Elementary and Secondary Classroom Management Using Expert System

A Thesis
Presented to
the Faculty of Graduate School
Samar State University
Catbalogan City, Samar

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in Information Technology

JOHANNA C. FLORA

May 2020

APPROVAL SHEET

This thesis entitled **ELEMENTARY AND SECONDARY CLASSROOM MANAGEMENT USING EXPERT SYSTEM,** has been prepared and submitted by **JOHANNA P. CABALHIN-FLORA** who having passed the comprehensive examination, is hereby recommended for oral examination.

SWEET MERCY P. PACOLOR
Associate Dean, College of Arts and Sciences
Adviser

Approved by the Committee on Oral Examination on July 30, 2020 with a rating of PASSED.

ESTEBAN A. MALINDOG JR., Ph. D.
Dean, Graduate School, SSU
Chairman

RONALD L. ORALE, Ph.D.
Vice President for Research & Extension Services
Member

FELISA E. GOMBA, Ph.D.
Vice President for Academic Affairs
Member

ENGR. JON ALVIN MACARIOLA
Faculty, College of Engineering
Member

Accepted and approved in partial fulfillment of the requirements for the Degree, **Master of Science in Information Technology**.

July 30, 2020
Date of Oral Defense

ESTEBAN A. MALINDOG JR., Ph. D.
Dean, Graduate School

ACKNOWLEDGEMENT

I am very grateful to **GOD ALMIGTHY** for without His graces and blessings, this study would not have been possible.

Immeasurable appreciation and deepest gratitude for the help and support extended to the following persons who in one way to another have contributed in making this study possible.

Prof. SWEET MERCY F. PACOLOR, my thesis adviser, for her support, advices, guidance, valuable comments, suggestion in the success of this study. For sharing her knowledge and help me in the technicalities of the system. And lastly by giving an endless help to finish this manuscript. I am very thankful for having such a good adviser like you.

The member of the committee **Dr. Esteban Malindog, Jr.**, **Dr. Felisa Gomba**, **Dr. Ronald Orale** and **Prof. John Arvin Macariola** for oral examination who manifested their distinguished skills and expertise in their own field as seen in their way of corrections and ideas shared.

I would also like to extend my gratitude to **Ms. Evangeline Miranda,** Samar College Elementary principal, for her patience and support for answering all my queries without given me any hassle in collecting the data that I needed during the data gathering. For his expertise of evaluating the system, for she is the primary user of the system.

I would also want to express my heartfelt gratitude to **Samar Colleges Community** for a good camaraderie, specially to the Chairman of Board of Trustees and School President **Mr. Rhett Caesar L. Piczon** for the brimful support that Samar College gave to his Employees, through his Faculty Development Fund program for the continuity of the professional growth of their employees.

I would also like to extend my sincerest gratitude to my colleagues: Ms. Venus, Ms. Abby for their encouragement and advices that we can make it through. For their words of support "kaya mo eto", Thank You! Similarly, to the Staff and Secretary of Graduate Studies, for a very accommodating staff and answering all the queries.

Lastly, I would like to thank my Family for their never ending support. The love, patience, guidance that you have given me through this time you are all my strength and a big part in the success of this study.

KUDOS TO ALL OF YOU!

-JOH

DEDICATION

This study is wholeheartedly dedicated to my beloved parents, Namy and Doroteo;
To my husband, Robert
and to our children, Sean ,Yuna, Sophia ,Yuan and Liam
who have been my inspiration, who have given me
strength when I thought of giving up, and
who continually provide their
moral, spiritual, emotional and financial support.
To my brothers and sisters, Jovanni, Bernadeth, Joseph, Jennifer, & Mary Joy;
Sister in-law and mother in-law, Eulalia;
Friends, and colleagues who shared their
Words of advice
and encouragement to
finish this study.
And lastly, I dedicate this book to
The Almighty God,
Thank you for Your guidance, strength
Protection.skills,
And good healt.h
All of these I
Offer to
You.

ABSTRACT

Successful classroom management is subjected to so much pressure that, more often than not, results to inefficient and overlapping schedule of outputs and thus need re-scheduling. This is the aimed of the study, the development and evaluation of Classroom Management using Expert System(CMES), a software that would assist Samar College's Faculty of Elementary and Junior High School Departments on the said matter. Developmental-evaluative research design was used to develop a Classroom Management using Expert System (CMES) that generates schedule of teacher's program and classes uploads schedule to enrolment system projects total student population downloads list of enrolled students per section, and imports class schedules to teachers' portable class record. It also attained the assessment of system quality attributes as to functionality, reliability, usability, efficiency, maintainability, and portability as to International Standards Organization (ISO)/International 13 Electrotechnical Commission (IEC) standards. CMES easily managed and the record data which would easily tracked, processed, and stored, because the system has the ability to project. It is written in C# programming language as its front end and SQL light as its back end. It runs on a typical type of windows computer. It is a small and simple expert system, but has a powerful application written in an AI language that can do decision making and support for the administrator. The CMES conformed with the standards of ISO/IEC 2510:2011 as per rated by the users.

TABLE OF CONTENTS

Chapter 1

THE PROBLEM AND ITS SETTING

Introduction

Classroom management is a very crucial factor in an educational institution. Several tasks comprise classroom management, such as: plotting the resources, capacity of every classroom, arranging of class schedules, and the ratio of the number of teachers to the students, the number of population to be accepted every academic year (Ed Glossary, 2014). The most complex of these tasks is academic scheduling which generally refers to institutional activities that support delivery of subjects and academic programs. It is a complex process that lies at the center of operations in a school head's office (Usman, 2016). However, successful classroom management is subjected to so much pressure that, more often than not, results to inefficient and overlapping schedule of outputs and thus need re-scheduling (Johnston, 2021).

Samar College Elementary School is an educational institution which offers basic education services to its clientele, more particularly to the community of the Samar Island. The school regularly operates starting from pre-enrolment period to the end of the academic year. Some of the activities that teachers and department heads do before the start of classes include: arrangement of class schedules and teachers' load, settlement of classroom use for the students and

faculty, in the Basic Education Department. These activities require physical strength and time.

However, the school uses the manual way in the preparation and submission of class schedules to the school's enrollment system and submission of computed grades. For example, the school's current scheduling method - does not really help in scheduling classes efficiently because of undetermined number of students, availability of resources and, rapid shifted of manpower. This method still needs an expert user to configure the task and do it in a timely manner.

The use of this study's Classroom Management Using Expert System (CMES) in Samar College will help keep track of the available resources, project the total number of population to be accommodated upon enrollment, auto-generate class schedules, upload class schedules to the enrollment system, and automated grading system.

The CMES is important in distributing the resources needed thoroughly and correctly. Nowadays, there are several solutions or applications that help in generating a classroom management system with the help of artificial intelligence. Some of these methods that are available are expert system, genetic algorithm, and fuzzy logic. Each method has its own way of performing a solution.

Classroom Management using Expert System(CMES) is a software that aims to assist Samar College's Faculty of Elementary and Junior High School

Departments in scheduling a teacher's program, generating class schedules, uploading generated schedule to Samar College enrollment system, projecting the total student population, downloading lists of enrolled students per section, and importing class schedules to teachers' portable class record.

This system conforms to the school policies and national standards of classroom management in the elementary and secondary levels. This system also used a user-friendly graphical interface. The use of CMES in Elementary and Secondary Department would provide convenience to the school heads as the system will save them from tedious manual job and laborious task. This system would also augment the school's productivity because of the reduced time they would need to allot and consume in the class management of students and teachers, giving them more time to work on other important matters.

Therefore, the researcher of this study stands very optimistic that the final output of this Class Management using Expert System would be very useful to Samar College's elementary and secondary levels' school heads.

Statement of the Problem

This study aimed to develop and evaluate the Classroom Management using Expert System(CMES), a software that would assist Samar College's Faculty of Elementary and Junior High School Departments.

It specifically attained the following:

1. Develop a system with the following features:

1.1 schedules of a teacher's program;

1.2 schedules of classes;

1.3 uploads schedule to Samar College enrollment system

1.4 projects total student population;

1.5 downloads list of enrolled students per section, and

1.6 imports class schedules to teachers' portable class record;

2. Evaluate the abovementioned features of the developed system along these system quality attributes:

2.1 functionality;

2.2 reliability;

2.3 usability;

2.4 efficiency;

2.5 maintainability, and

2.6 portability

Conceptual Framework

This section presents the conceptual consideration in developing the system. Figure 1 below represents the three dimensions of the conceptual paradigm: input, process, and output.

The first stage includes gathering of primary inputs from Samar College's Elementary and Junior High School departments of in order to develop the application. System requirements, which basically include the infrastructures needed for the development, were considered. Other external resources were also considered.

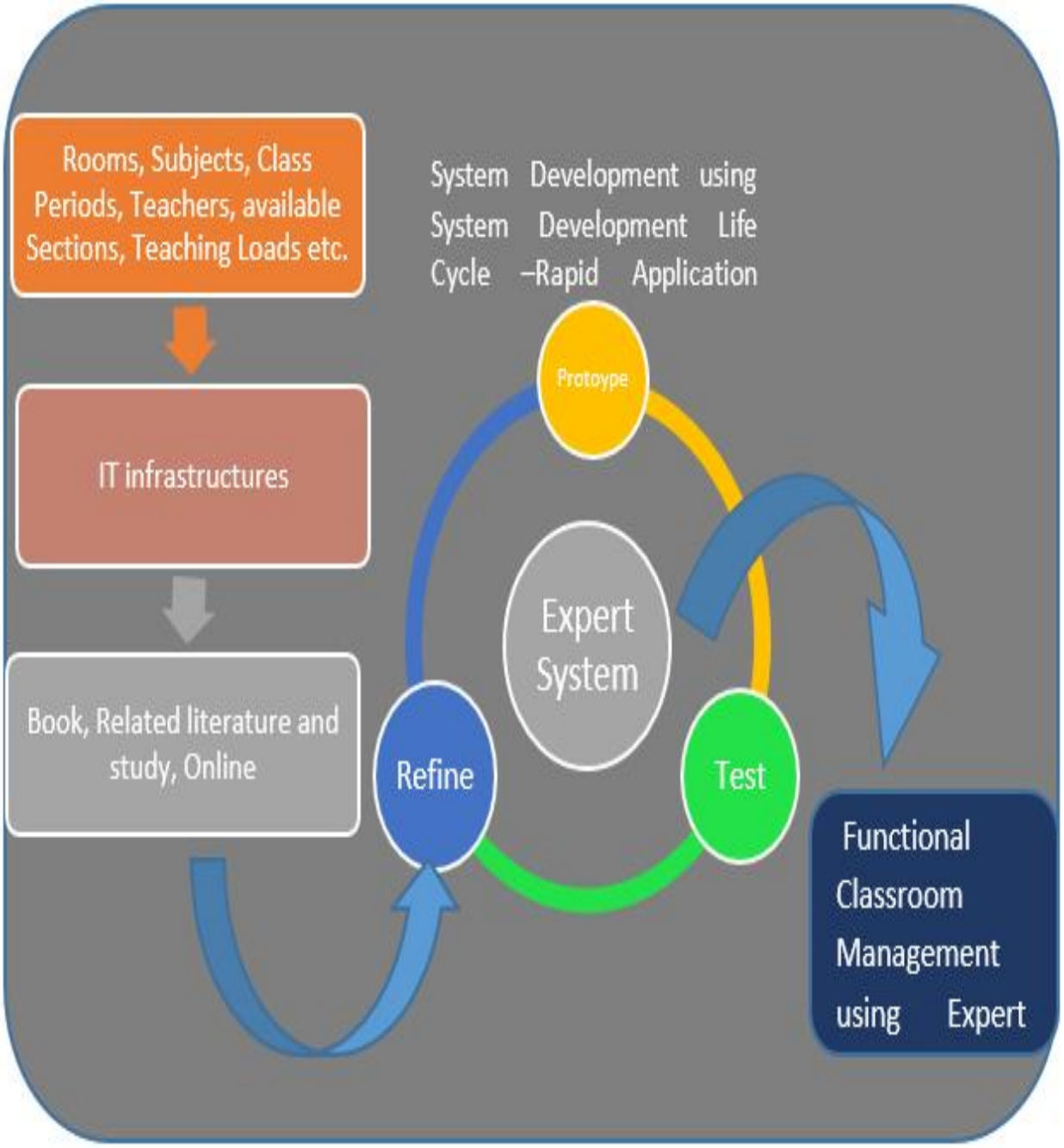


Figure 1. *The Conceptual Framework of the Study*

The second stage is process phase. The System Development Life Cycle (SDLC) Rapid Application Development (RAD) model, which is a conceptual model used in project management, was adopted in this study. Specifically, the system used the rule-based type of an expert system to determine the success of the algorithm implementation through coding and logic programming.

The last stage, the output phase, is the development of the system, the Classroom Management using Expert System (CMES) for Elementary and Junior High School Departments.

Significance of the Study

The results of the study provide advantages to the Samar College administration as well as the students, faculty, school heads, and coordinators, likewise, future researchers are believed to benefit from this study. Specifically, this study is significant to the following stakeholders:

School Administration. With this project, the administrators of Samar College can provide improved quality service to its stakeholders which would also be beneficial for accreditation purposes.

School Heads. The proposed computerized scheduling system could provide the school heads and the coordinators with reliable faculty loading and class scheduling system

Teachers. Conflicts on class schedules of teachers could be minimized. In one-way or another they could be used this in every year to lessen papers that were used when it comes to schedules.

Students. Changes in schedules which cause the delay of regular classes could be minimized with the proposed computerized scheduling system, they could easily access class schedules even without going or visiting the school, very timely for during the conduct of the study amidst pandemic.

Future Researchers. This study is beneficial to future researchers who plan to delve on studies similar with this research. This could serve as their reference and could provide them with necessary inputs in case they opt to work on modifications or updates on this study.

Scope and Delimitation

This software is mainly developed for Samar College to manage and establish a well-organized class management. The departments that are involved are Elementary and Junior High School Departments. This project was developed in a Universal Windows Platform (UWP) and used data, resources, and subjects for the academic year 2019-2020. It used the C# programming as the code generation and SQL light as the database management.

Definition of Terms

In order to provide clearer understanding for the readers of this study, the following terms are conceptually and operationally defined:

Algorithm. Conceptually, this term is defined as the process or set of rules to follow in calculations or other problem-solving operations, especially by a computer (Oxford Dictionary). Operationally, this term is defined as the rules used in the system development to come up with the proposed project.

Auto generated. This term is defined as a computer algorithm to automatically create query (Oxford Dictionaries). Operationally, this term is defined as the automatic generating of class schedule by the system.

Class Management. Conceptually, this term is defined as the wide variety of skills and techniques that teachers use to keep students organized (www.edglorassy.org). Operationally, this term is defined as the project title by automation of the existing system.

C#. Conceptually, this term is defined as a programing language developed by Microsoft (www.techterms.com). Operationally, this term refers to the programing language in developing this project.

End-user. This is conceptually defined as the person who actually uses a particular product. Operationally, this term pertains to the principal and the coordinators who would use the system.

Expert system. Conceptually, this term is defined as a piece of software that is programmed using artificial intelligence techniques (Oxford Dictionaries). Operationally, this term is defined as the governing body of the system to produce AI output.

Forward chaining. Conceptually, this term refers to a scenario where the AI has been provided with a specific problem and must "work forwards" to figure out how to solve the set problem (https://www.webopedia.com/). Operationally, this term is defined as initial data that the user input which must be forwarded to the system to solve a problem.

Inference engine. Conceptually, this term refers to a component of a system that applies logical rules to the knowledge base to deduce new information (https://www.webopedia.com/). Operationally, this term is defined as the logical rules in a system.

Optimization. Conceptually, this term refers to the action of making the best or the most effective use of a situation or resources (Oxford Dictionary). Operationally, this term is defined as the effective utilization of resources.

Rule-Based Model. The term is associated with computer programming algorithm to solve a problem (https://www.webopedia.com/). Operationally, this term is defined as the series of code sets to the system to solve the problem automatically.

REST

Scheduling. This is the process of arranging, controlling and optimizing work and workloads in a production process or manufacturing process (WikiPedia.com). Operationally, this term pertains to the process of inputting the data of class schedules.

Software Development Life Cycle. Conceptually and operationally, this term is defined as the process of developing a software (www.stackify.com).

SQL light. Conceptually, this term is defined as an in-process library that implements a self-contained, zero-configured, server less, and transactional SQL database engine (https://www.techopedia.com/). Operationally, this term refers to the back end of the system.

Timetable. Conceptually, this term is defined as schedule (something) to take place at a particular time (Miriam Dictionary). Operationally, this is a table for coordinating students, teachers, rooms, and other resources

Universal Windows Platform. Conceptually and conceptually, this term is defined as the API developed by Microsoft to help develop universal apps that run on windows 10.

Utilization. This is the action of making practical and effective use of something (Oxford Dictionaries). In this study, this term refers to the period of time when their quality is realized in practical use to achieve intended objectives and to support accomplishment of storage and maintenance.

Chapter 2

REVIEW OF RELATED LITERATURE AND STUDIES

This section presents and discussed studies of foreign and local authors published in books, journals, magazines, and newspapers. This review also includes excerpts from unpublished materials such as master's theses and dissertation papers that are relevant to the present study.

Related Literature

Reliable data from school-based surveys can provide the quality Information and Communication Technology (ICT) use data required to better inform education policy and practices, especially in developing countries. Capturing the complex set of factors involved would paint a more accurate picture of what is available and used by both students and teachers. This includes information, such as availability of digital infrastructure; internet connection speed; school activities in which teachers use ICT; training received by teachers to empower them to integrate ICT into their practices; strategies implemented by schools to develop digital skills (Montaya & Barbosa, 2020).

