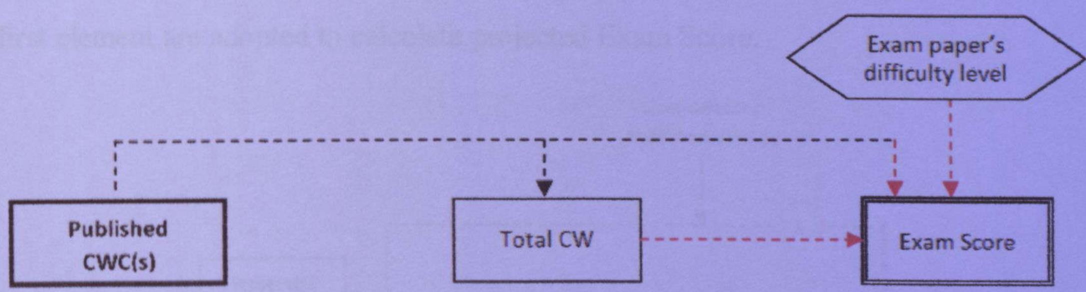


Figure 4.8 Indicators of Exam score

<u>Equations</u>:

Three(3) main elements that are adopted to calculate *Projected Exam Score* are:

- *Projected Total CW* : Hypotheses H1 and H3
- *Published CW Components* that are expected to be proportional to *Exam Score* (subjectively selected by user) : Hypothesis H2
- *Difficulty level* of the Exam paper (subjectively selected by user) : Hypothesis H4

The first(1^{st}) element, *Projected Total CW*, is adopted into the algorithm in two(2) complementary forms:

- *Projected Total CW Score* (out of 100) : Hypothesis H1

- *Projected CW Lost Score* (out of total CW weightage) : Hypothesis H3

Mathematically and logically, these two forms are actually one because *Lost score* is directly derived from *Total CW Score*:

CW Lost score = Total CW weightage - Total CW score

However, by using different approach on each form, the two outputs are complementary and together they contribute to a more accurate *Projected Exam Score*.

Figure 4.9 below illustrates how these three(3) elements and the two(2) forms of the first element are adopted to calculate projected Exam Score:

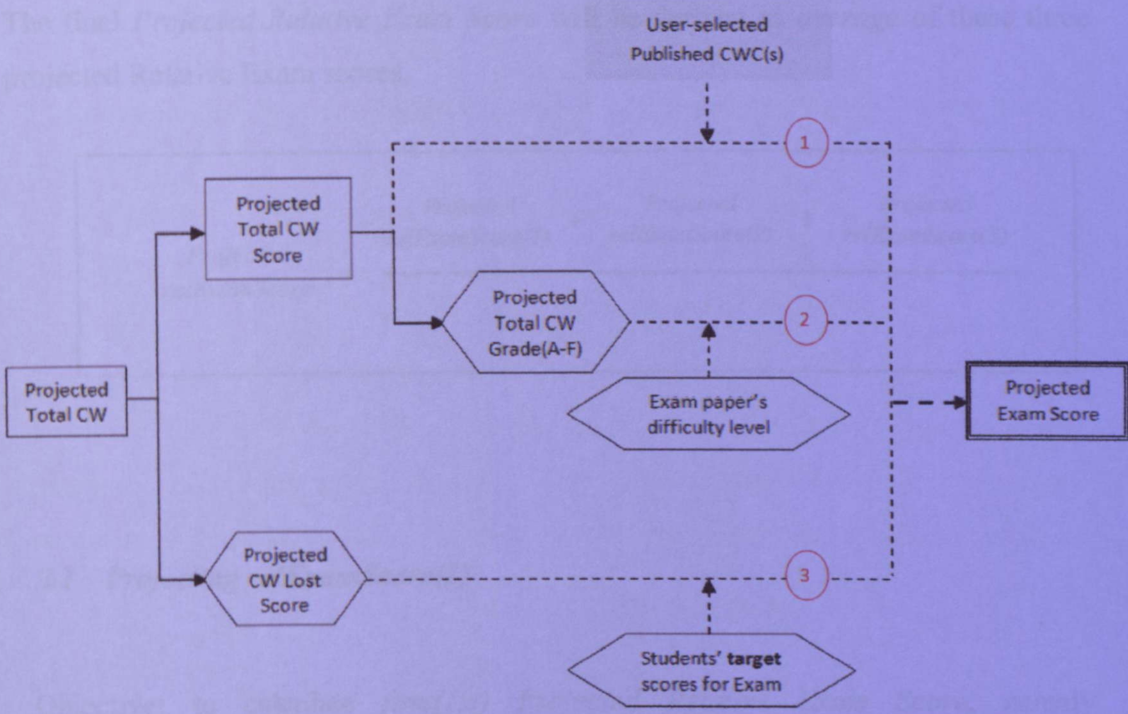


Figure 4.9 Revised Indicator(s) of Exam Score

Relative Exam Score is the percentage of student's actual *Exam Score* out of *Exam Weightage*. It is equal to:

$$relExamScore = \frac{\textit{Exam Score}}{\textit{Exam weightage}}$$

Then, the *Projected Exam Score* can be easily calculated from the *Projected Relative Exam Score*. Hence, our aim is to calculate this *Projected Relative Exam Score*.

For each of the three revised indicators above, we will calculate a corresponding *Projected Relative Exam Score*, namely relExamScore(1), relExamScore(2) and relExamScore(3)

The final *Projected Relative Exam Score* will be derived as *average* of these three projected Relative Exam scores.

$$\text{Projected } relExamScore = \frac{\text{Projected } relExamScore(1) + \text{Projected } relExamScore(2) + \text{Projected } relExamScore(3)}{3}$$

ii.1 Projecting relExamScore(1)

Objective: to calculate *first(1st) Projected Relative Exam Score*, namely *relExamScore(1)*, using the first revised indicator (*Projected Total CW Score* and *user-selected Published CWCs*).

Figure 4.10 below shows this first indicator of Exam Score. It is extracted from Figure 4.9.

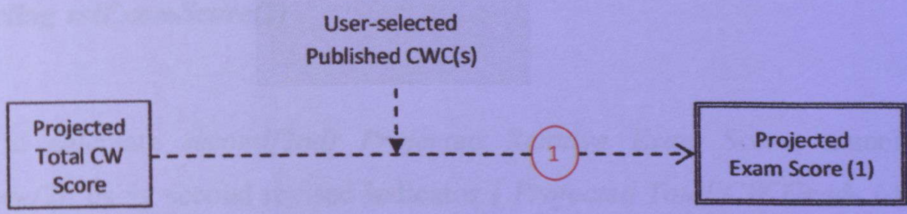


Figure 4.10 First indicator of Exam Score

Equations:

Based on Hypothesis H1, *Total CW Score* is expected to be proportional to *Exam score*.

Based on Hypothesis H2, only those *Published CWCs* that are expected to be proportional to *Exam Score* can be adopted to calculate *Projected Exam Score*.