School day is the time that elementary and secondary levels establish for teaching and learning activities. The assignment of class schedules must meet the standards of current legal requirements, the institutional educational project and also the curriculum of the school. In general, the planning of the school day is based on the availability of teachers and classrooms and takes into account some

special conditions of each school and the overlapping of class assignments. However, the assigning of class schedules must also consider to cycles of greater or lesser degree of accuracy when performing tasks, which vary in frequency. So, students do not always have the cognitive capacity to assimilate certain knowledge or to carry out certain learning activity at any given time of the day (Suarez-Chilma, Castrillón-Gomez, & Guerrero-Aguirre, 2013).

In a high proportion of institutions, the class schedule is manually made, which additionally of taking either days or weeks of work, does not allow to set certain types of restrictions that optimize the use of both resources human, and infrastructure, as well as give errors for subjects overlaps and the teacher assignments (Khalil, 2018), thus a successful classroom management is not attained.

Classroom management is the process of organizing and conducting the business of classroom and the school as well as a whole. Many perceive it as the preservation of order through teacher control. Classroom management is much more than that, however, it also involves the establishment and maintenance of the classroom environment so that educational goals can be accomplished (Stronge, Tucker, & Hindman, 2021).

Developing classroom management is a complex process which can be a positive challenge for an experienced teacher, but it is a very daunting experience for those who are novice in organizing the classroom settings (Scrivener, 2012). Whether you starting a new managing an existing one is to one student device

program, understanding the purpose class management software has on instruction and technology is essential to leveraging the benefits student devices bring to the classroom (Green, 2020).

With reviews of literature, it is deemed necessary to create classroom management system to address the existing problems on schedules of teacher's program, classes, and the like both in the elementary and secondary levels.

Related Studies

The following readings of conducted studies are presented from the previously developed system up to the most recent studies which has similarities on the developed Classroom Management System of the proponent.

The study of Lee and Wu (1995) titled, CLXPERT: A Rule-Based Scheduling System" focuses on a conflict resolved by using breadth-first search in conflict tree. They collected the necessary information like teachers' names, room numbers, and subjects that are necessary as input in producing results of the system. Similarly, this current study required data so that the system can auto-generate the output consuming less time. Their research development and this study are similar because both of their developed systems use rule-based expert system, wherein the syntax of algorithm is based on an if-then statement. If a system implements rules in the form of if-conditions, that is, if the condition is satisfied, then the statement would be executed. With this approach, it would be

easier to manage not only the scheduling task, but also the entire school management.

In the study of Majiwan (2005), he stated that in generating timetables for faculty, a prototype would implement a rule-based approach to overcome time tabling constraints. This study' CMES has a similarity with a few differences. Majiwan (2005) system focuses only on timetabling or the faculty members. On the other hand, the present study not only focuses on time tabling, but also on class management as well. The previous still needs human users in generating a result; but the present study has auto capabilities a used just need to input the required data and the system would auto-generate the timetable.

A study by Ho and Lu (2005) titled Web-based Expert System for Class Schedule Planning using JESS have developed an expert's system for class scheduling in a web-based platform. This system collects data from students in order to have a guide about their desired schedule. It uses Java expert system shell to process the input and can generate a feasible schedule for student. The only similarity of this study to the present study is that it uses a knowledge-based data so the system can generate an output. Although both studies have a lot of differences such as the front end of their study, a php was used because it is a web interface-based form. In contrast, the current study uses visual studio framework for it is a stand-alone system. It does also require students as users of the system. Nevertheless, both studies used the same approach of using an expert system in addressing the problems encountered during the process.

A local study titled Class Scheduling Management System for University of Rizal conducted by Abcede, Gonzaga, Ontoca, Zapanta, & Abante (2013) focuses on how to develop and deploy a Class Scheduling Management System that would enhance the traditional method of scheduling system used in the University of Rizal Antipolo Campus. They also stressed that the main objectives of his study are: 1) to develop an effective and efficient scheduling of the university classrooms; 2) to help minimize the problems encountered by the administration, department heads, and instructors during the start of classes; and 3) to maximize availability of classes for the students of the University. Similar with aforesaid study, this present study aimed to help attain the maximum capacity of available resources and to ease the burden of the school administrator amid complex tasks of managing its resources. Though the previous system did not use the expert system approach unlike this current study, it still managed the same output. The advantage of the present study is that through the use of AI-based expert's system, it can give auto-generated results and provide results of the projections scheme. Additionally, it has more efficient features that does not limit alone in scheduling, like the monitoring of classes by tapping the CCTV of each classroom and by monitoring the humidity of room temperature as well.

Another study titled Rule-based Expert Systems for Supporting University Students of Engin, Aksoyer, Avdagic, Bozanli, Hanay, Maden, & Ertek (2014), showed a great similarity with the current study. In his study he pointed out that using a simple expert system with a powerful application written in AI

languages would cater to all kinds of administrative works. That would help ease the burden of administrative staff in their tasks. Their previously developed system includes two expert systems. First expert system is on course advising incoming students of a university using expert's system and the knowledge-based data. As to process, the user inputs the data and the system would come up with a result of advice identifying a particular course which students should take up. The second one is about suggesting scholarship to the students based on their admissibility data that they input in the system. This is not similar with the current study. Nevertheless, both studies use expert system approach in advising projecting an output. Likely, in the present study, the system is capable of projecting the student population to be accommodated during enrolment process as shown in the knowledge-based data that was stored in the system database.

A paper of Guyette, Hamidian, & Tuazon (2014) titled, A Rule-based Expert System Approach to Class scheduling also showed a similar principle with CMES. The authors of the study states that class scheduling is an unenviable task in which a team of scheduling experts must assign faculty members to their requested courses based on the university and department guidelines. The authors claim that the complexity of this task is compounded by introducing the various necessary faculty and course parameters. They also added that although there have been recent attempts to create the class scheduling process via conventional programming, this form of software is poorly suited to emulate the essential human decision-making process. However, developing a rule-based

expert system approach can provide a practical solution to the problems encountered. The authors are very optimistic that with this kind of study the challenges, hence making the productivity rate of a human expert increase.

Similarly, Botangen and Khan (2014) developed a program on Scheduling System for the Central Luzon State University (Abridged), the designed and developed class-scheduling system allows collaborative preparation of schedules among several users. The system integrated five components: the data management module, course assignment module, scheduling module, result storage module, and the report module. It has an engine that uses the greedy algorithm for creating schedules and detecting conflicts. The algorithm mainly executes this sequence of processes; selecting available time, finding available room, and looking for an appropriate faculty while considering different constraints and preferences set by users. The score is similar to the current study but other attributes were different.

Another paper titled A Knowledge-Based Expert System for Scheduling in Services Systems by Lopez-Santana and Méndez-Giraldo (2016) proposed a system mainly for services scheduling particularly in manufacturing services which focus on health care. Similar with the present study, it uses a knowledge-based expert system to emulate the process. They also stressed out the different processes in manufacturing and services scheduling.

The reviews of studies contributed in the attainment of the current investigation. It would serve as bases to which components of the presently

developed system have similarities to at least able to improve or enhance their classroom management.

Chapter 3

METHODOLOGY

This chapter discusses the methods and procedure that were taken in developing the system. It specifically discusses the methods used in developing the proposed project.

Research Design

This study used developmental-evaluative research method. This was used since the research aimed to develop a Classroom Management using Expert System (CMES) which generates schedule a teacher's program and schedules of classes; uploads schedule to enrolment system; projects total student population; downloads list of enrolled students per section, and imports class schedules to teachers' portable class record.

The evaluative research designed was used at the same time in the assessment of the developed system quality attributes as to functionality, reliability, usability, efficiency, maintainability, and portability. Since the developed system was beta-tested to two groups, the elementary and junior high school users, their evaluations were compared to validate the system quality attributes requirements. Thus, statistical tools such as mean scores and t-test for independent samples were employed in the data analyses.

The situation-specific research design, as used on this study, is implied in the design process diagram shown in Figure 2

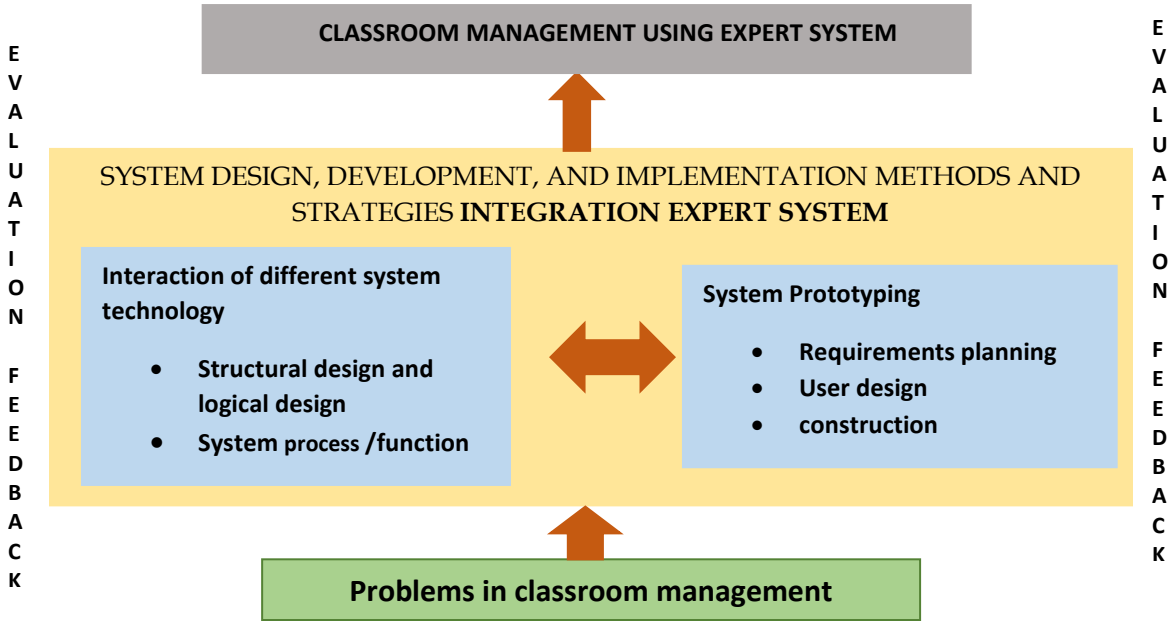


Figure 2. *Research Design Process*

The research design process led to the realization of this study. The data and information on the current problems and issues in class management with limitations on the use of existing technology. The said information was combined and became the basis in the formulation of methods and strategies of the study. System design and implementation strategy was based on the evaluation of the existing technologies that are similar with this study. System evaluation was based on prototyping which aims to examine the current model and to look for a particular area of improvements.

The current problems and issues in conflict of schedule, time management, resources management and teachers' schedule are very common in the current system. Considering the feedback's data on class management will result to the finding possible solutions. Ideas and projected solutions would then be used as bases in designing a more holistic approach in addressing the problems which is the goal of this study.

Expert System

Expert system is one of the well-known research fields in Artificial Intelligence. Expert system is a computer application developed to solve complex problems in a particular domain at the level of an extraordinary human intelligence and expertise. Basically it is composed of knowledge, Interface Engine and User Interface.

Knowledge is the data being collected through the end user of the system. This is the raw data of class schedule management in setting class schedule, rooms, subjects, class period, teachers, and other. Prior to the operation of the system, it will ask for the initial data like *list of rooms* and their attributes (i.e. the room name, room location) This is done in order to prioritize the primary level in occupying the ground floor, while the second floor is for the intermediate level. Then, the list of teachers for priority listing should be inputted. Teachers' priority listing is based on the following: 1) post graduate studies; 2) Licensure Examination for Teachers (LET) passer; and 3) number of years in teaching. The

last one is the list of subject and class time duration for every grade level. Then categorization and organization of these information follow.

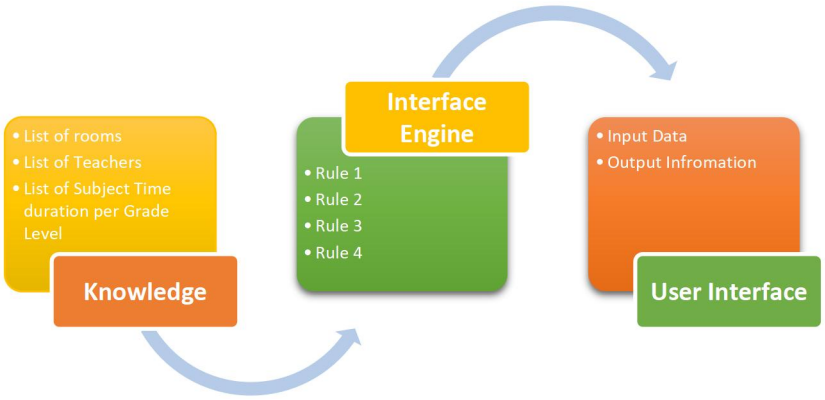


Figure 3. *Expert System*

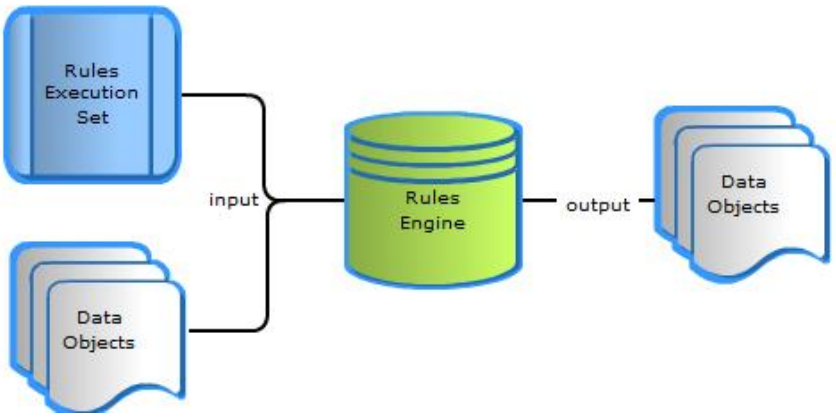


Figure 4. *Rule-based Expert System*

Interface Engine

Rule-based is the simplest form of AI that uses a data to make a rule to solve a certain problem. This works when a data is taken from an end user and translate the data into a series of hardcoded rules. The rules are normally conditional statements in an IF-THEN algorithm. Rule-based will apply a series

of rules to an available data and add new knowledge. This will resolve rules conflict when multiple rules are applicable to a particular case. To illustrate the solutions rule-based, the researcher used a simple rule in the knowledge base for the Class Management system as shown in the Figure on the next page.

Rule 1: class period
Rule 2: subjects
Rule 3: Teachers availability
Rule 4: Generate Class Program Scheduling.

sample code

```
If
   {Number of section per grade level
                And Class size per grade level
Then
      Total number of population per grade level}
```

In the rule-based model, a series of rules is implemented by the expert system to enable to generate a solution. First, by defining the number of student population per grade level and class size, the system will generate the total number of sections per grade level. Then, an available room per section in the grade level, the subjects, and assigned teachers are selected next. After that, the rules will go back to the number of section_count. Then, repeat the cycle until all the sections have all been selected. The same process is used until it reaches to the last grade level.

Rapid Application Development

The researcher will use the Rapid Application Development (RAD) system development life cycle which acts as a foundation and will make the system work. The RAD has four phases of approach to problem solving which may be iterative:

Requirements Planning, User Design, Rapid Construction, and Cutover.

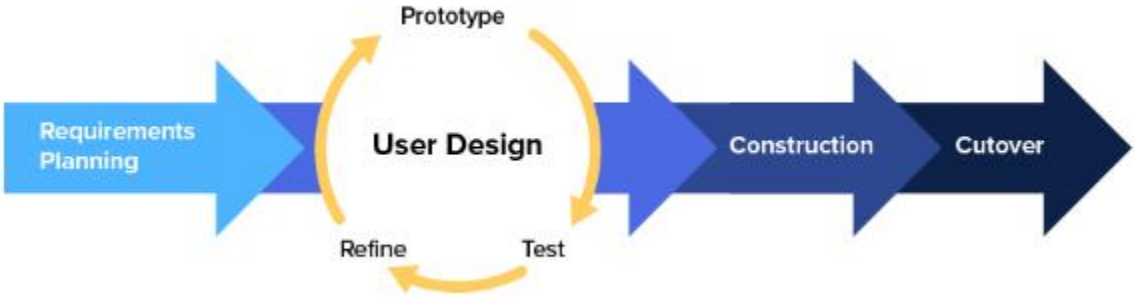


Figure 5. *The Rapid Application Development (RAD)*

The planning phase, which is a detailed study of the system, was taken out from the statements of the problem because this would involve an in-depth study of the whole operation of the system. In this process, the data available were collected and interviews and on-site observations were used for a detailed study of the system. All the data and the findings were documented in the form of detailed data flow diagrams (DFDs), data dictionary, logical data structures, and miniature specifications. It also included planning for the new system, analysis of requirement, system constraints, functions and proposed system architecture, prototype of the proposed system, and its analysis.

Phase two is the user design phase where actual data were collected and the process was analyzed. This phase also involves identifying problems and giving suggestions to improve the functionality of the system. This included studying the entire system processes, gathering operational data, understanding the information flow, finding out bottlenecks, and evolving solutions for overcoming the weaknesses of the system so as to achieve the organizational goals. It also included subdividing of complex process involving the entire system, identification of data store, and manual processes. It was more of a thinking process which involves creative skills. It attempted to produce a new efficient system that satisfied the current needs of the user and provide an opportunity for future growth despite the organizational constraints.

It also was based on the user requirements and the detailed analysis of a new system. The logical system design of system analysis result was converted into physical system design such as for the inputs, outputs, databases, forms, codification schemes, and processing specifications. These were drawn up in detail. In the design stage, the programming language used C#- Universal Windows Platform (UWP) as the front end of the system and SQL light as the back end.