Hence, the *first Projected Relative Exam Score* is calculated using following equation:

$$\boxed{\textit{Projected relExamScore(1)} = \frac{\textit{Projected Total CW + User-selected Published CWC(s)}}{\textit{Total CW weightage + Total weightages of those CWC(s)}}}$$

Given the scenario, when 'Test 1' and 'Assignment' results are published, the Projected Total CW Score is calculated as in previous section. Then, if among these two components, the lecturer (user) selects only 'Test 1' to be a good predictor of the Exam Score, the Projected relExamScore(1) is calculated as below:

$$\textit{Projected relExamScore(1)} = \frac{\textit{Projected Total CW Score + Score(Test 1)}}{60 + 10}$$

In which, 60 is the weightage for *Total CW* and 10 is the weightage for 'Test 1'. Also, the *Projected Total CW Score* was from previous section.

ii.2 Projecting relExamScore(2)

<u>Objective</u>: to calculate *second(2nd) Projected Relative Exam Score*, namely *relExamScore(2),* using second revised indicator (*Projected Total CW Grade* (A-F) and *user-selected Difficulty level* of exam paper).

Figure 4.11 below shows this second indicator of Exam score. It is extracted from Figure 4.9.

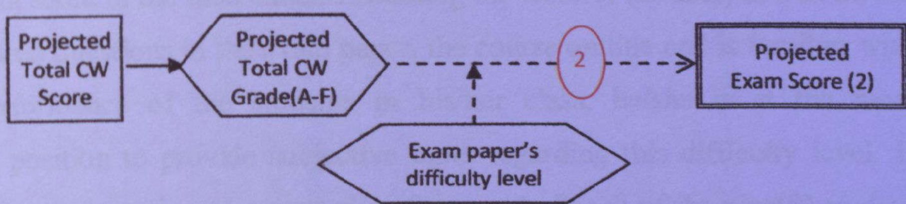


Figure 4.11 Second indicator of Exam score

<u>Equations:</u>

a. Total CW Grade(A-F)

From *Projected Total CW Score*, we can easily derived the *Projected Total CW Grade* accordingly, using a simple rule-based algorithm based on the standard grading scheme:

Table 4.5 UTP Grading Scheme

Score	Grade
85 – 100	A
80 – 84.9	A-
75 – 79.9	B+
65 -74.9	B
55 – 64.9	C+
50 – 54.9	C
45 – 49.9	D+
40 – 44.9	D
0 - 39.9	F

The "Score" we should use to compare against the first column is the Projected Relative Total CW (in percentage), not the Projected Total CW score. For example, if the Projected Relative Total CW is 0.73 (or 73%), we will use "73" as the score to classify the corresponding Grade. In that case, the corresponding grade is B.

b. Exam paper's difficulty level

The approved hypothesis H4 indicates that the difficulty level of exam paper does affect student score in the final exam. Assuming the lecturer (as user) is well aware of the prepared questions in the exam paper, the course outline and is familiar with average performance of the students in his/her class, he/she is at the most appropriate position to provide subjective input regarding this difficulty level. It reflects his/her perspective on expected performance of each of the nine(9) student groups (clustered based on their Projected Total CW Grade calculated in step a). Eg. group B+, group D+, group A-, etc.

The lecturer will be prompt to select one level from a qualitative scale of "Easy", "Moderately Easy", "Intermediate", "Moderately Difficult" and "Difficult".

○ Easy ○ Moderately easy ○ Intermediate ○ Moderately Difficult ○ Difficult

Figure 4.12 Qualitative scale of Difficulty level

In order to transfer this *qualitative scale* into *quantitative figures* to be used in the mathematical model, the writer adopts a *reference table* as showed in Table 4.6 below:

Table 4.6 Quantitative scale of Exam paper's difficulty level

	Expected Exam Score (*out of 100*)	
Student Group	**From**	**To**
A	L1	U1
A-	L2	U2
B+	L3	U3
B	L4	U4
C+	L5	U5
C	L6	U6
D+	L7	U7
D	L8	U8
F	L9	U9

There are three columns in this reference table. The first column is "Student Group" based on their *Projected Total CW Score*, which have been classified at ***step a.*** above. The next two columns are to capture *ranges of Exam Score* that the lecturer expects each of the student groups to score during the final exam. Specifically, students of group "A" are expected to score within L1-U1 range in their final exam*; L1 is the lower limit* and *U1 is the upper limit* for this group.

Each of the difficulty level in the qualitative scale (from "Easy" to "Difficult") is assigned with one reference table. Table 4.7 below shows a default reference table for user-ranked "Difficult" exam paper.

Table 4.7 Reference Table for "Difficult" Exam paper

	Expected Exam Score (*out of 100*)	
Student Group	**From**	**To**
A	75	95
A-	70	80
B+	65	75
B	50	65
C+	40	50
C	35	40
D+	30	40
D	25	35
F	5	25

Similarly, Table 4.8 below shows a default reference table for user-ranked "Easy" exam paper.

Table 4.8 Reference Table for "Easy" Exam paper

	Expected Exam Score (*out of 100*)	
Student Group	**From**	**To**
A	90	100
A-	90	100
B+	85	95
B	75	85
C+	65	75
C	60	70
D+	55	65
D	50	60
F	40	55

As noted, the difference between these two default reference table is the *ranges of Exam Score* that the lecturer expects each student group to get in the final exam. For example, with "Difficult" paper, the lecturer expects "Group F" student to score very low marks in the range of 5-25 out of 100; however, with "Easy" paper, he/she expects same group of students to score better in the range of 40-55 out of 100.

Both these two tables are *default tables* which are intuitively designed by the writer. In the developed system, the user is provided with an option to flexibly edit any of the reference table to fit his/her subjective perspective on the students' expected performance in final exam.

See Appendix B for all five(5) default reference tables for all the difficulty level.

Importance:

These quantitative tables utilize one of the most important elements, that helps the lecturers(users) to improve projected performance of the students, which is the lecturers' interactively knowledge on their students' capabilities, strengths and weaknesses.

For example, the lecturer may recognize the performance gap between this year class of students and last year class of students; hence, his subjective inputs for these two classes are most probably different. Also, within the same class, depending on the exam questions which are well aware by the lecturers, they can estimate a high result (eg. 85-100/100) for all B+ group and above, while other groups remain average result. In another special case, only group A students are expected to score over 90/100, while all other groups are expected to score much less.

As the course goes on, the user can revise their inputs to improve the accuracy of predictive outputs.

Capture input from reference table into the mathematical model:

There are three(3) values of the *second Projected Exam Score* that we can get from these reference table: lower limit (minScore), average (meanScore) and upper limit (maxScore).

$$minScore = L_i$$

$$maxScore = U_i$$

$$meanScore = (L_i + U_i) / 2$$

Respectively, we can easily derive three corresponding values of the *second* (2^{nd}) *Projected Relative Exam Score,* namely *relExamScore(2)*:

> *Projected relExamScore(2, minScore) = minScore/100*
> *Projected relExamScore(2, maxScore) = maxScore/100*
> *Projected relExamScore(2, meanScore) = meanScore/100*

In summary, from the *Projected Total CW Grade* grouping and user's *subjective inputs* on *Exam paper's difficulty level*, we can generate three(3) different values of the second(2^{nd}) *Projected Relative Exam Score*, namely: *relExamScore(2, minScore), relExamScore(2, maxScore)* and *relExamScore(2, meanScore).*

The reason we need to capture all these three values is because they will be used to generate a *Projected Final Grade Range*, which will be explained in further details in later section.

ii.3 Projecting relExamScore(3)

Objective: to calculate the *third(3rd) Projected Relative Exam Score*, namely *relExamScore(3),* using the third revised indicator (*Projected Total CW Lost Score* and its relationship with *students' target scores for Exam*).

Rationale: at the end of a course, the most important target that the students pay attention to is the Final Grade(A-F), not the Final Score (0-100). For example, a student who scores 85/100 has the *same final grade* of "A", which is equivalent to 4.0 points per credit hour, with another student who scores 98/100. The *approved* *hypothesis H3* indicates that there is a direct relationship between the Lost Score and students' target for their exam.

Figure 4.13 below shows this third indicator of Exam score. It is extracted from Figure 4.9.

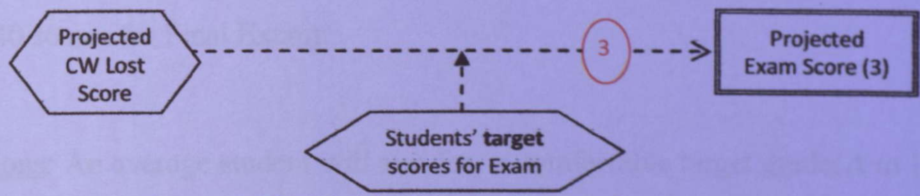


Figure 4.13 Third indicator of Exam Score

Equations:

From the previous UTP Grading Scheme table (Table 4.5), the writer intuitively generates two tables showing the direct relationship between *Total CW Score*, *Total CW Lost Score* and students' corresponding *Target Scores* in Final Exam.

Table 4.9 below is simply derived from Table 4.5. It shows *ranges of scores* that one student is allowed to lose (out of total 100 score for both *CW* and *Final Exam*) in order to secure a Grade (A-F).

Table 4.9 Final grade vs. Final lost score

Final Grade	Final Score	MinFinalLost	MaxFinalLost
A	85 - 100	0	15
A-	80 -84.9	15.1	20
B+	75 - 79.9	20.1	25
B	65 - 74.9	25.1	35
C+	55 - 64.9	35.1	45
C	50 - 54.9	45.1	50
D+	45 - 49.9	50.1	55
D	40 - 44.9	55.1	60
F	0 - 39.9	60.1	100

MinFinalLost: the minimum score one student may lose in their total final scores to still get certain grade.
MaxFinalLost: the maximum score one student can bare to lose in their total final scores to secure certain grade.

Specifically, given the scenario, if a student aims for an "A" grade at the end of the course, he is allowed to lose from 0 to 15 out of the total 100 scores (60 scores for CW and 40 scores for Final Exam).

<u>Assumptions</u>: An average student will aim for a comfortable target grade(A to F), not target final score, which most suits his/her capabilities, studying times and efforts.

With this assumption being stated, Table 4.10 below show how a student may aim for different final grade based on his/her CW Lost Score.

Table 4.10 CW Lost score vs. Targeted Final Grade

CW Lost Score	Targeted Final Grade
0 - 9.9	A
10 - 14.9	A-
15 - 19.9	B+
20 - 24.9	B
25 - 29.9	C+
30 - 39.9	C
40 - 44.9	D+
45 - 54.9	D

Meaning that if one student lost 12 out of 60 scores for Total CW (CW Lost Score = 12), he would comfortably aim for an "A-" Grade (Table 4.10). Then, in order to get that "A-", he is allowed to lose 15.1 - 20 scores out of total 100 scores (Table 4.9) throughout the whole course including final exam.

Next, noted that:

Targeted Lost Score in Exam = Targeted Final Lost score - CW Lost Score

Hence,

TargetedMaxExamLost = TargetedMaxFinalLost - CW Lost Score

TargetedMinExamLost = TargetedMinFinalLost - CW Lost Score

From the above Table 4.9 and Table 4.10 and with same approach as in **iii.2**, we can calculate the three(3) values of the *Projected Exam Score* as following:

minScore = Exam weightage - TargetedMaxExamLost
maxScore = Exam weightage - TargetedMinExamLost
meanScore = (minScore + maxScore) /2

For example, the above student would aim for an Exam Score in the range of 32 - 36.9 (out of 40 scores allocated for Final Exam , a.k.a. Exam weightage).

minScore = 40 - TargetedMaxExamLost = 40 - (20 - 12) = 32

maxScore = 40 - TargetedMinExamLost = 40 - (15.1 - 12) = 36.9

meanScore = (32 + 36.9) / 2 = 34.45

Respectively, we can easily derived three values of the third(3^{rd}) *Projected Relative Exam Score,* namely *relExamScore(3)*:

Projected relExamScore(3, minScore) = minScore/100
Projected relExamScore(3, maxScore) = maxScore/100
Projected relExamScore(3, meanScore) = meanScore/100

In summary, from the *Projected Total CW Lost Score* and students' *Target Score in Final Exam* accordingly, we can generate three(3) different values of the third(3^{rd}) *Projected Relative Exam Score*, namely: *relExamScore(3, minScore), relExamScore(3, maxScore)* and *relExamScore(3, meanScore).*

**** Final step in calculating the Projected Exam Score from relExamScore(1), relExamScore(2) and relExamScore(3)***

By utilizing the *Projected Total CW, user-selected Published CWCs, Exam paper's level of difficulty* and *Projected CW Lost Score*, the writer used different equations to calculate three major *Projected Relative Exam Score*, namely relExamScore(1), relExamScore(2) and relExamScore(3).

The final *Projected Relative Exam Score*, which then is used to calculate the *Projected Exam Score*, is average of all three Relative Exam Scores. This is applied to relative minScore, relative maxScore and also relative meanScore.

$$\begin{array}{c}\textit{Projected}\\ \textit{relExamScore}\end{array} = \frac{\begin{array}{c}\textit{Projected}\\ \textit{relExamScore(1)}\end{array} + \begin{array}{c}\textit{Projected}\\ \textit{relExamScore(2)}\end{array} + \begin{array}{c}\textit{Projected}\\ \textit{relExamScore(3)}\end{array}}{3}$$

As mentioned in Figure 4.6 (overview), the *Projected Exam Score* will have more than one value, which gives a range for possible Final Score and possible Final Grade that one student may get. These multiple values are derived from the three values of relExamScore(2) and relExamScore(3) : minScore, maxScore and meanScore.

The *Projected Average Exam Score* is calculated from the the *Projected Relative Exam MeanScore*:

*Projected Average Exam Score = relMeanExamScore * Exam weightage*

Similarly for the *Projected Minimum and Maximum Exam Score*:

*Projected Min Exam Score = relMinExamScore * Exam weightage* *Projected Max Exam Score = relMaxExamScore * Exam weightage*

In summary, the *Projected Exam Score* of a student is a range from the *Projected Min Exam Score* to the *Projected Max Exam Score* as calculated above. The *Projected Average Exam Score* is a single value to represent this range.

iii. Final Score, Final Grade and Grade Range

iii.1 Final Score

Objective: to derive *Projected Final Score* from the *Projected Total CW Score* and the *Projected Exam Score*.