This also covered development or coding of the system to have prototype in which the program specifications were converted to computer instructions or programs. The written program manages the entire process and controls the flow of the system. Different modules of the program were used for the fast

development of the system. A series of testing and refinement were conducted during this phase. Checking for bugs and testing the system were also conducted.

The actual coding and testing was administered during the construction phase. The expert system will come in this phase using the Rule- based modeling of algorithm. Rule-based system provides automatic problem solving tools for capturing the human expertise and decision making. Rule-based system was used to store and manipulate knowledge given a fixed reasoning method. The process of searching through alternative solutions could be affected by the structuring and ordering of the rules in implementations. A rule of the form "if a and b and c then d" may be interpreted in backward chaining as in a procedure of four steps: to do d, first do a, then do b, then do c. Although the procedural interpretation of rules reduces the advantages of declarative representation, it can be used to reflect more efficient heuristic solutions and strategies.

Block Diagram

The system is composed of two security levels. In the first security level, the system requires the end user to input their username and password for the system's data integrity. The username and password can be set through the first installation of the system to the end user's device or computer.

Next security level in the system process focuses on the core of the system where there are different modules iteratively, namely: knowledge module,

Interface Engine, and the User Interface module. Knowledge module is the initial data inputted by the end user to the system and store it in the database. This is the basis of the expert system to calculate possible solution to a given problem. This includes the data on the list of teachers and the given attributes for them in the priority list part such as, postgraduate Studies, Licensure Examination for Teachers (LET) passer, and number of years of teaching experience. Initial data is the list of rooms and their attributes such as, room name and location and priority for the primary level in occupying the first floor of the building while the second floor is occupied by the intermediate level respectively.

Interface Engine module with a series of rules is implemented to give solution to all the queries made by the end user. This involves the logical algorithm that will occur in this module. A series of calculations, predictions, and prioritizations are made in this module. Calculations start with the possible total number of sections and class size in order to project the total number of population per grade level to be accommodated upon enrollment. Then the system will auto-generate the schedule starting with the of the class schedule, checking the total number of rooms and their availability, assigning of teachers through priority listing, matching the availability of teachers' time schedule, and assigning subjects and time duration per section. All of these tasks are handled by the expert system.

In the user interface module, the interaction between the user and the system is shown. End user will only input data like the desired total number of

sections per grade level and the class size. The system automatically generates a class program and manages the entire class by checking the status of every classroom by tapping the CCTV camera in each classroom. Once the schedule has been generated, the end user has the option to upload the schedule directly to Samar College Enrollment System through Representational State Transfer (REST) Application Programming Interface (API). Once the schedule has been uploaded, Samar College Enrollment system can now facilitate the enrollment based on the uploaded schedule. System is also capable of importing class schedule to its portable grading system through USB flash disk for faster inputting of grades. Figure 6 below shows the block diagram of the study were it show the whole functionality of the system. Through the input to the process and the output of the system.

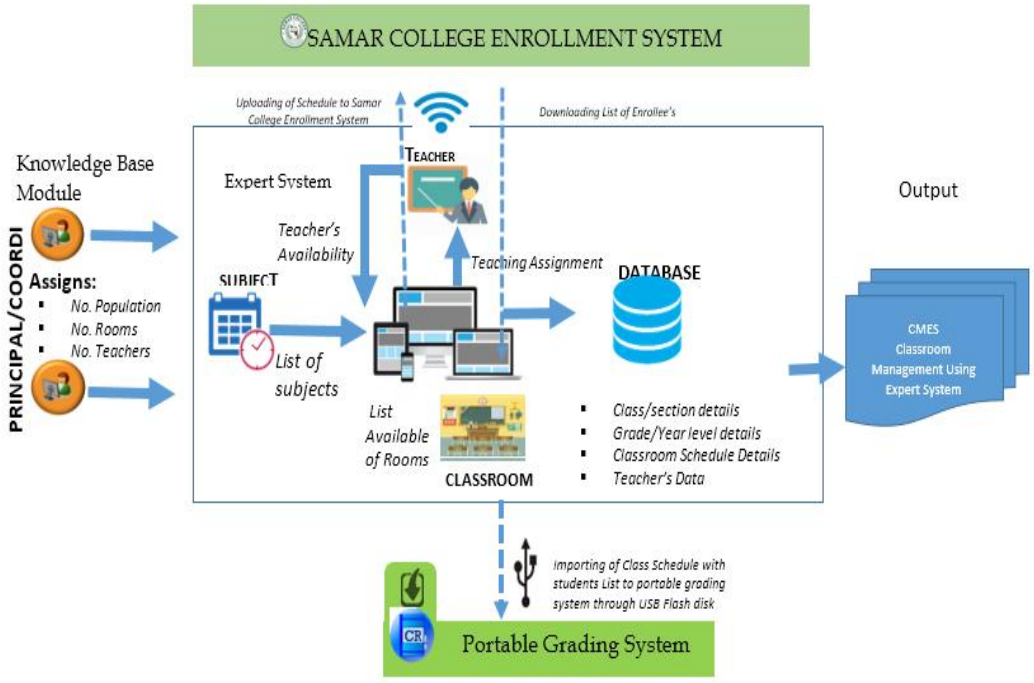
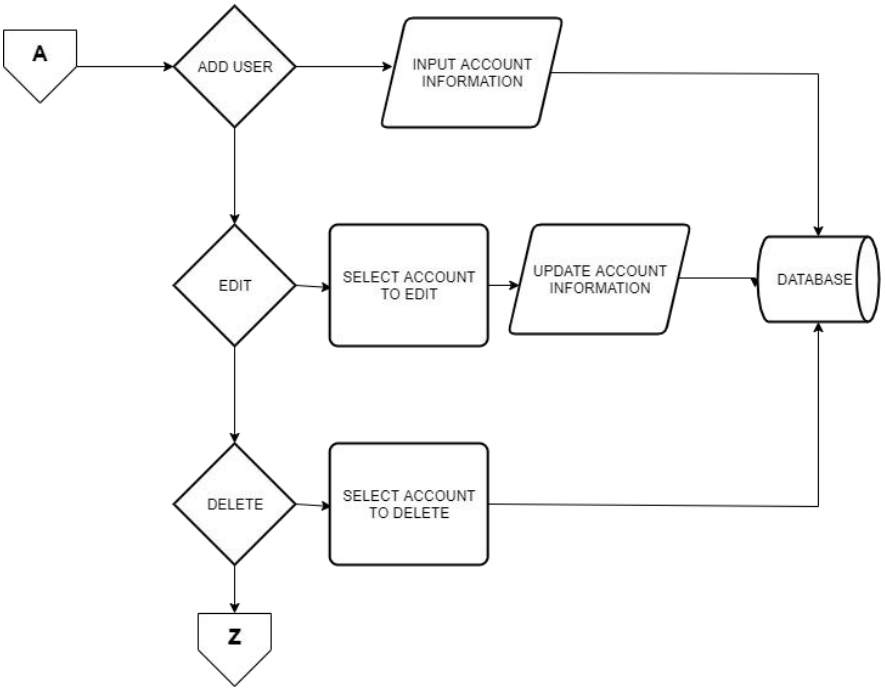
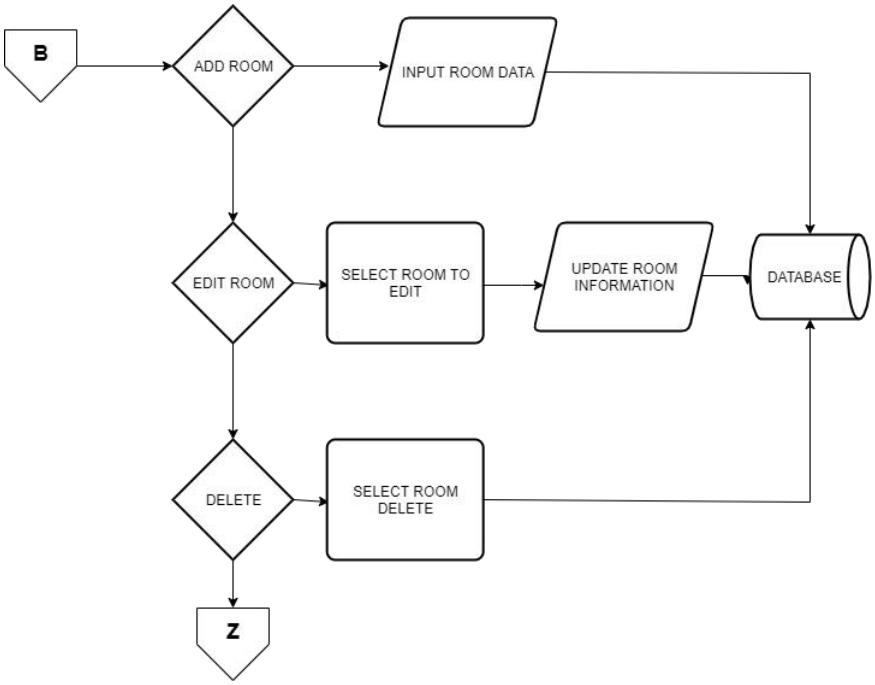
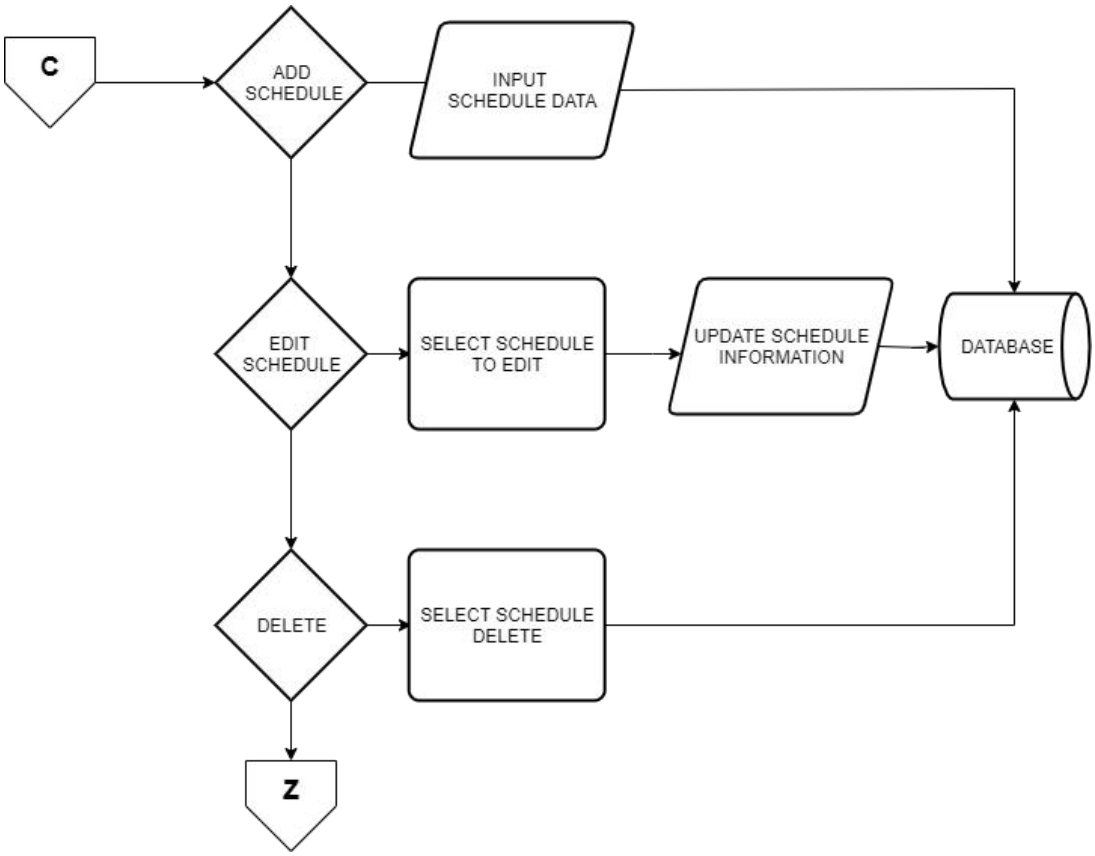
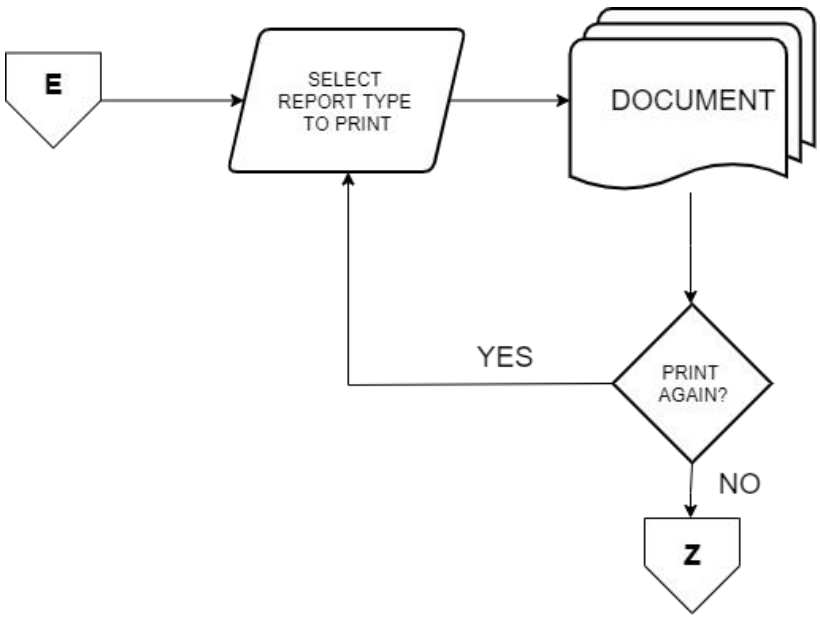


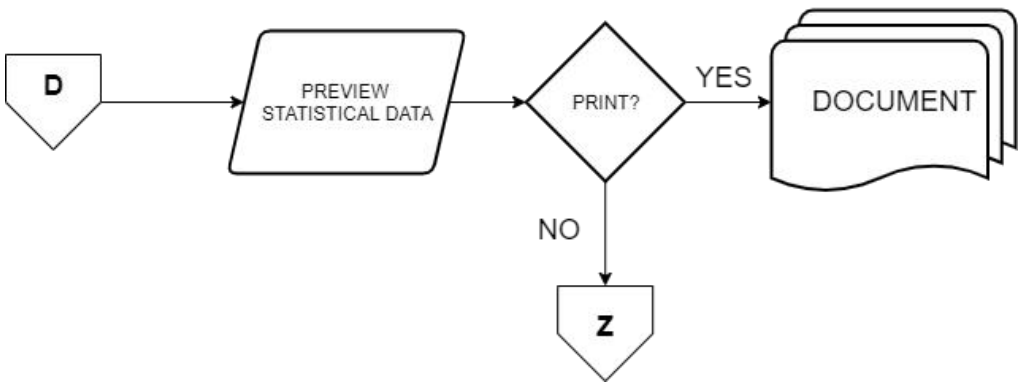
Figure 6. *The Block Diagram*


A
ADD USER
INPUT ACCOUNT INFORMATION
EDIT
SELECT ACCOUNT TO EDIT
UPDATE ACCOUNT INFORMATION
DATABASE
DELETE
SELECT ACCOUNT TO DELETE
Z


B
ADD ROOM
INPUT ROOM DATA
EDIT ROOM
SELECT ROOM TO EDIT
UPDATE ROOM INFORMATION
DATABASE
DELETE
SELECT ROOM DELETE
Z


C
ADD SCHEDULE
INPUT SCHEDULE DATA
EDIT SCHEDULE
SELECT SCHEDULE TO EDIT
UPDATE SCHEDULE INFORMATION
DATABASE
DELETE
SELECT SCHEDULE DELETE
Z


E
SELECT REPORT TYPE TO PRINT
DOCUMENT
YES
PRINT AGAIN?
NO
Z


D
PREVIEW STATISTICAL DATA
PRINT?
YES
DOCUMENT
NO
Z

The system starts with the first level, the log-in module. The system asks the end user to input his/her username and password to be validated by the system. If the log in is not valid, then the end-user can access the four different modules, namely: manage user account, manage facilities, schedule, and statistical.