Figure 4.14 below shows the elements:

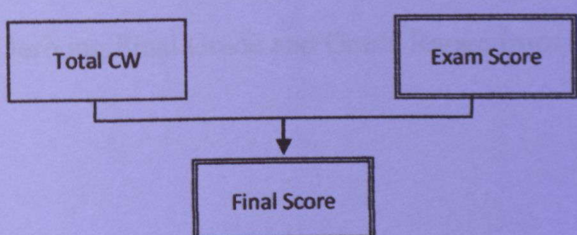


Figure 4.14 Elements of Final Score

Equations:

As there are three different values of the *Projected Exam Score*, there are also three(3) values of the *Projected Final Score* accordingly:

Projected Average Final Score = Projected Total CW + Projected Average Exam Score

Projected Min Final Score = Projected Total CW + Projected Min Exam Score

Projected Max Final Score = Projected Total CW + Projected Max Exam Score

However, as to show one single projected value to the user on the system, the *Projected Average Final Score* is selected and is displayed in "Final Score" column. The other Min and Max values are displayed in another column called "Grade Range".

iii.2 Final Grade & Grade Range

Figure 4.15 below shows the final step:

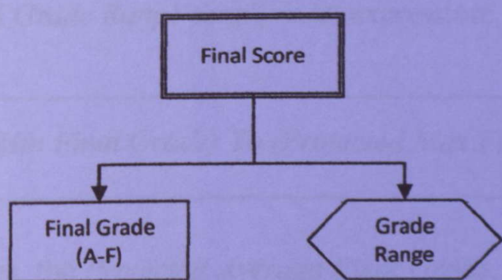


Figure 4.15 Deriving Final Grade and Grade Range from Final Score

a. Final Grade:

Objective: to derive *Projected Average Final Grade* from the *Projected Average Final Score.*

Algorithm:

Similar to deriving the *Projected Total CW Grade(A-F)* from *the Projected Total CW Score*, we can easily derived the *Projected Final Grade* easily using a simple rule-based algorithm based on the standard grading scheme (refer to Table 4.5)

b. Min Final Grade, Max Final Grade and Final Grade Range

Objective: to derive *Projected Min Final Grade*, *Projected Max Final Grade*, and *Projected Final Grade Range* from the relevant *Projected Final Scores*.

Algorithm:

Similar to the above *Projected Average Final Grade*, we can also easily apply the rule-based algorithm to derive the *Projected Min Final Grade* and the *Projected Max Final Grade*.

Then, the *Projected Final Grade Range* simply is an expression:

*"**From** (Projected Min Final Grade) **To** (Projected Max Final Grade)"*

Importance: In some cases, the *Projected Average Final Grade*, with only a single value, may not be sufficient to cover all possible end results (final grades) of a student's performance. Hence, this *Projected Grade Range* will provide a wider range of expected final grades for each student, with lower degree of exactness yet higher degree of accuracy for predictive outputs.

For example, one student may have the *Projected Average Final Grade* is "A-", yet his/her expected full Grade Range is *"From A- to A";* meaning that though lower, there is still chance he/she will get an A. The developed system is able to cover all of possible outputs.

<u>Note</u>: In many cases, the *Projected Min Final Grade* <u>is the same as</u> the *Projected Max Final Grade* ; then the *Projected Average Final Grade* 's single value is the only projected output of the system and the *Projected Grade Range* is set to "N/A" - Not Applied.

4.3 Prototype Testing

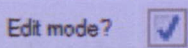
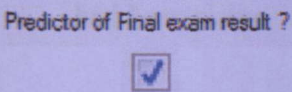
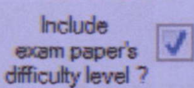
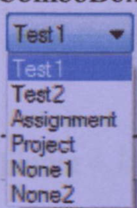
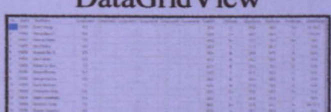
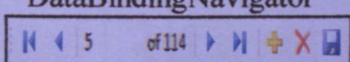
4.3.1 Functionality testing

Objective: to test functionality of each component of the system.

Result:

Table 4.11 Functionality Test result

System component	Expected Function	Status
Button Edit Component List	To open "Edit Component List" window; then to allow users to edit the number of CW components and their weightages	Fully functioning
Button Load Total CW	To project students' *Total CW score* from all available *Published CW Components.*	Fully functioning
Button Load Exam Score & Final Results	To project students' *Exam Score* and their *Final Score*, *Final Grade* and *Grade Range*.	Fully functioning
Button View Summary	To open "Summary" window; then to allow users to view summarized table, graphs of predictive outputs.	Fully functioning
Button Save & Close	To save users' updates on DataGridView table and then close the corresponding window.	Fully functioning
CheckBox Complete ?	To set status for each CW component. If it is checked, all students' published score in certain CW component are	Fully functioning

	entered into the DataGridView table and are updated to database.	
CheckBox 	If it is checked, users are allowed to enter, update or delete students' score in the corresponding *Published CW component*.	Fully functioning
CheckBox 	If it is checked, the corresponding *Published CW Component,* as compared to other components, is incorporated as a good predictor *Exam Score*.	Fully functioning
CheckBox 	If it is checked, "Exam paper's difficulty level" window is open. Then, the users are prompt to enter their subjective input to improve accuracy level of predictive outputs.	Fully functioning
ComboBox 	The list of items is loaded into this ComboBox from the user-edited "Component List" table. Then, the users are to select the name of the CW Component column in relation with the "CW Data" DataGridView table.	Fully functioning
DataGridView 	To link the system with Access database tables and to allow users to insert, update, delete the records.	Fully functioning
DataBindingNavigator 	To support users to navigate through the records in DataGridView tables. Users can add records, delete records and save latest updates into the database.	Fully functioning

4.3.2 Accuracy testing

Objective: to test accuracy level of this predictive model's outputs.

Test data: the writer used detailed courseworks and final exam scores of 114 students in a 14-week programming course at Universiti Teknologi PETRONAS. It was provided by UTP Exam Unit. The name of the course had been purposely not mentioned here as requested by the Exam Unit.

The course consisted of 60% Coursework(CW) and 40% Final Exam. There were four(4) CW components in following timely order:

Test 1 (10%) → Test 2 (10%) → Assignment (20%) → Project (20%)

Test 1 was the first published CW component, then was Test 2, Assignment , and Project was the last one.

Figure 4.16 below shows a snapshot of the test data:

ID	Test 1	Test 2	Assignments	Projects	Total Coursework	Final Exam	Final score	Grade
1	5.00	5.33	0.00	7.87	18.20	21.80	40.00	D
2	5.60	6.67	15.50	12.53	40.30	27.60	67.90	B
3	7.50	6.67	17.50	15.23	46.90	25.40	72.30	B
4	4.00	4.33	8.00	13.87	30.20	18.80	49.00	D+
5	6.50	5.00	8.00	10.53	30.03	19.20	49.23	D+
6	6.80	7.00	16.50	15.60	45.90	28.80	74.70	B
7	7.00	8.67	18.50	17.27	51.43	32.60	84.03	A-
8	5.20	6.67	17.50	14.87	44.23	29.40	73.63	B
9	6.60	7.33	18.50	11.80	44.23	35.60	79.83	B+
10	7.10	7.67	18.00	16.67	49.43	29.20	78.63	B+

Figure 4.16 Snapshot of tested data

Procedure:

From students' scores in the four(4) CW components, the writer attempted to use the developed predictive model to predict students' *Total Coursework* score, *Final Exam* score and ultimately their *Final Score*. Then, the *projected Final Score* were compared against students' actual *Final Score* available in the test data to calculate the accuracy level.

For each of the 114 students, the deviation percentage of the projected Final Score was calculated as following:

$$\text{Deviation (\%)} = \frac{\text{Difference between Acutal Final Score and Projected Final Score}}{\text{Actual Final Score}} * 100$$

Figure 4.17 below shows a snapshot of the projected values and corresponding deviation percentages:

ID	Final score	Projected Final Score	Deviation(%)
1	40.00	26.23	34.4
2	67.90	68.29	0.6
3	72.30	79.97	10.6
4	49.00	50.99	4.1
5	49.23	51.65	4.9
6	74.70	79.13	5.9
7	84.03	86.60	3.1
8	73.63	74.89	1.7
9	79.83	75.41	5.6
10	78.63	82.02	4.3
11	70.43	77.96	10.7

Figure 4.17 Snapshot of predictive model's output

Then the **mean deviation percentage** of the model is calculated as the **average** of all the deviation percentages. Finally, the model's accuracy level is

$$\text{Model's Accuracy level (\%)} = 100 - \text{Mean Deviation Percentage}$$

$$= 100 - \frac{34.4 + 0.6 + 10.6 + 4.1 + 4.9 + 5.9 + 3.1 + \dots}{114} = 93.1\ (\%)$$

<u>Results:</u>

Table 4.12 below summarizes the accuracy testing result:

Table 4.12 Accuracy Testing result

	Test 1	**Test 1 + Test 2**	**Test 1 + Test 2 + Assignment**	**Test 1 + Test 2 + Assignment + Project**
<u>Before</u> including *Test 1* & *Test 2* (if available) as good predictor(s) of exam score	83.0%	86.9%	90.3%	92.6%
<u>After</u> including *Test 1* & *Test 2* (if available) as good predictor(s) of exam score	83.0%	86.9%	90.8%	93.1%

Improved accuracy level

Improved accuracy level

Using the same set of test data, the predictive model was run 8 times with 8 different settings as showed in the table.

When *Test 1* result was published, two tests was conducted with 2 different settings: Before and After including Test 1 as a good predictor of exam score. (refer to section 4.3.3.ii.1 - projecting relExamScore(1))

Next, when *Test 2* result was published, we had both Test 1 and Test 2 result now. Another two tests was conducted also with the 2 different settings: Before and After including Test 1 & Test 2 as a good predictor of exam score.

Similar process was applied when the result of third component, *Assignm*ent, was published.

Finally, when *Project* result is published (all four CW components had been completed), the final two tests were conducted.

In total, the writer had conducted 8 tests and the accuracy levels were summarized in Table 4.11 above.

Conclusion:

- Throughout the course, as more and more CW components were published, the accuracy level of the predictive model increased.
- With same set of published CW component(s), the predictive model was approved more accurate after including *Test 1 & Test 2* (if available) as good predictor(s) of exam score.

- The average accuracy level of the developed model was around **90%**. Especially when <u>all</u> CW components had been published and <u>after</u> including *Test 1 & Test 2* as good predictors of exam score, the accuracy level reached its max of **93.1 %,** a desirably high level of accuracy.

<u>Notes:</u>

- In the programming course from which the writer collected this test data, the course's lecturer had inputted that among the four(4) CW components, only *Test 1* and *Test 2* were the two good predictors of her students' exam score. This was subjective input from the user and the result has proved that it helps improve the accuracy level of predictive outputs.
- Exam paper's difficulty level was not incorporated in the model during the test because the lecturer was not able to remember performance of each specific student group (A-F) in her class. Hence, she decided to not include the difficulty level when projected students' result. This was possible because the system allowed users to select whether or not to include the difficulty level into the projection. If the user had decided to include it, the better her subjective input on expected exam score of the students was, the more accurate the predictive outputs were. It could be more or less than 93.1 % depending on user input.

4.4 Final evaluation of the three(3) fundamental ideas and the five(5) hypotheses

Based on the survey result which lead to the first evaluation of the hypotheses (Table 4.4) and the accuracy testing result which proved a high level of accuracy (Table 4.11), the writer had successfully justified his proposed three(3) fundamental ideas (refer to 3.2.2) :

within the scope of this study, - **P1**: Excessive coursework breakdown analysis is SUFFICIENT for acceptable predictive outputs. - **P2**: Lecturers'/users' SUBJECTIVE INPUTS (such as which coursework components are good predictors of final exam score and how final exam paper's difficulty level affects students' scores), are helpful to improve the accuracy of outputs. - **P3**: Students' demographic profiles and behavioral contents are NOT necessary to be included into the data sources.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

This project had successfully developed a short-term predictive model which served as a relatively convenient tool for UTP lecturers to fully-utilize richly informative and readily available coursework data for predicting students' final performance. With straightforward yet fully-functioning design, the system fulfills the project's objective of developing a suitable software application for lecturers' use. Two main functional requirements, which are performance prediction and performance monitoring, had been tested successfully; also, the accuracy test had showed a high level of accuracy in predictive outputs. The developed system, if to be implemented in real OBE environment, promised to greatly support educators to systematically analyze, predict and continuously monitor students' performance throughout a short-term course, in order to provide timely intervention and adjustment. Ultimately, the system contributed itself to help educators mitigate student performance gap and achieve OBE's objectives.

By successfully adopting a creative approach with a simplified set of educational data sources and another crucial addition of dynamically subjective inputs from the users, the writer had initially justified a promising new trend in short-term prediction techniques. Using only coursework components as primary input, while no complex students' demographic profile and behavioral contents factors are directly considered, the model delivers its promised advantage of omitting the burden of heavy loads of input data. This is expectedly in favor of average users with fundamental needs to systematically and accurately predict and monitor students' performance.

A highly-customized mathematical model was constructed to facilitate the system design. Especially, the writer managed to create a unique method to quantitatively measure the existing influence of exam paper' difficulty level on students' expected performance in final exam.

Apart from all the advantages, the developed system still carried some limitations to be resolved.

5.2 Limitations

Partly due to technical, economical and timely constraints, following are some limitations existed within the system:

- Moderate efforts from the users are expected in order to continuously provide their subjective inputs, which is one of the core success factors of the system.
- Embedded database management functions are stopped at very basic level which are adding, deleting, updating and sorting.
- Network layer is not included into the system architecture. The developed model works separately and independently as an offline standalone window application.
- The scope is limited to programming courses at Universiti Teknology PETRONAS.

5.3 Recommendations

Future works suggest including a more advanced database management system, which has more complex functions of data filtering, searching, dynamic views, etc. embedded into the application. Also, integrated network solutions such as monitoring students' performance in simultaneous short-term courses in one academic semester are there to be developed.

Besides, the scope of the project can also be extended to other disciplines and other academic institutions, with proper adjustment relating to the *hypotheses* and the *mathematical model* for each particular case.