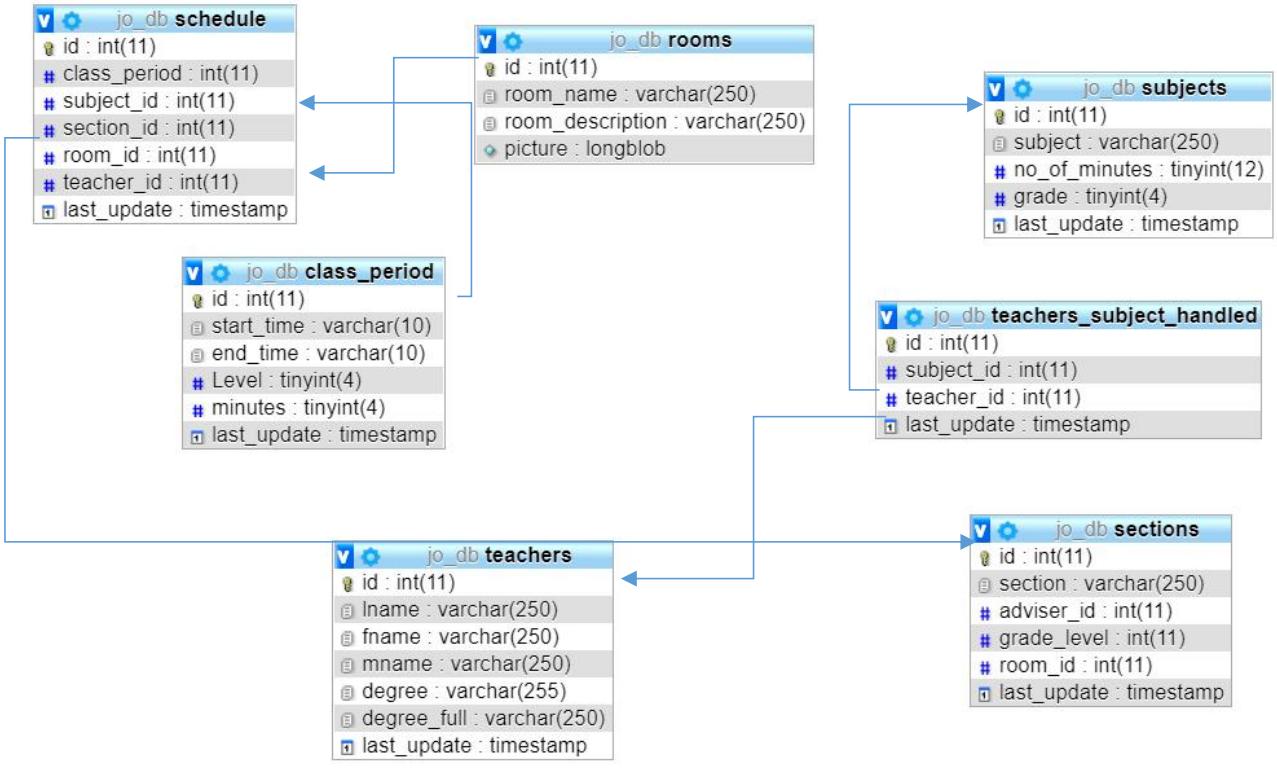
Managing schedule module displays the time allotted per subject and the entire schedule where the end-user can plot the schedule per section and per grade level. Once the grade level and section are selected, the end user can now assign a subject and a teacher for a particular subject. A menu pops up for the selection of subjects and teachers. The end user should choose a subject with the corresponding allotted time per subject.

Next is the inputting of information about and displaying the basic information of teacher-in charge and the percentage of the teaching assignment/load of a teacher. A teacher is required to render a service of 6 hours a day or 360 minutes a day.

Then, inputting of subject/s follows. The end-user inputs all the subjects according to a grade level with its corresponding allotted time. Adding the rooms is next, wherein the end user inputs a room assignment for each grade level and section. This also includes inputting basic information of the room. Then sections are added for every grade level. Last process is on the viewing and printing of the class home program.

Database Schema

Data Base Schema shows the relationship of the different tables on how the system works. It starts with the teacher table_id. The end user will input the information of the teacher which is connected to the subject_id table. This table requires the subject's name, the number of minutes allotted to a particular subject, and the grade level enrolled in a particular subject. Next table is the schedule_id table which is connected with other table like the room_id table and the class_period table. The Entity Relationship Diagram below shows the connectivity of the different tables in the database. This Figure basically shows the flow of the system. The diagram also shows the management, retrieval, storage, update on the database. Also it will show how the data will be display according to the end user preference.



Data Dictionary

sections

Column	Type	Null	Default	Links to	Comments	MIME
id *(Primary)*	int(11)	No				
section	varchar(250)	No				
adviser_id	int(11)	No				
grade_level	int(11)	No				
room_id	int(11)	No				
last_update	timestamp	No	CURRENT_TIMESTAMP			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	2	A	No	

subjects

Column	Type	Null	Default	Links to	Comments	MIME
id *(Primary)*	int(11)	No				
subject	varchar(250)	No				
no_of_minutes	tinyint(12)	No				
grade	tinyint(4)	No				
last_update	timestamp	No	CURRENT_TIMESTAMP			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	90	A	No	

class_period

Column	Type	Null	Default	Links to	Comments	MIME
id *(Primary)*	int(11)	No				
start_time	varchar(10)	No				
end_time	varchar(10)	No				
Level	tinyint(4)	No				
minutes	tinyint(4)	No				
last_update	timestamp	No	CURRENT_TIMESTAMP			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	0	A	No	

rooms

Column	Type	Null	Default	Links to	Comments	MIME
id *(Primary)*	int(11)	No				
room_name	varchar(250)	No				
room_description	varchar(250)	No				
picture	longblob	No				

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	3	A	No	

teachers

Column	Type	Null	Default	Links to	Comments	MIME
id *(Primary)*	int(11)	No				
lname	varchar(250)	No				
fname	varchar(250)	No				
mname	varchar(250)	No				
degree	varchar(255)	No				
degree_full	varchar(250)	No				
last_update	timestamp	No	CURRENT_TIMESTAMP			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	24	A	No	

teachers_subject_handled

Column	Type	Null	Default	Links to	Comments	MIME
id *(Primary)*	int(11)	No				
subject_id	int(11)	No				
teacher_id	int(11)	No				
last_update	timestamp	No	CURRENT_TIMESTAMP			

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	id	0	A	No	
subject_id	BTREE	No	No	subject_id	0	A	No	
teacher_id	BTREE	No	No	teacher_id	0	A	No	

Research Instrument

The study conducted a prototyping test used as an instrument in evaluating the acceptability of the proposed system titled Elementary and Secondary Classroom Management System using Expert System, it is derived from the website of International Standards Organization (ISO)/International Electrotechnical Commission (IEC). The quality attribute indicators include functionality, reliability, usability, efficiency, maintainability and portability. The

instrument is a Likert-Scale type which answered by the respondents through checking.

Data Gathering Procedure

The proponent provided questionnaire in the assessment of the developed system. Elementary and Junior High School Principals, Administrators and Teachers are the primary respondents of the system. The training was conducted one-on-one through face-to-face following the necessary health protocols. Prior to one-on-one survey conducted by the proponent, the respondents were asked of their available time and pursued with the system training. Likewise, the proponent conducted an onsite observation on the actual process of the previously used system. The gathered data were the bases for the information required in the development of the proposed system.

Statistical Treatment of Data

The statistical treatment of data used for the beta-testing are descriptive statistics such as mean and weighted mean were also used to reflect the overall score of the potential users with regards to the system quality attributes. The descriptions for each mean of quality attribute indicator is adopted from the Five-Likert Scale of International Standards Organization (ISO)/International Electrotechnical Commission (IEC) 2510:2011, this was similarly used for the questionnaire provided to the trained users. Where, 1.00 – 1.50 mean rating is

described as Very Poor (VP); 1.51 – 2.50 as Poor (P); 2.51 – 3.50 as Fair (F); 3.51 – 4.50 as Good (G) and 4.51 – 5.00 as Excellent (E).

Ethical Consideration

A communication letter was sent to the Office of the Senior Vice-President of the Academic Programs to conduct the study to the current approach that they used in Timetabling. The researcher also sought the permission of the Elementary and the Junior High School Principals because they also served as the primary respondents of the study. All collected data were handled in extreme confidentiality and were used only for this research study.

Chapter 4

PRESENTATION, ANALYSES, AND INTERPRETATION OF DATA

This chapter discusses and presents the final developed system. The discussion includes the functions and operations, the different system logical operations and processes, and the systems generated data and results of operations among others.

System Design CMES (Classroom Management using Expert System)

CMES is an expert system that can do class and room scheduling which are tasks that burden the department head every school year. It is written in C# programming language as its front end and SQL light as its back end. It runs on a typical type of windows computer. It is a small and simple expert system, but has a powerful application written in an AI language that can do decision making and support for the administrator.

Through this study, the class management process would be easily managed and the record data would be easily tracked, processed, and stored, because the system has the ability to project. This system is deemed to be of big help to the administrator. With the use of an expert system, the system would generate an automatic and error-free schedule and can generate projections based on the given data. The system is capable of interacting with other system

technology like the Samar College enrolment system through wireless fidelity access and portable class record using universal serial bus flash disk.

System Design Process and Operation

Modeling the administrator's decision making in class management is primarily important in designing and building the process. The system has a projection scheme generator wherein the end user can input a class size and section number projected in every grade level. The system will derive a result of the possible number of population in every grade level. Expert system comes up with results regarding the unassigned classroom and unassigned teachers based on the data that have been inputted during the configuration process.

Below is Figure 7 showing the total number of teachers and rooms with a summary result.

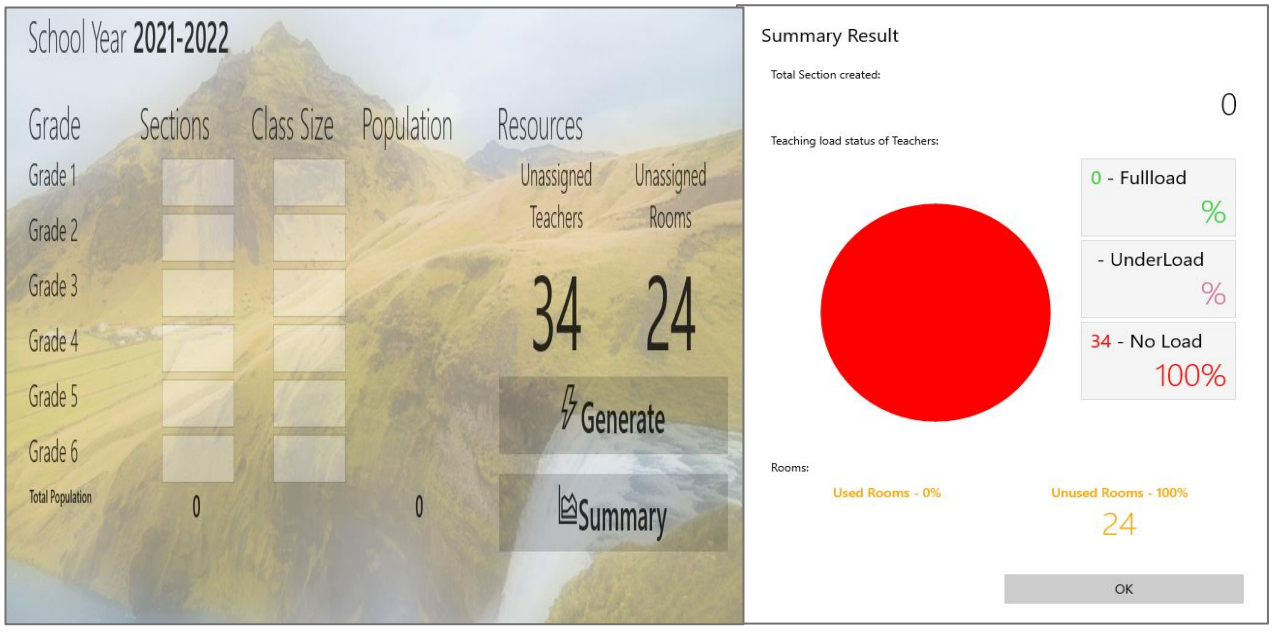


Figure 7 *Total Number of Teachers and Rooms & Summary Reports*

To start the system, the necessary data such as the list of subjects, rooms and teachers must be encoded. Then the user will input the desired number of sections per grade level and its class size. With inputted data from the user, the system will show a result with a projected population to be accommodated upon enrollment. Once it is set, the system can generate the entire schedule for a particular school year. Below is Figure 8 showing the summary of the available rooms and teaching load status of teachers.

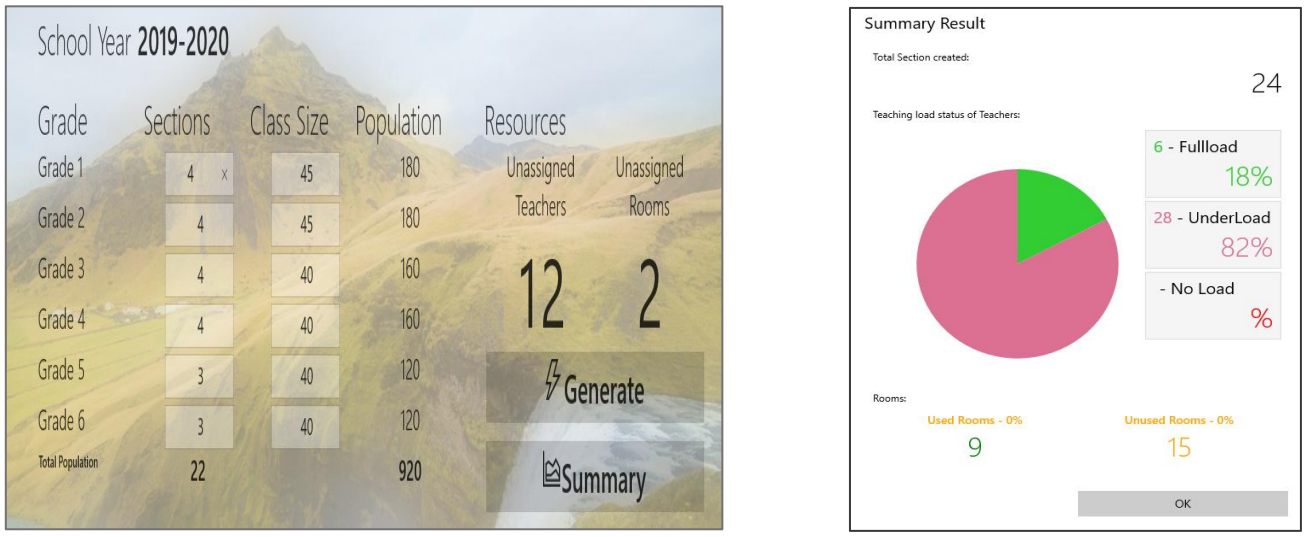


Figure 8. *Class Management Projection Scheme & Summary*

In generating the schedule, the system executes rules that are represented by a set of IF-THEN rules such as Class_Period, Subject, and Teachers_ availability. Class Period defines the time allotment in every class period for each grade level. Elementary Department is divided into two class types: the primary class which comprise of grades 1 to 3 and the intermediate class which comprise of grades 4 to 6. Each class type has different timetable pattern that is used as a

guide to set the schedule. Subject rule defines the subjects in every grade level to be scheduled. The subjects have different time durations which range from 30 minutes to 50 minutes. Once the Subject is match up with the class_Period then the third rule will be executed. Teachers_ availability rule defines the list of teachers to be scheduled on a priority basis. Four levels of priority are implemented: 1) tenureship of the teachers; 2) postgraduate degree; 3) the length of service to the institutions; and 4) probationary/non-LET passer status. The system matches the available teacher to a specific time slot. The inference engine compares each rule stored in the knowledge base with data contained in the database. When the condition is satisfied then the statement is executed based on the data stored in the database. The satisfied-execute cycle stops when no further rules can be matched. The inference chain indicates that the expert system applies the rules to reach a conclusion. The following Figure shows the inference engine cycles via a satisfied- executes Procedures of the data that are executed and stored in the database.

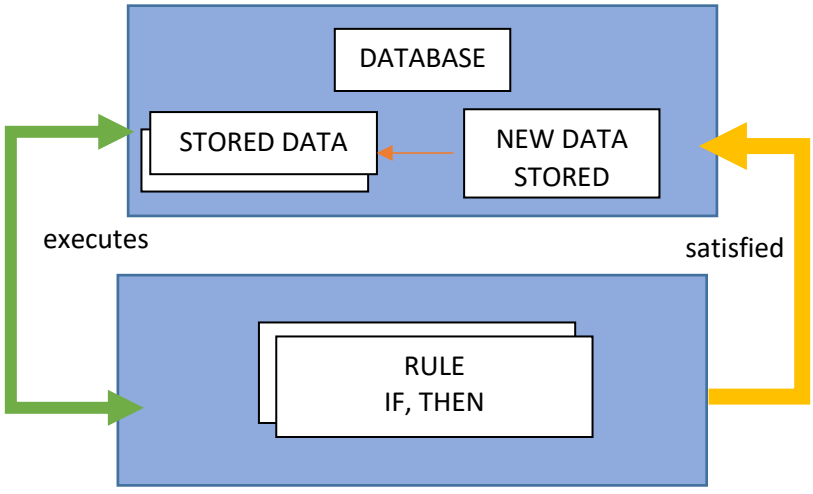


Figure 10 *The Inference Engine Cycles via a Satisfied-Executes Procedure*

User Interface

For an expert system to be operational, knowledge concerning the specific application must be provided. If the application knowledge is changed, then the input to the expert system must be modified. The application versatility which is allowed by the modification of data unfortunately creates a burden to the user. The more complex the input data, the greater the expertise the user must have. Therefore, one of the main objectives of this expert system is to create a user-friendly environment. This may include the creation of a natural language interface to simplify the input of data to the expert system. Another technique to generate a user-friendly environment is to reduce the amount of knowledge the user needs to know about operating the expert system. The classroom management project combined both of these approaches to produce a generalized expert system.

Input/Output Files

To achieve a user-friendly environment, the CMES project uses natural language in a file structured system. As shown in Figure 11, the files include class period file, faculty file, subjects file, and result file.

Figure 11. *Generated Schedule with Different Time Pattern*

Each file is responsible for data input. The information found in the input files allows the expert system to custom-tailor for a specific application. The class period file provided the expert system with knowledge concerning the available time slot. This information advises the expert system when classes maybe thought in a given week.

The "class_period" statement (e.g. `//Class_period [ days of the week] [hours]//`)at the beginning of each line indicates that time slot information is provided. Since there are only two types of class_period in every week for Elementary and a one-time block for Junior High Shcool in every week, the class_period was embedded to the expert system.

At the beginning of the classroom management process, it was assumed that the department would provide information on its teachers. This information includes their names and their priority rank. This knowledge is presented in the teachers' file using the following format:

```
//teacher_name:[name_of_faculty_member]//
//teacher_priority_rank:[overall]
```

Each teacher will have his/her own knowledge block for the expert system to draw a form. The following Figure shows the input/output file structuring were it shows from where does the input of data come from, then after the data have been input the process of the expert system were occurred, then the output.