REFERENCES

1. Butler, M., 2004. Outcomes Based/ Outcomes Focused Education . Retrieved Oct 16, 2011 from http://www.kfshrc.edu.sa/saudization/files/Outcomes%20Based%20Education.doc
2. What is OBE. Halcyon House. Retrieved Oct 18, 2011 from http://www.halcyon.org/obeinfo.html
3. Ayesha, S., Mustafa, T., Sattar, A.R., and Khan, M.I., 2010. Data Mining Model for Higher Education System. European Journal of Scientific Research, 43(1): 24-29.
4. Han,J., Kamber, M., 2006. Data Mining: Concepts and Techniques. 2nd edition. The Morgan Kaufmann Series in Data Management Systems, Jim Gray, Series Editor.
5. Educational Data Mining. Retrieved October 18, 2011 from http://www.educationaldatamining.org/
6. Baker, R.S.J.D., Yacef, K., 2009. The State of Educational Data Mining in 2009: A Review and Future Visions. Journal of Educational Data Mining, 1(1): 3-17.
7. Fishman, T., 2011. Mathematical Modeling or Data Mining. Retrieved Jan 14, 2012 from http://quant.stackexchange.com/questions/1501/which-approach-dominates-mathematical-modeling-or-data-mining
8. Choudhury, A., 2011. Basic of Mathematical Model. Retrieved Jan 12, 2012 from http://gonitsora.com/basic-of-mathematical-modeling
9. Donald, D. D., 2005. A Closer Look: Closing The Performance Gap. University of Kansas Center for Research on Learning. Retrieved Dec 18, 2011 from www.all4ed.org/files/ACloserLook.pdf
10. Ogor, E.N., 2007. Student Academic Performance Monitoring And Evaluation Using Data Mining Techniques. Department of Natural Sciences. Turks & Caicos Islands Community College.
11. Merceron, A., Yacef, K., 2005. Educational Data Mining: A Case Study. Pôle Universitaire Léonard de Vinci, France and School of Information Technologies, University of Sydney, Australia.

12. White, B., Larusson, J., 2009. Seeing, Thinking, Doing: Strategic Directives for Learning Management Systems. Brandeis University, USA.

13. Rusli, N.M., Ibrahim, Z., Janor, R.M., 2008. Predicting Students' Academic Achievement Comparison between Logistic Regression, Artificial Neural Network, and Neuro-Fuzzy. Faculty of Information Technology & Quantitative Science. Universiti Teknologi MARA Malaysia.

14. Oladokun, V.O., A.T. Adebanjo, and O.E. Charles-Owaba. 2008. Predicting Students' Academic Performance using Artificial Neural Network: A Case Study of an Engineering Course. Pacific Journal of Science and Technology, 9(1):72-79.

15. Ali, N., Jusoff, K., Ali, S., Mokhtar, N., and Salamat, A.S.A, 2009. The Factors Influencing Students' Performance At Universiti Teknologi MARA Kedah, Malaysia. Management Science and Engineering, 3(4): 81-90. Canadian Research & Development Center of Sciences and Cultures.

16. Class Attendance. Center for Excellence in Teaching and Learning (CELT). Retrieved October 17, 2011 from http://www.mnsu.edu/cetl/teachingresources/articles/classattendance.html

17. Rapid Application Development. Retrieved October 20, 2011 from http://www.jscgroup.com/rapid-application-development.html

18. Data Mining Technology. Retrieved Jan 12, 2012 from http://www.executionmih.com/data-mining/technology-architecture-application-frontend.php

19. Princeton WorldNet Dictionary. Retrieved Nov 22, 2011 from wordnetweb.princeton.edu/perl/webwn

20. Glenn, 2007. Likert Scale & Surveys – Best Practices. Retrieved Jan 18, 2012 from http://intelligentmeasurement.wordpress.com/2007/11/20/likert-scale-surveys-best-practices/

APPENDIX A - User Interfaces

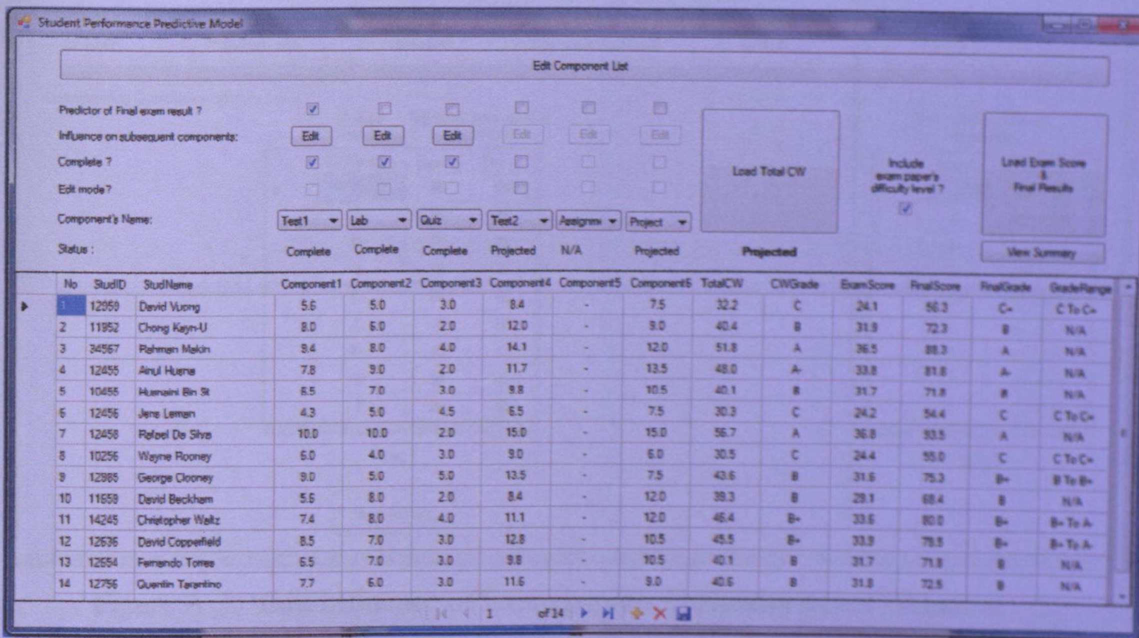


Figure A-1: Main Window

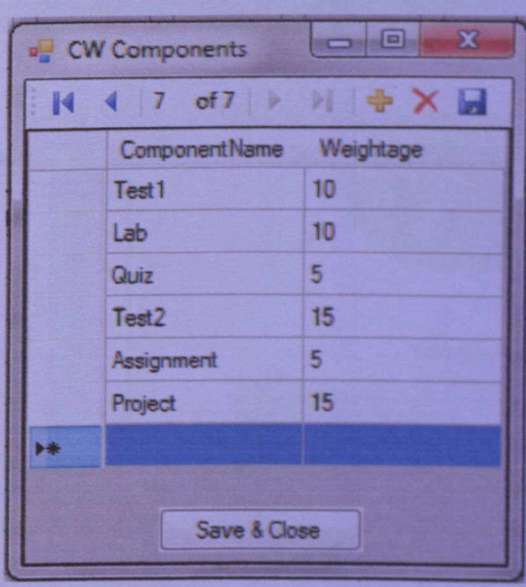


Figure A-2: 'Edit CW Components' Window

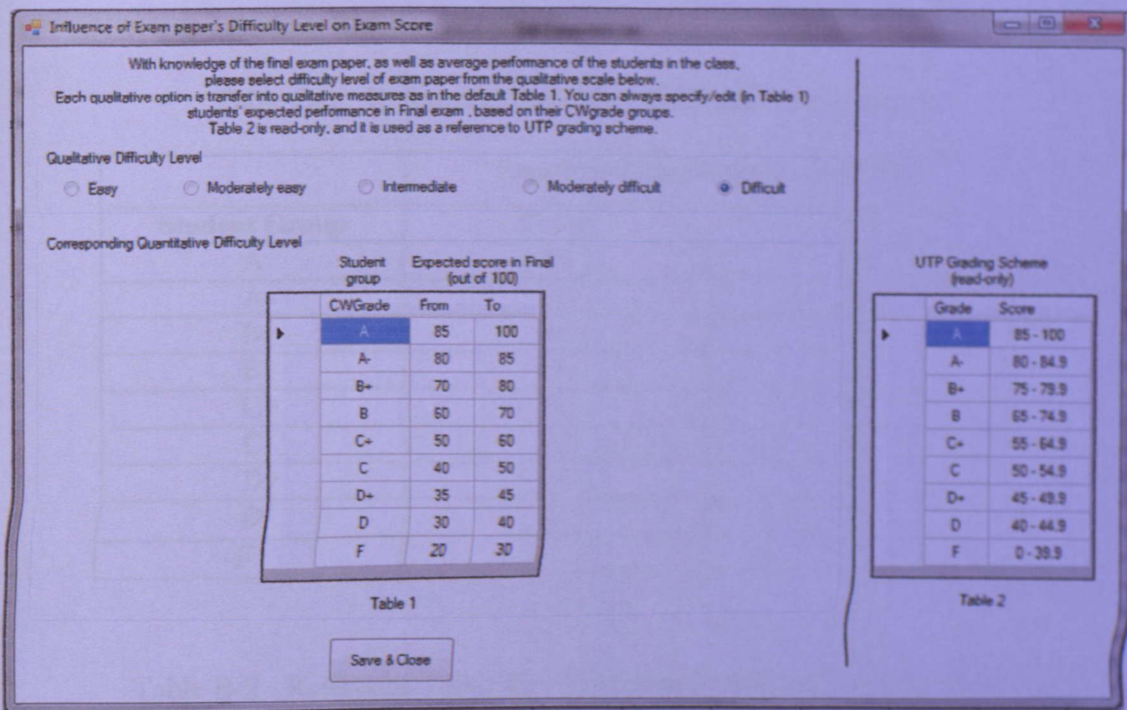


Figure A-3: 'Edit Influence of Exam paper's Difficulty Level' Window

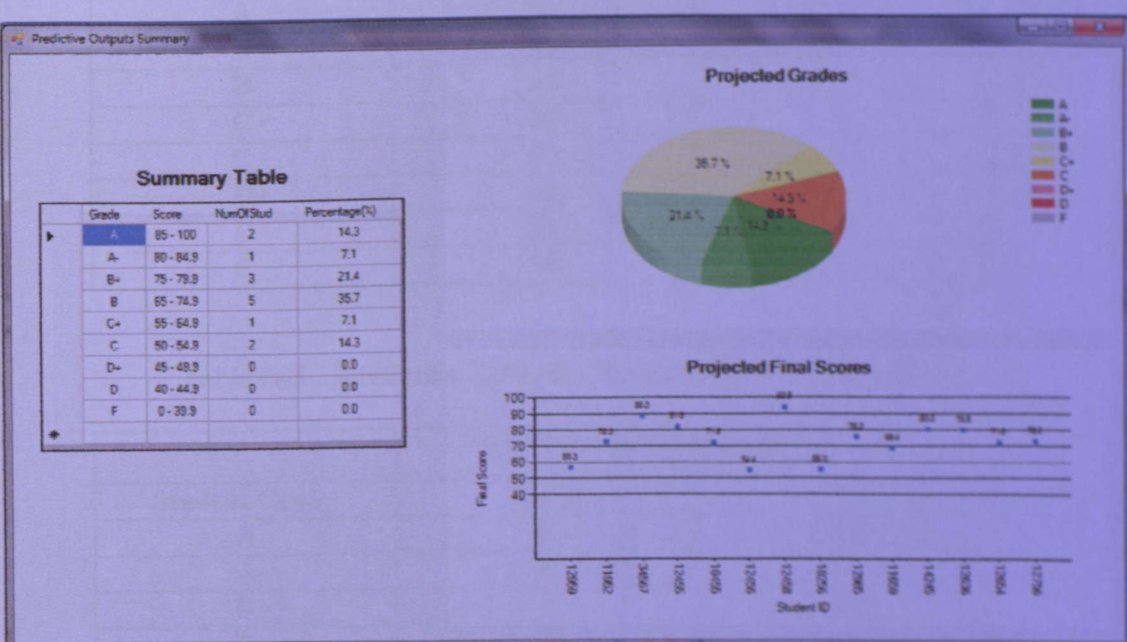


Figure A-4: 'Predictive Outputs Summary' Form

APPENDIX B - Default Reference Tables

Table B-1 Reference Table for "Difficult" Exam paper

	Expected Exam Score (*out of 100*)	
Student Group	**From**	**To**
A	75	95
A-	70	80
B+	65	75
B	50	65
C+	40	50
C	35	40
D+	30	40
D	25	35
F	5	25

Table B-2 Reference Table for "Moderately difficult" Exam paper

	Expected Exam Score (*out of 100*)	
Student Group	**From**	**To**
A	80	100
A-	75	85
B+	70	80
B	60	75
C+	50	65
C	45	55
D+	40	50
D	35	45
F	10	35