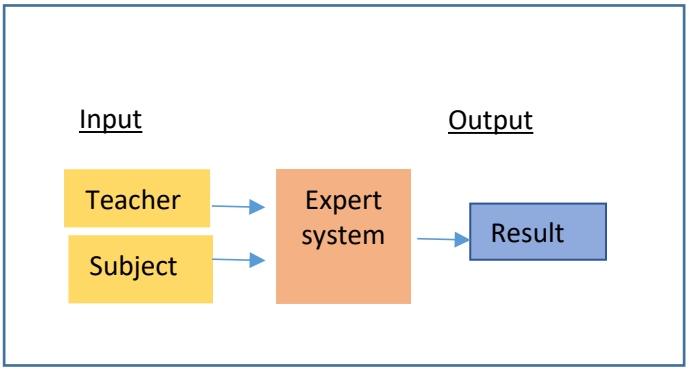


Figure 13. *Input/output File Structure*

System Scheduling Logical Algorithm

```
List<Section> CreatedSections = new Sections<Section>()
      projected_section     = 22;
      while( CreatedSections.count < projeted_section){
            Section    nSection = new Section();
            Class_period[] classperiods =  getClassPeriod();
            nSection.setClassPeriod( classperiods );
            foreach( nSection.getClassPeriod() as classperiod){
                  Subject[] subjects = getUnAssignedSubjects();
                  foreach( subjects as subject){
                        if(classperiod.duration                    ==
subject.duration){
                        classperiod.setSubject(subject)//subject
selected*
```

The logical process starts with the total number of sections inputted in the projection generating scheme. This means that the system can make an array of sections based on the projections. Once a new section is created, it will be

assigned with a set of class period. These class periods are the timetable patterns allotted to a particular grade level. It is composed of different time durations. The following codes are used for the teachers in assigned the subject teacher and filtering the qualification of teachers in the stored data.

```
Teacher[] teachers= getTeachers()
      foreach( teachers as teacher ){
                  if(teacher.IsAvailable( class_period.period ){
            if( teacher.IsLETPASSER(){
                  classperiod.setTeacher(teacher)//assign   teacher
and exit block
                              Goto jump_point;
            }else if( teacher.hasPostGraduateDegree(){
                  classperiod.setTeacher(teacher)//  assign  teacher
and exit block
                              Goto jump_point;
            }else if( teacher.legnth of service(){
                  classperiod.setTeacher(teacher)//  assign  teacher
and exit block
                              Goto jump_point;
            }else{// probationary
      classperiod.setTeacher(teacher)//  assign  teacher  and  exit
block
Goto jump_point;
}//subject
jump_point:
}// classperiods
CreatedSections.addSection( nSection );}
```

After assigning a subject for a particular period, the next step is to search for teachers who are available in that time period. The result could be an array of teachers in the list. With that, the priority rule is executed, wherein each available teacher is filtered based on the following priority attributes: 1) to check if a teacher is a regular employee/LET passer; 2) if the teacher has a postgraduate degree; 3) if the teacher has rendered a service of at least a year; and 4) if the teacher is in a probationary status/non-LET passer. The first level of priority attributes filters out all the teachers' list, and then if the result comes up with more than one teacher, the second level of priority is executed. If it still can produce more than one result, third priority is also executed, until such that the result is equal to one. But if result has more than one, the system randomly picks a teacher from the given result. If the result returns to one, the teacher is assigned on that class period. The process is repeated for the next class period until all class periods have been assigned to a subject and a teacher. Then, it will proceed by creating a new section. The cycle will be repeated until the projected total number of sections is created.

System Capabilities

The primary operations of the school are consistent starting pre-enrolment period up to the end of school year. Several activities are executed by the department heads before classes start, namely: 1) arranging class and teachers' schedules; 2) projecting the number of sections to be opened in every grade level; 3) monitoring the capacity of every classes or class size for every section; 4)

considering priority of teachers for teaching load; and 5) preparing classrooms for the students and faculty. CMES eliminates the need to involve a human expert in the class management process as it emulates human decision mechanism. CMES is developed with the knowledge and logic necessary to replace the current human expert with classroom management using expert system that can detect conflict automatically, filter teaching status of teachers, distributes classroom resources well, print teacher's program and classroom program of every grade level, an d interact with other system technology like the Samar College enrollment system and portable class record through wireless access fidelity and universal serial bus flash disk drive.

The following features and functionalities are integrated into the system to achieve maximum performance:

1. The system has the capability to switch between the current semester and the upcoming semester, and allow user to set schedules in advance without affecting the current schedules.

2. The system has the capability to monitor the current enrollment status such as shown in the figure below

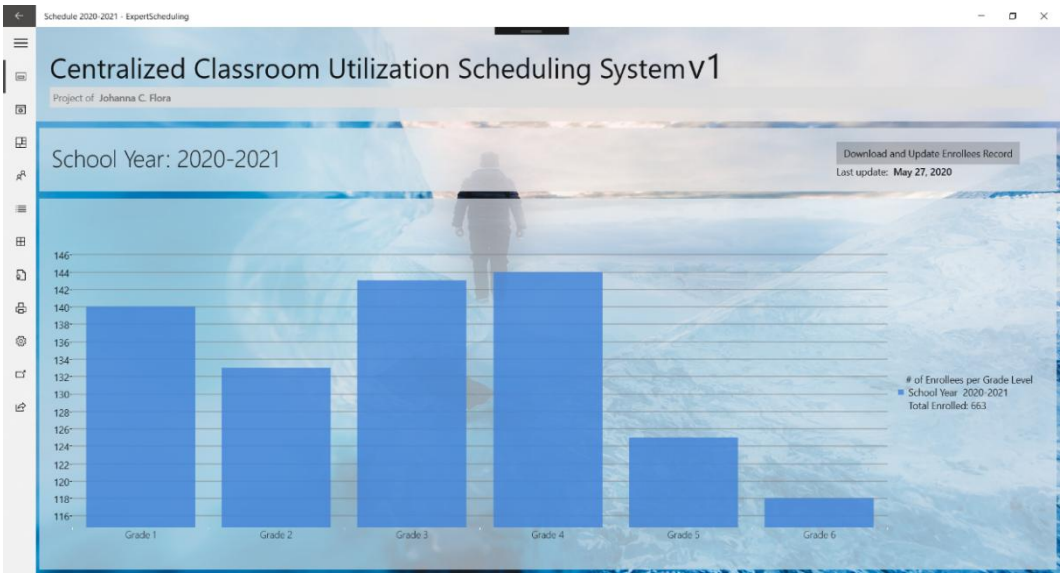


Figure 13. *Monitoring the current enrollment status*

1. The system has the capability to upload the generated schedule to Samar College enrollment system such as what is shown in the Figure below

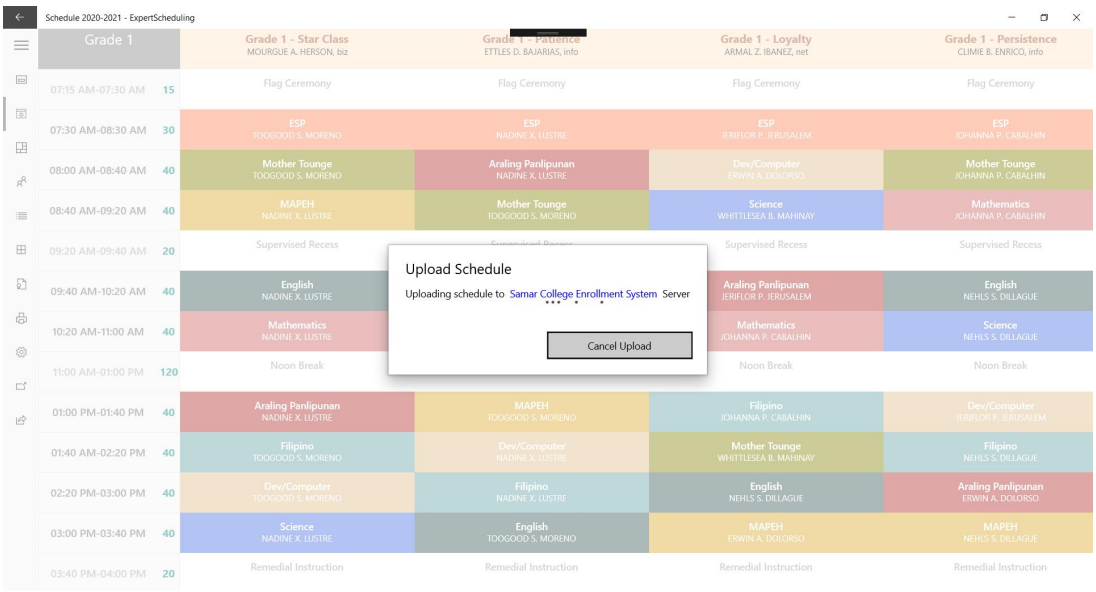


Figure 14. *Generated Schedule to Samar College Enrollment System*

2. The system has the capability of exporting teaching load to portable class record through universal serial bus flash disk drive such as what is shown in the Figure below.

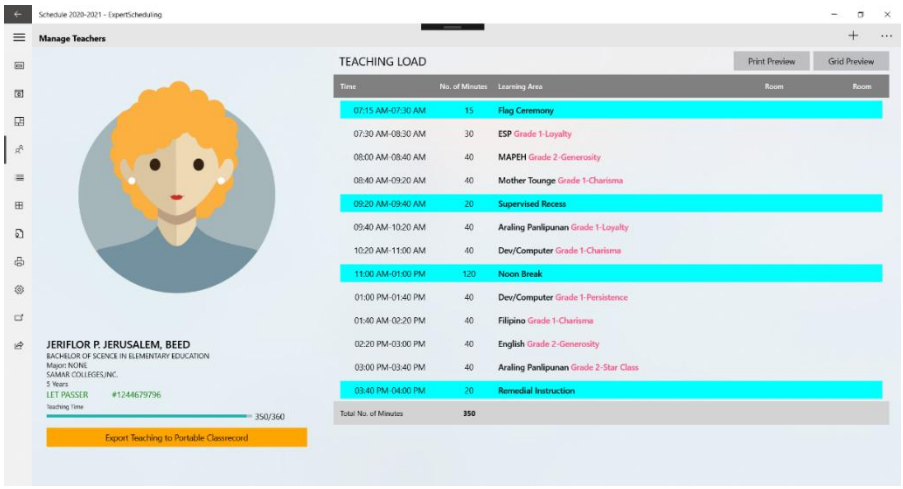


Figure 15. *Exporting teaching load to portable class record*

3. The system has the capability to detect conflicts automatically while plotting schedules.

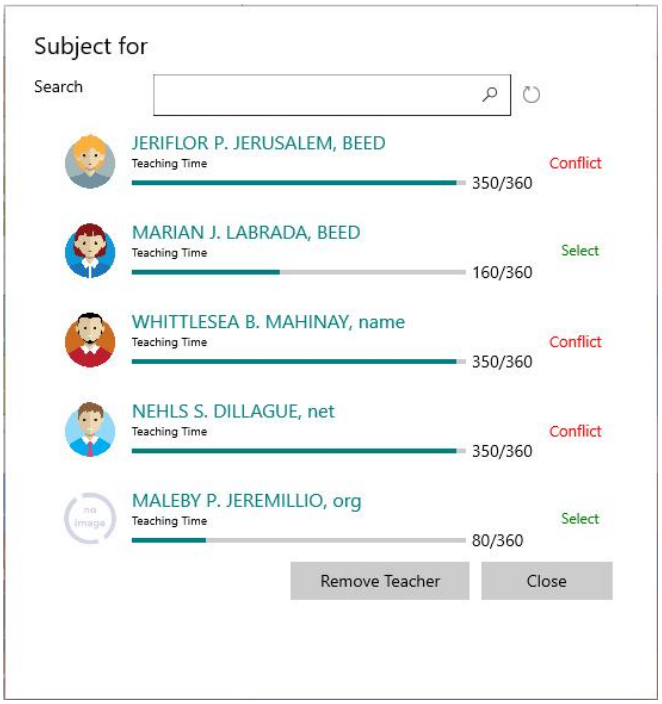


Figure 17. *Detecting conflicts*

3. The system can generate specific reports for the instructor, section, or room schedule

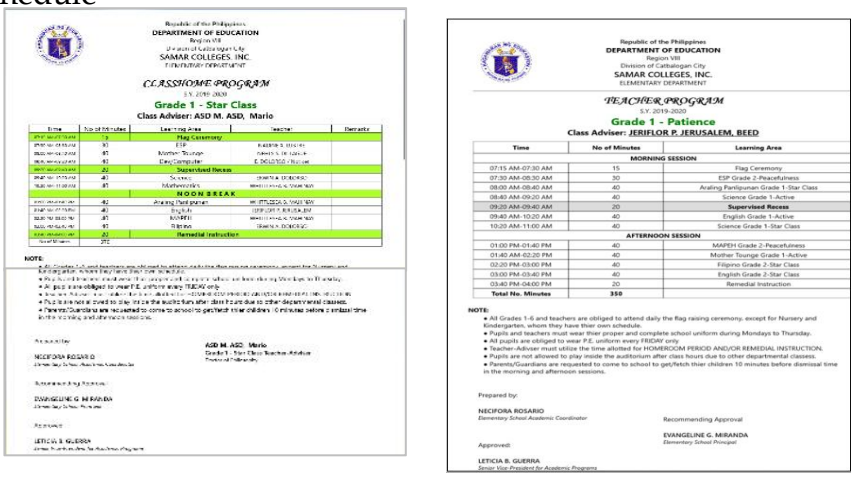


Figure 18. *Teachers Program Print out*

4. The system has capability to export specific schedule reports into MS Excel Application.
5. The system can monitor each classroom if their classes are held through the CCTV that is installed in each classroom.

System Evaluation

Tables 1 to 6 present the evaluation of the two groups of respondents, elementary and junior high school users for the System Quality Attributes as to functionality, reliability, usability, efficiency, maintainability, and portability. The descriptions for each mean of quality attribute indicator is adopted from the Five-Likert Scale of International Standards Organization (ISO)/International Electrotechnical Commission (IEC) 2510:2011, this was similarly used for the

questionnaire provided to the trained users. Where, 1.00 – 1.50 mean rating is described as Very Poor (VP); 1.51 – 2.50 as Poor (P); 2.51 – 3.50 as Fair (F); 3.51 – 4.50 as Good (G) and 4.51 – 5.00 as Excellent (E).

Functionality. For this system quality attribute, as can be gleaned in Table 1, the mean rating of the two groups pegged at 4.51 – 5.00 in which according to ISO/IEC (2011), the functionality of the developed system is excellent. This is supported with combined means of the two groups for the system quality attribute indicators since three out five of it are with descriptive letter E, which refer to excellent functionality of the developed system.

Table 1

System Quality Attributes as to Functionality

Quality Attribute' Indicators	**Elementary Users' Mean**	**Junior High Users' Mean**	**Combined Mean**
1. The proposed system has available all function required for its execution.	4.80 (E)	4.90 (E)	4.85 (E)
2. The proposed system is precise in its results.	4.30 (G)	4.40 (G)	4.35 (G)
3. The proposed system interacts with specified modules.	4.30 (G)	4.70 (E)	4.50 (G)
4. The proposed system complies with standards, laws, etc.	4.90 (E)	4.70 (E)	4.80 (E)
5. The proposed system has secured access through password.	4.80 (E)	4.90 (E)	4.85 (E)
Grand Weighted Mean	**4.62 (E)**	**4.66 (E)**	**4.64 (E)**

Reliability. For this system quality attribute along reliability, as can be viewed in Table 2, the mean rating of the two groups pegged at 4.51 – 5.00 in which according to ISO/IEC standards the reliability of the developed system is excellent. This is supported by the combined means of both users from elementary and junior high since three out of three system quality attribute indicators are with descriptive letter E, which refer to excellent reliability of the developed system regardless of the groups who rated the system and the number of indicators provided.

Table 2

System Quality Attributes as to Reliability

Quality Attribute' Indicators	**Elementary Users' Mean**	**Junior High Users' Mean**	**Combined Mean**
1. The proposed system reacts appropriately when failure occurred.	4.90 (E)	4.90 (E)	4.90 (E)
2. The proposed system informs user concerning invalid data entry.	4.40 (E)	4.90 (E)	4.65 (E)
3. The proposed system is capable of recovering data in the event of failure.	4.70 (E)	4.80 (E)	4.75 (E)
Grand Weighted Mean	**4.66 (E)**	**4.87 (E)**	**4.77 (E)**

Usability. When it comes to system quality attribute, usability. It is display in Table 3 that the mean rating of the two groups pegged at 4.51 – 5.00 in which according to ISO/IEC standards, the usability of the developed system is excellent.

This is supported by the combined means of both users from elementary and junior high since three out of three system quality attribute indicators are with descriptive letter E, which again into excellent usability of the developed system.

The results further describe that the Classroom Management using Expert System (CMES) is user-friendly, it is easy to use and can be easily manipulated, the elements of informational components (indicator 1), navigational component (indicator 2) and input controls (indicator 3) are featured in CMES.

Table 3

System Quality Attributes as to Usability

Quality Attribute' Indicators	**Elementary Users' Mean**	**Junior High Users' Mean**	**Combined Mean**
1. It's easy to understand the concept and application.	4.80 (E)	4.80 (E)	4.80 (E)
2. It's easy to learn how to use.	4.90 (E)	4.60 (E)	4.75 (E)
3. It's easy to operate and control.	4.70 (E)	4.80 (E)	4.75 (E)
Grand Weighted Mean	**4.80 (E)**	**4.73 (E)**	**4.76 (E)**

Efficiency. As to the System Quality Attribute- efficiency, it is presented in Table 4 the indicators and scores provided by users from elementary and junior high school groups.

Table 4

System Quality Attributes as to Efficiency

Quality Attribute' Indicators	**Elementary Users' Mean**	**Junior High Users' Mean**	**Combined Mean**
1. The system response time is appropriate.	4.80 (E)	4.70 (E)	4.75 (E)
2. The system execution time is appropriate.	4.70 (E)	4.80 (E)	4.75 (E)
3. The resources used are appropriate.	4.60 (E)	4.60 (E)	4.60 (E)
Grand Weighted Mean	**4.70 (E)**	**4.70 (E)**	**4.70 (E)**

From the table, the mean rating of the two groups pegged at 4.51 – 5.00 in which according to ISO/IEC standards, the efficiency of the developed system is excellent. This is supported by the combined means of both users from elementary and junior high since three out of three system quality attribute indicators are with descriptive letter E, which refer to excellent efficiency of the developed system regardless of the groups who rated the system and the number of indicators provided.

Maintainability. When it comes to system quality attribute, usability. It is display in Table 3 that the mean rating of the two groups pegged at 4.51 – 5.00 in which according to ISO/IEC standards, the maintability of the developed system is excellent. This is supported by the combined means of both users from

elementary and junior high since three out of three system quality attribute indicators are with descriptive letter E, which again into excellent maintainability of the developed system.

Table 5

System Quality Attributes as to Maintainability

Quality Attribute' Indicators	**Elementary Users' Mean**	**Junior High Users' Mean**	**Combined Mean**
1. It's easy to find failure when it occurs.	4.70 (E)	4.70 (E)	4.70 (E)
2. It's easy to modify and adopt.	4.90 (E)	4.70 (E)	4.80 (E)
3. Changes are easy to test.	4.60 (E)	4.60 (E)	4.60 (E)
Grand Weighted Mean	**4.73 (E)**	**4.67 (E)**	**4.70 (E)**

Portability. There are four indicators for the System Quality Attribute - Portability, as shown in Table 6, all the combined mean of each indicator range from 4.51 – 5.00, a standard quality portability attribute as excellent. It is supported by the grand weighted mean of 4.79 in which it also falls on the said range and standards.

Table 6

System Quality Attributes as to Portability

Quality Attribute' Indicators	**Elementary Users' Mean**	**Junior High Users' Mean**	**Combined Mean**
1. It's easy to adopt with other environment.	4.90 (E)	4.80 (E)	4.85 (E)
2. It's easy to install in other environment.	4.90 (E)	4.80 (E)	4.85 (E)
3. It is in agreement with portability standard	4.70 (E)	4.90 (E)	4.80 (E)
4. It's easy to use to replace another program	4.70 (E)	4.60 (E)	4.65 (E)
Grand Weighted Mean	**4.80 (E)**	**4.78 (E)**	**4.79 (E)**

It is presented in Table 7 the validation of results for each system quality attribute, the researcher conducted validation test with the help of statistical tool (t-test for assumed unequal variances) to prove the null hypothesis *there is no significant difference on the ratings of the two groups of respondents*. Once this is proven accepted, it means that the system has followed the ISO/IEC standards and which part of the system was overdone or needs improvement.

Table 7

Validation of System Quality Attributes

System Quality Attributes	Group Means		t-critical	df	Computed t-value	Decision/ Interpretation
	Elem	Junior				
Functionality	4.62	4.66	2.11	18	-0.87	Accept Ho/NS
Reliability	4.66	4.87			-4.18	Reject Ho/S
Usability	4.80	4.73			-0.03	Accept Ho/NS
Efficiency	4.70	4.70			0.00	Accept Ho/NS
Maintainability	4.73	4.67			-0.02	Accept Ho/NS
Portability	4.70	4.68			-0.03	Accept Ho/NS

Legend: *Significant (S); Reject Ho if t-computed is higher than the critical value at 0.05 margin of error*
Not Significant; Accept Ho if t-computed is lower than the critical value at 0.05 margin of error

From the result, all the system quality attributes rated by the two groups have no significant differences (NS), however it is detected that there is a significant difference (S) between the rating of users from the elementary group and junior high school group. The users from the junior high school group found reliability as excellent while the user from the elementary group found it good, specifically on the system quality attribute indicator 2, the proposed system informs user concerning invalid data entry (kindly refer to Table 2). Thus, it is deemed necessary to improve this attribute so the developed system can fully comply to ISO/IEC 2510:2011 standards.

Chapter 5

SUMMARY OF FINDINGS, CONCLUSIONS AND RECOMMENDATIONS

This chapter presents the summary and conclusion of the study derived from the findings both for the development and evaluation of Classroom Management using Expert System (CEMS).

Summary of Findings

A rule-based expert system was successfully developed for classroom management. The human decision-making process required to perform a classroom management task was divided into its basic parts. The reasoning derived from these elements was reproduced by a set of rules contained in each of the logic module. The logic module was combined with a user-friendly interface to effectively and reliably emulate the class scheduling process. With this and in reference to the different results presented in the previous chapter, the following summary of the results are made:

1. A Classroom Management System using Expert system (CMES) for Elementary and Junior High School Department was designed and developed.

2. CMES could provide a well-distributed classroom management with full report and could detect conflicts on plotting the schedules.

3. The system software was designed and coded usingC# and SQL light which executed the logical processes execution of the system.

4. The system has a projection scheme which could project the total number of population to accommodate during enrolment period.

5. The system has a capability of generating results in less time.

6. The system has capability of interacting with other system technology.

7. Aside from being usable, the system also has the ability to adapt in any devices such as a desktop computer, a mobile phone, or a tablet, provided that they have a Windows platform.

Conclusions

Classroom management using expert system (CMES) is beneficial to the success of the entire institution especially in elementary and junior high school department. The system makes tasks more productive and effective in a way that it can amplify the work of human employees. This ultimately increases productivity and show better results at a lower expense. It is more time-efficient and gives employees more time for other priorities. Because of its usability, the user can be any of the staff or faculty in the institution. It does not need an expert user in operating the system. The CMES conformed with the International Standards Organization (ISO)/International Electrotechnical Commission (IEC) 2510:2011 as per rated by the users.

Recommendation

It is a recommendation to validate the developed system since there is a need to improve system quality attribute as to functionality specifically on precision of generated result to fully conform with the International Standards Organization (ISO)/International Electrotechnical Commission (IEC) 2510:2011 as per rated by the users.

B I B L I O G R A P H Y

A. Books/Published Journals

Abcede, K. J., Gonzaga, R. A., Ontoca, R. J., Zapanta, J. N., & Abante, M. V. (2013). Class Scheduling System for URS – Binangonan Campus. *Fatima University Research Journals*, *5*(1).

Botangen, K., & Khan, C. L. (2014). Class-Scheduling System for the Central Luzon State University(Abridged). *26th CLSU In-House Review of Completed and On-Going R and D Projects*. https://doi.org/10.13140/RG.2.2.12729.57445

Engin, G., Aksoyer, B., Avdagic, M., Bozanlı, D., Hanay, U., Maden, D., & Ertek, G. (2014). Rule-based Expert Systems for Supporting University Students. *Procedia Computer Science*, *31*, 22–31. https://doi.org/10.1016/j.procs.2014.05.241

Guyette, L., Hamidian, K., & Tuazon, J. O. (2014). A rule-based expert system approach to class scheduling. *Computers & Electrical Engineering*, *20*(2), 151–162. https://doi.org/10.1016/0045-7906(94)90027-2

International Standards Organization (ISO, & International Electrotechnical Commission . (2011). ISO. https://www.iso.org/obp/ui/#iso:std:iso-iec:25010:ed-1:v1:en.

Ho, K. K., & Lu, M. (2005). Web-based expert system for class schedule planning using JESS. *IRI -2005 IEEE International Conference on Information Reuse and Integration, Conf, 2005*. https://doi.org/10.1109/iri-05.2005.1506468

Lee, S.-J., & Wu, C.-H. (1995). CLXPERT: A rule-based scheduling system. *Expert Systems with Applications*, *9*(2), 153–164. https://doi.org/10.1016/0957-4174(94)00058-4

López-Santana, E. R., & Méndez-Giraldo, G. A. (2016). A Knowledge-Based Expert System for Scheduling in Services Systems. *Communications in Computer and Information Science*, *657*, 212–224. https://doi.org/10.1007/978-3-319-50880-1_19

Usman, Y. D. (2016). Educational Resources: An Integral Component for Effective School Administration in Nigeria. *Research on Humanities and Social Sciences*, *6*(13), 27–37.

B. Theses/Dissertations

Khalil, M. (2018). *Learning Analytics on Massive Open Online Courses* (dissertation). Graz University of Technology, Styria, Austria.

Majiwan, F. (2005). *Time Table Scheduling Using Rule-Based Technique* (thesis). University of College and Engineering & Technology , Malysia.

C. Online/Other References

Ed Glossary. (2014, November 26). *Classroom Management Definition*. The Glossary of Education Reform. https://www.edglossary.org/classroom-management/.

Johnston, T. (2021). Classroom Management. https://www2.palomar.edu/pages/tjohnston2/tag/classroom-management/.

Montoya, S., & Barbosa, A. (2020, May 19). *The Importance of Monitoring and Improving ICT Use in Education Post-Confinement*. UNESCO INSTITUTE FOR STATISTICS. http://uis.unesco.org/en/blog/importance-monitoring-and-improving-ict-use-education-post-confinement.

Stronge, J. H., Tucker, P. D., & Hindman, J. L. (2021). *Chapter 3. Classroom Management and Organization*. Classroom Management and Organization. http://www.ascd.org/publications/books/104135/chapters/Classroom-Management-and-Organization.aspx.

Suarez Chilma, V. F., Castrillón Gomez, O. D., & Guerrero Aguirre, Á. (2013). *Class Schedule Assignment Based on Students Learning Rhythms Using A Genetic Algorithm*. Ingeniería y Ciencia. http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S1794-91652013000100005.

APPENDICES

Appendix A

SYSTEM MANUAL

Classroom Management using Expert System for Elementary and Junior High School

Welcome to a new ways of classroom Management.

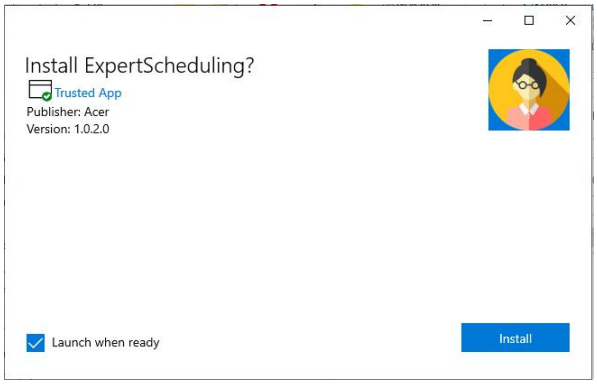
CMES

User's Manual

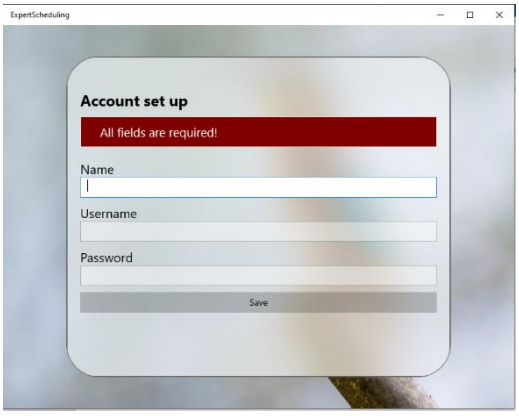
First run

Step-by-step Set up

1. First is the installation of the system. Click install

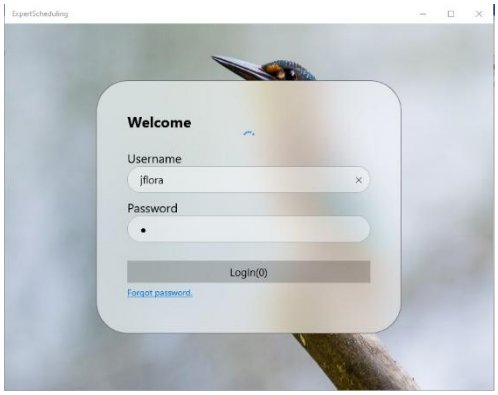


2. Set up your user's account



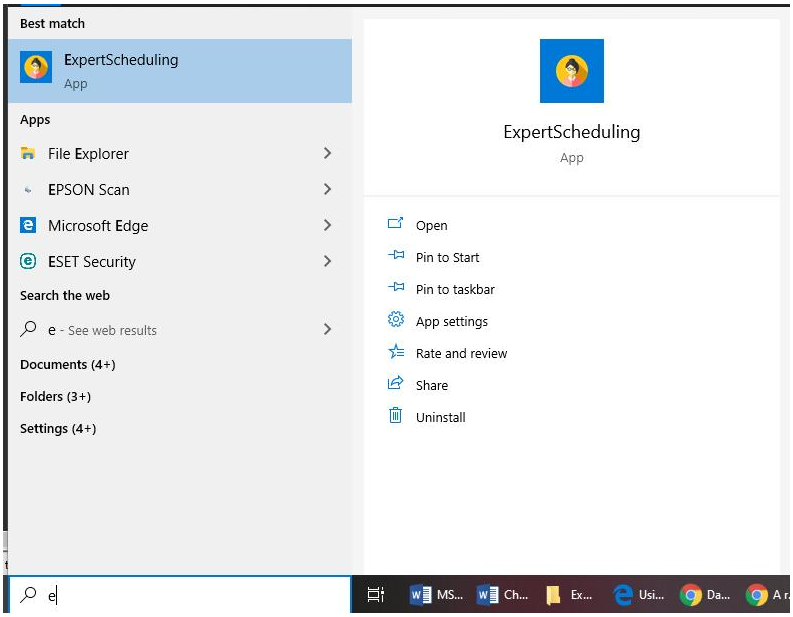
**Note: After filling out all fields, click save.*

3. Log in

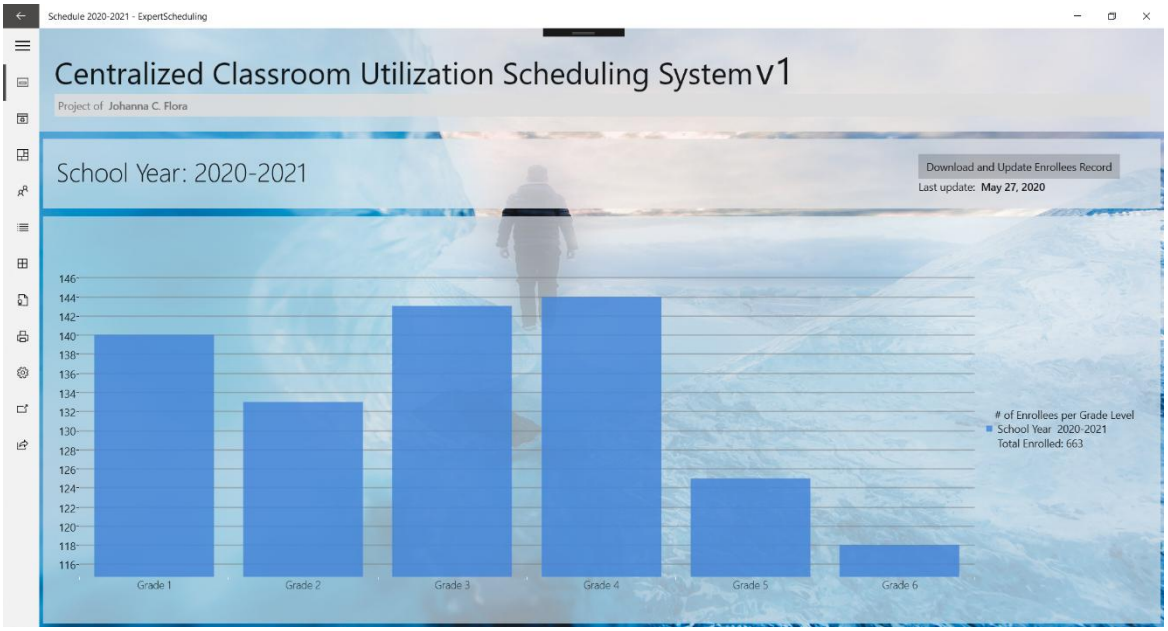


1 Launching the Class Management using Expert System (CMES)

To launch the program, just type Expert Scheduling on windows search bar



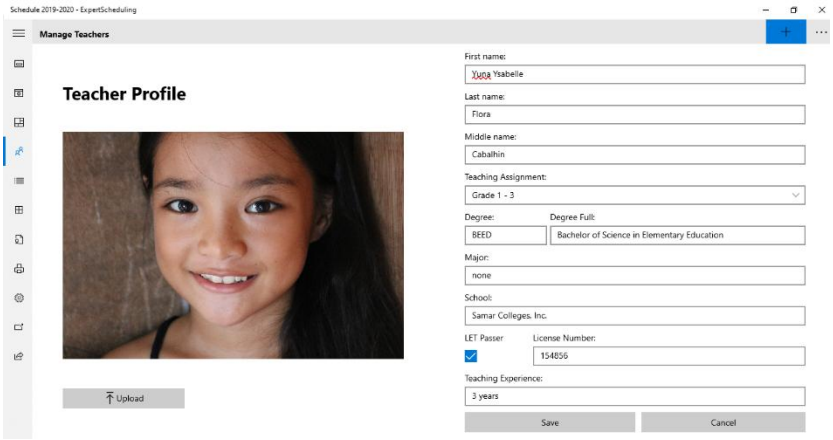
The main dashboard of the system is shown below