Table B-3 Reference Table for "Intermediate" Exam paper

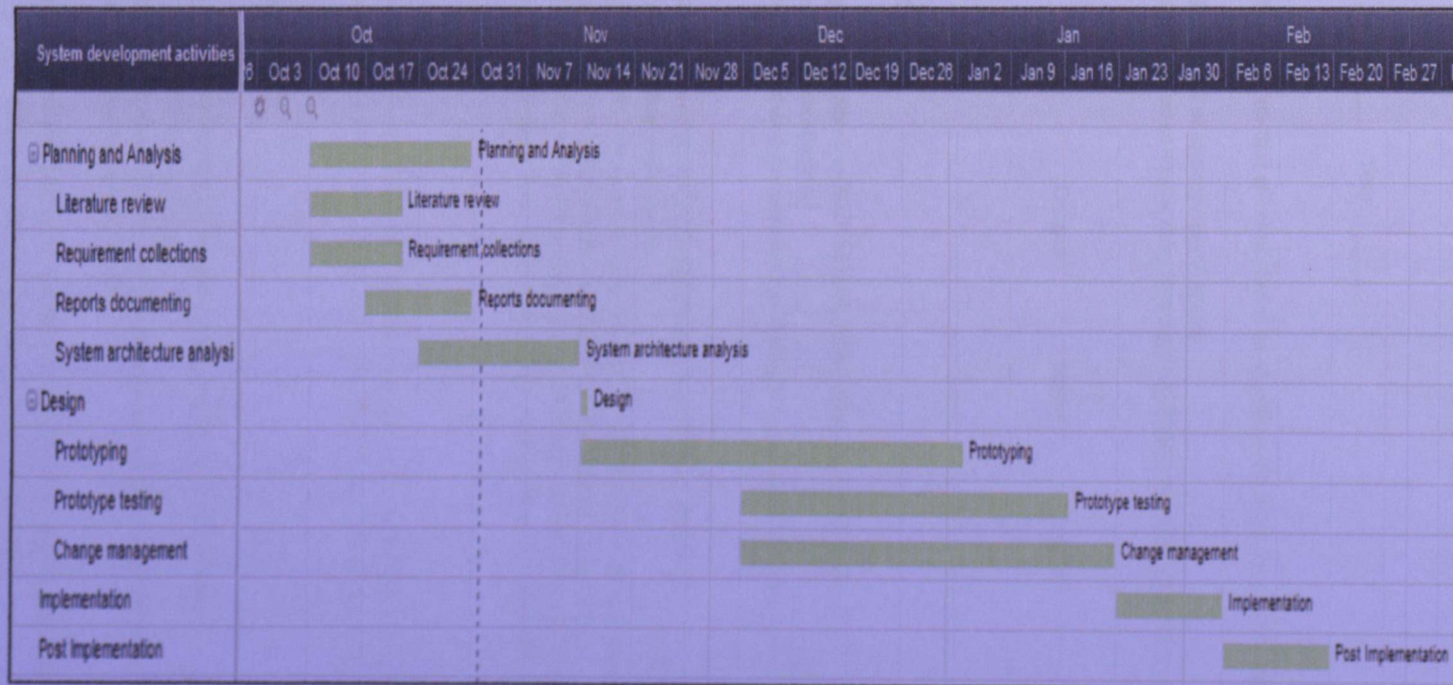
	Expected Exam Score (*out of 100*)	
Student Group	**From**	**To**
A	85	100
A-	80	85
B+	75	80
B	65	75
C+	55	65
C	50	55
D+	45	50
D	40	45
F	10	40

Table B-4 Reference Table for "Moderately easy" Exam paper

	Expected Exam Score (*out of 100*)	
Student Group	**From**	**To**
A	90	100
A-	85	90
B+	80	85
B	70	80
C+	60	70
C	55	60
D+	50	55
D	45	55
F	25	45

Table B-5 Reference Table for "Easy" Exam paper

	Expected Exam Score (*out of 100*)	
Student Group	**From**	**To**
A	90	100
A-	90	100
B+	85	95
B	75	85
C+	65	75
C	60	70
D+	55	65
D	50	60
F	40	55



APPENDIX C - Project Development Gantt Chart

APPENDIX D - Survey

FACTORS THAT AFFECT STUDENTS' ACADEMIC PERFORMANCE IN SHORT-TERM COURSES

BASIC INFO

Programme: ____________________ Year:________

SURVEY OVERVIEW

Purpose: To identify MAJOR FACTORS that affect students' academic performance in SHORT-TERM courses[1] (in months)

Scope: PROGRAMMING COURSES at undergraduate levels.

Eg. Structured Programming, Object-Oriented Programming, Internet Programming, Business Application Programming, etc.

	1	2	3	4	5
Scale:	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree

MAIN CONTENT

A. *TOTAL COURSEWORK MARKS*

1. One's coursework marks is *positively related*[2] to his/her final exam score, in terms of relative percentage to maximum allocated. 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

   E.g. One who scores 80-90%, out of total coursework marks allocated, is most expected to also score 80-90% in his/her final exam.

2. Given the *standard A-F grading structure*[3], total marks lost in his/her coursework DOES affect his/her target for final exam scores. 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

   E.g. One who lost 20-25% over 60% coursework is most probably to be satisfied with a **B** in final result, given him/her much less pressure preparing for the final exam, since he can afford to lose up to 15% out of 40%.

   Similarly, one who lost 10-13% out of total coursework (scored 47-50% over 60% allocated) most probably set his/her target for **A** (primary) or **A-** (secondary), which allows him/her to lose 7%, at max, over 40% allocated in the final.

   Comment (if any):

   ____

   ____

B. *COURSEWORK COMPONENTS*[4]

3. For those components WITH similar type of questions to final exam's, one's scores are *positively related* to his/her final exam score. 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

   E.g. One who scores well in '*Lab exercises*" (practical programming questions) is most expected to also score well in a final exam paper which consists of similar type of questions.

4. For those components WITHOUT similar type of questions to final exam's, one's scores are NOT DIRECTLY related to final exam score. They are *indirectly related* through total coursework instead, as mentioned in Section A. 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

   E.g. One's score in '*Group project*' is not directly related to his/her score in final exam.

1. Courses that finish in weeks/months. Eg. 14-week 'Structure Programming' course is offered to students in their first semester.
2. Positively related : change parallel in same direction. The first increases, the second increases too.
3. Grade : A (85-100), A- (80-84.9), B+ (75-79.9), B (65-74.9), C+ (55-64,9), C (50-54.9), D+ (45-49.9), D (40-44.9), F (0-39.9)
4. Including Tests, Lab exercises, Assignments, Quizzes, Group project, etc.

C. *FINAL EXAM PAPER* [5]

5. Covered scope, the broadness of learning concepts (chapters, references, etc.) covered in the exam, is *negatively related*[6] to students' final exam score. 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

6. Complexity level of the questions, the quality of each question to be compounded in terms of multiple learning concepts involved, is *negatively related* to students' final exam score. 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

7. Originality level of the questions, the quality of being new in the way lecturers apply taught concepts to the questions, is *negatively related* to students' final exam score. 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

8. Time requirements, average time to complete the paper as compared to the standard allocated 2-3 hours per paper, is *negatively related* to students' final exam score. 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

Comment (if any):

____

____

D. *OTHER FACTORS*

9. For **short-term** courses, the impact of students' demographic profile (gender, race, family background, education background,etc.) can be INDIRECTLY reflected in their courseworks. Prediction of students' final exam performance can IGNORE direct effects of these factors and consider merely *coursework marks* where their indirect effects tell. 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

10. For **short-term** courses, the impact of students' behavioral contents (attendance, involvement in extra-curriculum activities, etc.) can be INDIRECTLY reflected in their courseworks. Prediction of students' final exam performance can IGNORE direct effects of these factors and consider merely *coursework marks* where their indirect effects tell. 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐

Comment (if any):

____

____

** END OF SURVEY **

Thanks for your time and efforts.

Prepared by
VUONG TAN DAT
Computer and Information Sciences Department
Universiti Teknologi PETRONAS
Bandar Seri Iskandar, 31750 Tronoh, Perak, Malaysia
vuongtandat197@gmail.com
014-944-3515

5. All factors under this section are SUBJECTIVELY EVALUATED by instructors/lecturers themselves. The evaluation, though not from students' perspective but lecturers' own, is based on the average performance, understanding and capability of students in their classes.
6. The first increases, the second decreases.