2 Managing Teacher's File

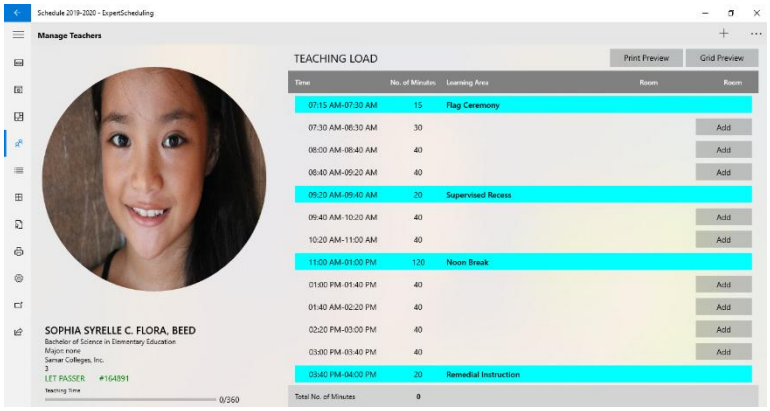
To create teacher's file

- Click on this icon 
- Then, click 
- Fill out all teachers' details Click save



To open Teacher's File

- Click on Manage Teacher icon
- Click on the Teacher's Icon/Picture 
- View teacher's File



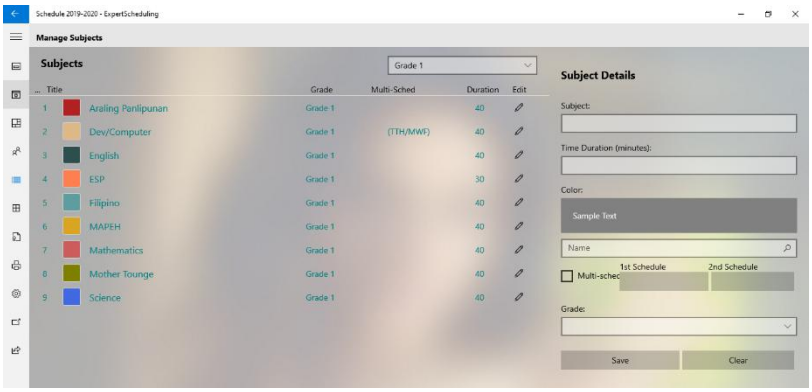
3 Managing Subjects

To create subject file

- Click on this icon 
- Fill out all the details
- Click save

To open subject file

- Click on “manage subject” icon 
- View subjects per grade level

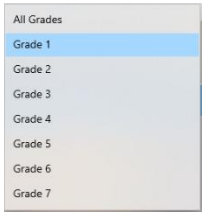


To edit a subject file

- Click on the “manage subject” icon 
- Click on this icon 
- Click save

To view subjects for different grade levels

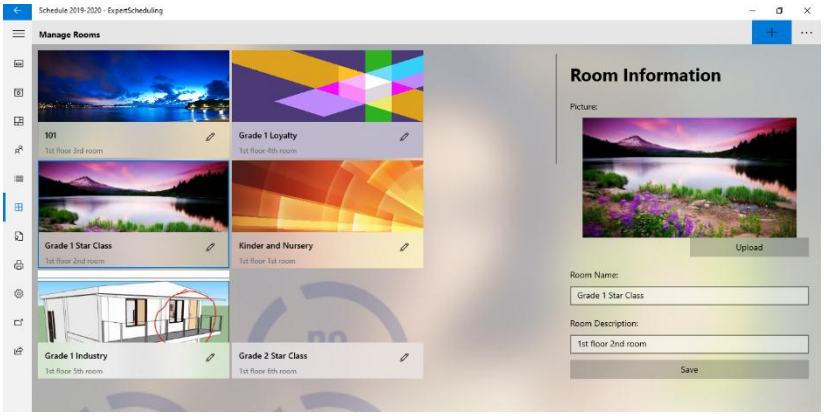
- Click on the “manage subject” icon
- Click on this icon 

- Choose grade level 

4 Managing Classroom File

To create a classroom file

- Click on the “manage classroom” icon 
- Then, click the plus sign icon 
- Fill out all the details
- Click save



To edit classroom

- Click on the manage classroom icon
- Click on this icon 
- Edit, then click save

5 Managing Class Section File

CMES is an auto generated system for adding class section file, but if the user wants to do it manually, the following instructions should be followed.

To create class section

- Click on the manage section icon
- Click on this icon to select for particular grade level

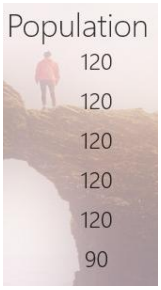
- Choose by clicking the desired grade level

- Fill out all the details
- Click save

6 Projecting Scheme Generator

To project the total number of population

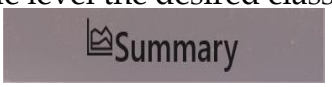
- Click and type for every grade level the desired section
- Click and type for every grade level the desired class size
- View the total number of population in every grade level



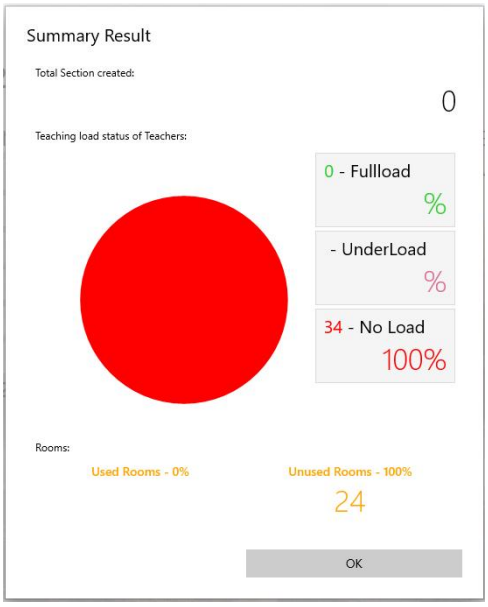
To view summary result of classroom and teacher's distribution

- Click on the projection scheme generator icon

- Click and type in every grade level the desired section
- Click and type in every grade level the desired class size
- Click on the icon summary



- View the summary report



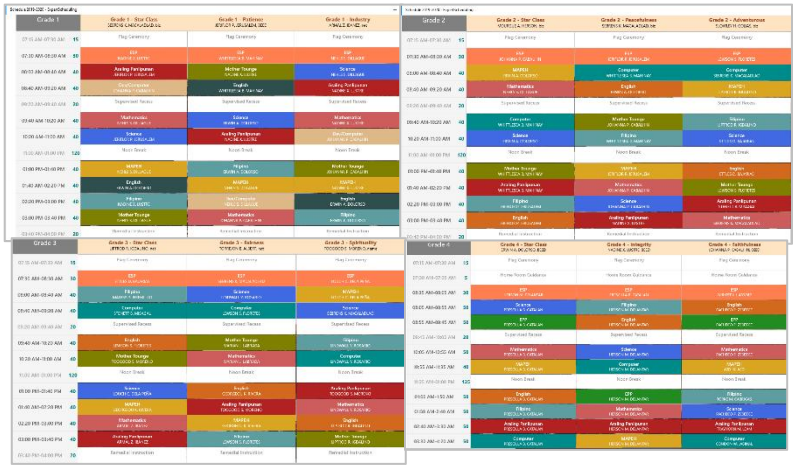
7 Generating Schedules

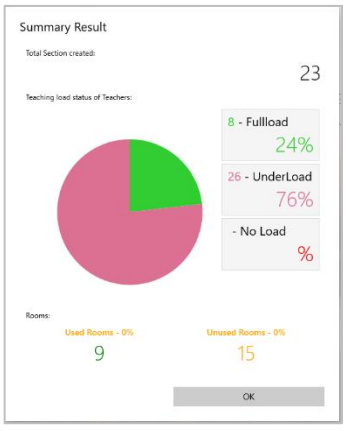
To generate schedule

- Click on the projection scheme generator icon

- Click and type in every grade level the desired section
- Click and type in every grade level the desired class size
- Click on the Generate icon

- View Generated schedule

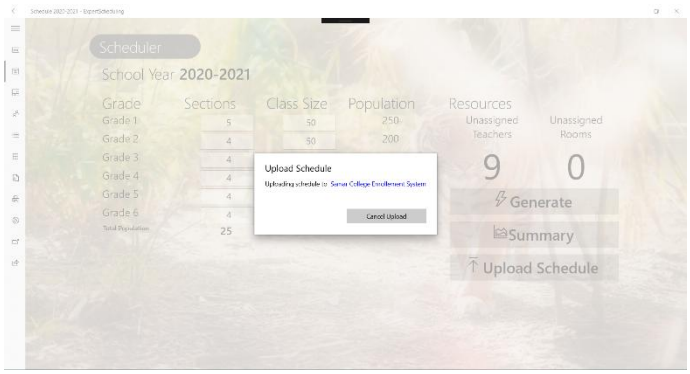


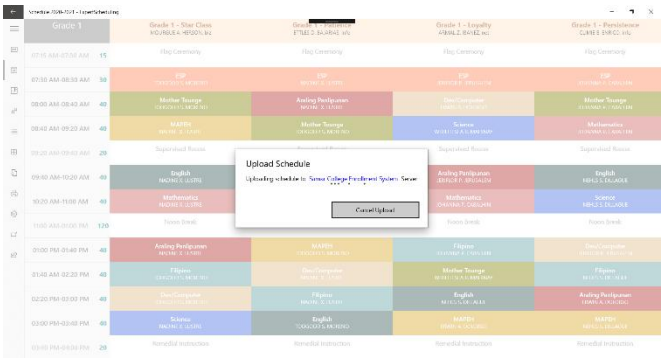


8 Interacting with Samar College Enrollment System

Uploading Class Schedule

- Click upload schedule on the projection generator scheme

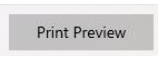




9 Printing

Printing Options

To print teachers' program

- Click on manage Teachers Icon 
- Click on teacher's photo 
- Click on print preview 

- Click print 

Republic of the Philippines
DEPARTMENT OF EDUCATION
Region VIII
Division of Catbalogan City
SAMAR COLLEGES, INC.
ELEMENTARY DEPARTMENT

TEACHER PROGRAM
S.Y. 2019-2020

Grade 1 - Patience
Class Adviser: JERIFLOR P. JERUSALEM, BEED

Time	No of Minutes	Learning Area
MORNING SESSION		
07:15 AM-07:30 AM	15	Flag Ceremony
07:30 AM-08:30 AM	30	ESP Grade 2-Peacefulness
08:00 AM-08:40 AM	40	Araling Panlipunan Grade 1-Star Class
08:40 AM-09:20 AM	40	Science Grade 1-Active
09:20 AM-09:40 AM	20	Supervised Recess
09:40 AM-10:20 AM	40	English Grade 1-Active
10:20 AM-11:00 AM	40	Science Grade 1-Star Class
AFTERNOON SESSION		
01:00 PM-01:40 PM	40	MAPEH Grade 2-Peacefulness
01:40 AM-02:20 PM	40	Mother Tounge Grade 1-Active
02:20 PM-03:00 PM	40	Filipino Grade 2-Star Class
03:00 PM-03:40 PM	40	English Grade 2-Star Class
03:40 PM-04:00 PM	20	Remedial Instruction
Total No. Minutes	350	

NOTE:
- All Grades 1-6 and teachers are obliged to attend daily the flag raising ceremony, except for Nursery and Kindergarten, whom they have thier own schedule.
- Pupils and teachers must wear thier proper and complete school uniform during Mondays to Thursday.
- All pupils are obliged to wear P.E. uniform every FRIDAY only
- Teacher-Adviser must utilize the time allotted for HOMEROOM PERIOD AND/OR REMEDIAL INSTRUCTION.
- Pupils are not allowed to play inside the auditorium after class hours due to other departmental classes.
- Parents/Guardians are requested to come to school to get/fetch thier children 10 minutes before dismissal time in the morning and afternoon sessions.

Prepared by:

NECIFORA ROSARIO
Elementary School Academic Coordinator

Recommending Approval

EVANGELINE G. MIRANDA
Elementary School Principal

Approved:

LETICIA B. GUERRA
Senior Vice-President for Academic Programs

To Print Class Home Program

- Click on this icon on the dashboard
- Search for the grade level 

- Click print icon 

10 Interacting with Portable Class Record System

Opening Teachers' File

- Click on Manage Teacher icon
- Click on export button 

Source Code

Color Class

```
using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using Windows.UI;

using Windows.UI.Xaml.Media;

namespace HamburgerMenuDemo.Models

{

   class AkonColorClass

   {

      public SolidColorBrush Color { get; set; }

      public String ColorText { get; set; }

   }

}

using System;
```

```
using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using Windows.UI;

using Windows.UI.Xaml.Data;

using Windows.UI.Xaml.Media;

namespace JFloraClassScheduler.Models

{

   class BGConverter: IValueConverter

   {

      public object Convert(object value, Type targetType, object parameter, string
language)

      {

         return new SolidColorBrush( (value.ToString() == "YES" ?
Colors.Transparent: Colors.Cyan ) );

      }
```

```
        public object ConvertBack(object value, Type targetType, object parameter,
string language)
        {
            throw new NotSupportedException();
        }
    }
}
```

Class period

```
using SQLite.Net.Attributes;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace HamburgerMenuDemo.Models
{
    public class ClassPeriod
```

```
    {
        [PrimaryKey, AutoIncrement]
        public int id { get; set; }
        public int period { get; set; }
        public string start_time { get; set; }
        public string end_time { get; set; }
        public string Level { get; set; }
        public string minutes { get; set; }
        public string regular { get; set; }
        public string last_update { get; set; }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
```

```
using Windows.UI;
using Windows.UI.Xaml.Media;
namespace HamburgerMenuDemo.Models
{
    public class ClassPeriodWithInfo : ClassPeriod
    {
        private ClassPeriod t;
        public ClassPeriodWithInfo(ClassPeriod t)
        {
            this.t = t;
            this.regular = t.regular;
            this.period = t.period;
            this.Duration = t.minutes;
        }
        public string GetID()
        {
            return t.id.ToString();
```

```
        }
        public String TimeText {
            get { return t.start_time + "-" + t.end_time; }
            set { }
        }
        public String Duration {
            get { return t.minutes; }
            set { }
        }

        Color _SubjectColor = Colors.Gainsboro;
        Color _SubjectForeColor = Colors.DarkGray;
        public bool IsRegularTime { get { return t.regular == "YES"; }   set { } }
        public SolidColorBrush SubjectColor { get { return new
SolidColorBrush((t.regular == "YES" ? Colors.WhiteSmoke :
Colors.WhiteSmoke)); } set { } }
        public SolidColorBrush SubjectForeColor { get {
```

```
            return new SolidColorBrush((t.regular == "YES" ? Colors.DimGray :
_SubjectForeColor));
        }
        set { } }
    }
}
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Windows.UI.Xaml.Data;
namespace HamburgerMenuDemo.Models
{
    class ColumnSpanChecker : IValueConverter
    {
        public object Convert(object value, Type targetType, object parameter, string
culture)
```

```
        {
            return ((bool)value ? 1 :3);
        }
        public object ConvertBack(object value, Type targetType, object parameter,
string culture)
        {
            throw new NotImplementedException();
        }
    }
}
*//Database Manager//*
using System.Text;
using System.Threading.Tasks;
using Windows.UI.Xaml.Media.Imaging;
using SQLite;
using SQLite.Net;
using RFloraClassScheduler.Models;
```

```
using System;

using System.IO;

using System.Linq;

using System.Collections.Generic;

using ExpertScheduling;

using ExpertScheduling.Models;

using ExpertScheduling.Pages;

namespace HamburgerMenuDemo.Models

{

   class DBManager

   {

      static SQLiteConnection fconn;

      public static string path() {

         return Windows.Storage.ApplicationData.Current.LocalFolder.Path;

      }

      public static String  FirstRun()

      {
```

```
        string config_path =
Path.Combine(Windows.Storage.ApplicationData.Current.LocalFolder.Path,
"config.rflora");

        fconn = new SQLiteConnection(new
SQLite.Net.Platform.WinRT.SQLitePlatformWinRT(), config_path);

        MainPage.setFConn(fconn);

        fconn.CreateTable<Models.Schoolyear>();

        var sy = fconn.Table<Schoolyear>().ToList<Schoolyear>();

        if( sy.Count == 0)

        {

           fconn.Insert(new Schoolyear() { Active ="0", SchoolYear="2018-2019" });

           fconn.Insert(new Schoolyear() { Active ="1", SchoolYear="2019-2020" });

           fconn.Insert(new Schoolyear() { Active ="0", SchoolYear="2020-2021" });

           fconn.Insert(new Schoolyear() { Active ="0", SchoolYear="2021-2022" });

           fconn.Insert(new Schoolyear() { Active ="0", SchoolYear="2022-2023" });

           fconn.Insert(new Schoolyear() { Active ="0", SchoolYear="2023-2024" });

        }
```

```
            fconn.CreateTable<ExpertScheduling.Models.SchoolYearConfig>();
            fconn.CreateTable<Models.Subject>();
            DBManager.initializeTableData("Subject");
            fconn.CreateTable<Models.ClassPeriod>();
            DBManager.initializeTableData("ClassPeriod");
            fconn.CreateTable<Models.Room>();
            fconn.CreateTable<Models.Teacher>();
            fconn.CreateTable<Models.Setting>();
            DBManager.initializeTableData("Setting");

            sy = fconn.Table<Schoolyear>().ToList<Schoolyear>();
            Schoolyear ActiveSchoolyear = sy.Where(saaa => saaa.Active ==
"1").FirstOrDefault();

            MainPage.SetTitleText("Schedule " + ActiveSchoolyear.SchoolYear);
            MainPage.setSchoolYear(ActiveSchoolyear);
```

```
            var path =
Path.Combine(Windows.Storage.ApplicationData.Current.LocalFolder.Path,
"qwieuyqweujasdad423423ASSDFWER234"+ActiveSchoolyear.id.ToString() +
".rflora"); ;


           var conn = new SQLite.Net.SQLiteConnection(new
SQLite.Net.Platform.WinRT.SQLitePlatformWinRT(), path);
            conn.CreateTable<Models.Schedule>();
            conn.CreateTable<Models.Section>();
            conn.CreateTable<PaperWorks>();
            MainPage.SetActiveSyDBConnection ( conn );
            return path;
        }
        internal static List<Schedule> GetSectionClassSchedule(int id)
        {
            return  MainPage.GetActiveSyDBConnection().Query<Schedule>("select *
from Schedule where section_id =?", id);
        }
```

```
        internal static void UpdateClassSchedule(object fieldToUpdate, object klasi,
object id)
        {
            throw new NotImplementedException();
        }
        internal static List<PaperWorks> getTeachersPaperWorks(string v)
        {
            return
MainPage.GetActiveSyDBConnection().Query<PaperWorks>("select * from
PaperWorks where TeacherID =?", v);
        }
        internal static int GetTotalUserRoom()
        {
            return MainPage.GetActiveSyDBConnection().ExecuteScalar<int>("select
count(*) from Schedule group by room_id;");
        }
        internal static List<SchoolYearConfig> GetConfiguration(int id)
        {
```

```
        return MainPage.GetDBConnection().Query<SchoolYearConfig>("select *
from SchoolYearConfig where SchoolYearId =?", id);
    }
    internal static void UpdateSettings(Setting setting)
    {
        MainPage.GetDBConnection().Update(setting);
    }
    internal static void initializeTableData(String table)
    {
        if( table == "Setting")
        {
            var settings = GetSettings();
            if(settings == null)
            {


                    fconn.Execute(TableStructure.Settings );
```

```
    }
}

if( table == "Subject")
{
    var subjects = GetSubjects();
    if( subjects.Count ==0)
    {
        String[] data = TableStructure.SubjectData.Split(';');
        foreach (var item in data)
        {
            if (item != "")
                fconn.Execute(item);
        }
    }
}
if( table == "ClassPeriod")
```

```
    {
        var ClassPeriodData = GetClassPeriod("1");
        if (ClassPeriodData.Count == 0)
        {
            String[] data = TableStructure.ClassPeriodData.Split(';');
            foreach (var item in data)
            {
                if(item!="")
                fconn.Execute(item);
            }


        }
    }
}
```

```
internal static void UpdateSchoolYearConfig(string text1, string text2, int id,
string grade)
{
    SchoolYearConfig schoolYearConfig =
MainPage.GetDBConnection().Query<SchoolYearConfig>("select * from
SchoolYearConfig  WHERE SchoolYearId=? AND GradeLevel=?", id,
grade).FirstOrDefault();
    if(schoolYearConfig != null)
    {
        schoolYearConfig.SectionCount = text1;
        schoolYearConfig.ClassSize = text2;
        MainPage.GetDBConnection().Update(schoolYearConfig);
    }
    else
    {
        schoolYearConfig = new SchoolYearConfig() { GradeLevel = grade,
SchoolYearId = id, ClassSize = text2, SectionCount = text1 };
        MainPage.GetDBConnection().Insert(schoolYearConfig);
```

```
        }

        //MainPage.GetActiveSyDBConnection().Execute("REPLACE INTO
positions (SectionCount, ClassSize) VALUES(?, ?) ; ", text1,  text2,  id,  grade);

    }

    public static Setting GetSettings()

    {

        return fconn.Table<Setting>().ToList<Setting>().FirstOrDefault();

    }

    public static  List<TeacherProfile> GetTeachersAsync(int period_id = -1,
string gradeLevel = "" )

    {

        List<TeacherProfile> listp = new List<TeacherProfile>();

        List<Teacher> list = new List<Teacher>();

        if(gradeLevel != "")

        {

            string k = MiscUtils.getTeaching(gradeLevel);

            list = MainPage.GetDBConnection().Query<Teacher>("select * from
Teacher where teaching_assingment =?", k );
```

```
        }
        else
        {
            list =
MainPage.GetDBConnection().Table<Teacher>().ToList<Teacher>();
        }

        foreach( Teacher t in list)
        {
            List<Schedule> tSchedule = null;
            if (period_id>=0)
            {
                tSchedule =
MainPage.GetActiveSyDBConnection().Query<Schedule>("select * from
Schedule where teacher_id =?", t.id);
            }
            listp.Add(new TeacherProfile(t, tSchedule, period_id));
```

```
    }
    return listp;
}
internal static void SavePaperWork(PaperWorks paperWorks)
{
    if (paperWorks.id == 0)
    {
        MainPage.GetActiveSyDBConnection().Insert(paperWorks);
    }
    else
    {
        MainPage.GetActiveSyDBConnection().Update(paperWorks);
    }
}
internal static List<Schoolyear>  getSchoolYear()
{
```

```
        return MainPage.GetFConn().Table<Schoolyear>().ToList<Schoolyear>();



    }
    internal static List<ClassPeriodWithInfo> getClassPeriodSample( string lvl)
    {
        List<ClassPeriod> classp =
MainPage.GetDBConnection().Query<ClassPeriod>("select * from classperiod
where level =? order by period", lvl);
        List<ClassPeriodWithInfo> cp = new List<ClassPeriodWithInfo>();
        foreach (var item in classp)
        {
            cp.Add(new ClassPeriodWithInfo(item));
        }
        return cp;
    }
    internal static void DeletePaperWork(PaperWorks paperWorks)
    {
```

```
        MainPage.GetActiveSyDBConnection().Query<PaperWorks>("DELETE
FROM PAPERWORKS WHERE ID=?",paperWorks.id);
    }
    internal static  ClassPeriod  getClassPeriod( string id)
    {
        return MainPage.GetDBConnection().Query<ClassPeriod>("select * from
classperiod where id =?", id).FirstOrDefault();
    }
    internal static void setActiveSchoolYear(Schoolyear nActive)
    {
        MainPage.GetFConn().Execute("UPDATE Schoolyear SET Active= ?
Where Id>0", "");
        MainPage.GetFConn().Execute("UPDATE Schoolyear SET Active= ?
Where Id = ?", 1, nActive.id);
        FirstRun();
    }
    internal static Subject GetSubject(string subject_id)
    {
```

```
        return MainPage.GetDBConnection().Query<Subject>("select * from
Subject where id =?", subject_id).FirstOrDefault();
    }
    internal static List<Subject> GetClassSubjects(string level)
    {
        return MainPage.GetDBConnection().Query<Subject>("select * from
Subject where grade =? or grade=-1", level);
    }
    internal static List<TSchedTeacher> GetTSchedTeacher()
    {
        return
MainPage.GetActiveSyDBConnection().Query<TSchedTeacher>("select count(*)
as schecCount, teacher_id  from Schedule group by teacher_id;");
    }
    internal static int GetTotalSection()
    {
        return MainPage.GetActiveSyDBConnection().ExecuteScalar<int>("select
count(*) from Section;") ;
```

```
        }

        internal static  Section GetSection(String id)

        {

          return  MainPage.GetActiveSyDBConnection().Query<Section>("select *
from Section where id =?", id).FirstOrDefault();

        }



        internal static List<SectionComplete> GetSections( String sql = "")

        {

           List<SectionComplete> listp = new List<SectionComplete>();

           List<Section> list = new List<Section>();

           if(sql == "")

           {

              list =
MainPage.GetActiveSyDBConnection().Table<Section>().ToList<Section>();

           }

           else

           {
```

```
            list = MainPage.GetActiveSyDBConnection().Query<Section>( sql );
        }


        list = list.OrderBy(sc => sc.grade_level ).ToList();
        int ctr = 1;
        foreach (Section t in list)
        {
            // BitmapImage img = GetBitmapIgAsync( t.imgArray ).Result;
            String AdviserName = "No Adviser";
            if(t.adviser_id != "0")
            {
                Teacher teacher = DBManager.GetTeacher(t.adviser_id);
                AdviserName = new TeacherProfile(teacher,null, -
1).fullname_with_degree;
            }
```

```
            listp.Add(new SectionComplete( t)

             {

               ctr = ctr,

            id = t.id,

               Adviser = AdviserName



            });

          ctr++;

       }

       listp = listp.OrderBy(grade => grade.grade_level).ToList();

       return listp;

    }

    internal static void UpdateClassSchedule(String updatefield, Klasi klasi,
String new_id, string teacher_id_2 = null)

    {

       if (klasi.update_schedule_id !="0")

       {
```

```
            if( teacher_id_2 != null)

            MainPage.GetActiveSyDBConnection().Execute("UPDATE schedule
SET "+ updatefield + " = ?, teacher_id_2=? Where Id = ?", new_id, teacher_id_2,
klasi.update_schedule_id);

            else

            MainPage.GetActiveSyDBConnection().Execute("UPDATE schedule
SET "+ updatefield + " = ? Where Id = ?", new_id, klasi.update_schedule_id);

         }

         else

         {

            MainPage.GetActiveSyDBConnection(). Execute("insert into schedule ("
+ updatefield + ", class_period, section_id, period  ) values (?,?,?,?) ", new_id,
klasi.period_id, klasi.sectionComplete.id, klasi.period);

         }

      }

      internal static void UpdateSchedule(String updatefield,  String
new_id,string update_schedule_id)

      {
```

```
            MainPage.GetActiveSyDBConnection().Execute("UPDATE schedule
SET "+ updatefield + " = ? Where Id = ?", new_id,  update_schedule_id);



        }
        internal static void ExecuteSQL(string SQL, string text, string v2)
        {
            MainPage.GetDBConnection().Execute(SQL,text, v2);
        }
        internal static int SaveSection(Section section)
        {
            int num = -1;
            if (section.id == 0)
            {
                MainPage.GetActiveSyDBConnection().Insert(section);
                    string sql = @"select last_insert_rowid()";
                num = MainPage.GetActiveSyDBConnection().ExecuteScalar<int>(sql);
            }
```

```
        else
        {
            MainPage.GetActiveSyDBConnection().Update(section);
        }
        return num;
    }

    internal static List<Schedule> GetSchedule()
    {
        List<Schedule> listp = new List<Schedule>();
        List<Schedule> list = new List<Schedule>();
        list =
MainPage.GetDBConnection().Table<Schedule>().ToList<Schedule>();
        foreach (Schedule t in list)
        {
            // BitmapImage img = GetBitmapIgAsync( t.imgArray ).Result;
            listp.Add(new Schedule()
```

```
            {

                id = t.id

            });

        }

        return list;

    }



    internal static  Schedule  GetOneSchedule(string id)

    {

      return  MainPage.GetActiveSyDBConnection().Query<Schedule>("select *
from Schedule where id =?", id).FirstOrDefault();

    }

          nternal static List<ScheduleWithData>
GetClassSchedule(SectionComplete sectionComplete)

    {

        List<ClassPeriod> classPeriods = GetClassPeriod(sectionComplete.level);
```

```
        List<Schedule> ischedules =
MainPage.GetActiveSyDBConnection().Query<Schedule>("select * from
Schedule where section_id =?", sectionComplete.id);

        List<ScheduleWithData> schedules = new List<ScheduleWithData>();

        foreach (ClassPeriod c in classPeriods)

        {

           Boolean found = false;

           foreach (Schedule s in ischedules)

           {

              if (c.id.ToString() == s.class_period)

              {

                 found = true;

                 schedules.Add(new ScheduleWithData(c, s));

              }



           }

           if(!found)

                 schedules.Add(new ScheduleWithData(c));
```

```
        }

        return schedules;

    }

    internal static List<SubjectLoad> GetTeacherSchedule(String teacherid)

    {



        List<Schedule> ischedules =
MainPage.GetActiveSyDBConnection().Query<Schedule>("select * from
Schedule where teacher_id =?", teacherid);

        List<SubjectLoad> s = new List<SubjectLoad>();

        foreach (Schedule item in ischedules)

        {

            ClassPeriod classPeriod =
MainPage.GetDBConnection().Query<ClassPeriod>("select * from ClassPeriod
where id=?", item.class_period).FirstOrDefault();

            String nRoom = "Not set";

            if (item.room_id != null) nRoom =
DBManager.getRoom(item.room_id).room_name;
```

```
            Section section = DBManager.GetSection(item.section_id);
            Subject oSubject = DBManager.GetSubject(item.subject_id);
            String subjectText = "Not set";
            if(oSubject != null)
            {
               subjectText = oSubject.subject;
            }
            s.Add(new SubjectLoad() {
               Duration = classPeriod.minutes,
               Time = classPeriod.start_time + "-" + classPeriod.end_time,
               Period  = classPeriod.period,
               Room = nRoom,
               Subject = subjectText,
               GradeAndSection = "Grade " + section.grade_level + "-" +
section.section
            });
         }
```

```
            return s.OrderBy(p => p.Period).ToList(); ;
        }
    private static  List<ClassPeriod> GetClassPeriod(String level)
        {
            List<ClassPeriodWithInfo> listp = new List<ClassPeriodWithInfo>();
            List<ClassPeriod> list = new List<ClassPeriod>();
            list = fconn.Query<ClassPeriod>("select * from ClassPeriod where level
=?", level);
            foreach (ClassPeriod t in list)
            {
                listp.Add(new ClassPeriodWithInfo( t ) );
            }
            return list;
        }
        internal static List<RoomView> GetRooms()
        {
            List<RoomView> listp = new List<RoomView>();
```

```
    List<Room> list = new List<Room>();
    list = MainPage.GetDBConnection().Table<Room>().ToList<Room>();
    foreach (Room t in list)
    {
       listp.Add(new RoomView()
       {
          id = t.id,
          room_name = t.room_name,
          room_description = t.room_description,
          picture = t.picture,
          photo = MiscUtils.BytesToImage(t.picture)
       });
    }
    return listp;
  }
internal static void AddTeacher(Teacher nTeacher)
 {
```

```
    if( nTeacher.id == 0)
    {
       MainPage.GetDBConnection().Insert(nTeacher);
    }
    else
    {
       MainPage.GetDBConnection().Update(nTeacher);
    }

 }
public static List<SubjectRow> GetSubjects()
 {
    List<SubjectRow> listp = new List<SubjectRow>();
    List<Subject> list = new List<Subject>();
    list = fconn.Table<Subject>().ToList<Subject>();
    int ctr = 1;
    foreach (Subject t in list)
```

```
{
   listp.Add(new SubjectRow()
   {
      id       = t.id,
      ctr     = ctr,
      grade   = t.grade,
      subject   = t.subject,
      no_of_minutes   = t.no_of_minutes,
      grade_level = "Grade " +t.grade,
      color = t.color,
       is_multi_sched = t.is_multi_sched,
       first_schedule = t.first_schedule,
       second_schedule = t.second_schedule
   });
   ctr++;
}
return listp;
```

```
        }
        internal static Teacher GetTeacher(string id)
        {
            return MainPage.GetDBConnection().Query<Teacher>("select * from
teacher where id =?", id).FirstOrDefault();
        }
    public static Subject getSubjectRow(String id)
        {
            return MainPage.GetDBConnection().Query<Subject>("select * from
subject where id =?", id).FirstOrDefault();
        }
    internal static void SaveSubject(Subject sSubject)
        {
            if (sSubject.id == 0)
            {
                MainPage.GetDBConnection().Insert(sSubject);
            }
            else
```

```
        {
            MainPage.GetDBConnection().Update(sSubject);
        }
    }
internal static void SaveRoom(Room room)
    {
        if (room.id == 0)
        {
            MainPage.GetDBConnection().Insert(room);
        }
        else
        {
            MainPage.GetDBConnection().Update(room);
        }
    }

    internal static Room getRoom(string id)
```

```
        {
            return MainPage.GetDBConnection().Query<Room>("select * from room
where id =?", id).FirstOrDefault();
        }
*//Has Assignment//*
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Windows.UI;
using Windows.UI.Xaml.Data;
using Windows.UI.Xaml.Media;
namespace HamburgerMenuDemo.Models
{
    public class HasAssignment : IValueConverter
    {
```

```
        public object Convert(object value, Type targetType, object parameter, string
culture)
        {
          //  if (value == null) return new SolidColorBrush(Colors.Gray);
            return new SolidColorBrush((value.ToString() != "Not set" ? Colors.Green :
Colors.Gray));
        }
        public object ConvertBack(object value, Type targetType, object parameter,
string culture)
        {
            throw new NotImplementedException();
        }
    }
}
*//hide nonregular period//*
using System;
using System.Collections.Generic;
using System.Linq;
```

```
using System.Text;

using System.Threading.Tasks;

using Windows.UI.Xaml;

using Windows.UI.Xaml.Data;

namespace HamburgerMenuDemo.Models

{

   class HideNonRegularPeriod  : IValueConverter

      {

         public object Convert(object value, Type targetType, object parameter,

string culture)

         {

         return  ((bool)value ? Visibility.Visible : Visibility.Collapsed);

          //   return   Visibility.Visible  ;

         }

         public object ConvertBack(object value, Type targetType, object parameter,

string culture)

         {

            throw new NotImplementedException();
```

```
            }
    }
}
*//if class is available//*
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Windows.UI.Xaml.Data;
namespace HamburgerMenuDemo.Models
{
    class IsAvailable : IValueConverter
    {
        public object Convert(object value, Type targetType, object parameter, string
culture)
        {
```

```
            return ((bool)value ? "Select" : "Conflict");
        }
        public object ConvertBack(object value, Type targetType, object parameter,
string culture)
        {
            throw new NotImplementedException();
        }
    }
}
*//class//*
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace HamburgerMenuDemo.Models
{
```

```
public class Klasi
{
    public SectionComplete sectionComplete;
    public List<ScheduleWithData> classSchedule;
    public String update_schedule_id = null;
    public String period_id;
    public String period;
    public bool IsMultiSchedule = false;
    public EditMode editMode;
    internal string fieldToUpdate;

    public enum EditMode
    {
        Subject,
        Teacher,
        Room
    }
```

```
        public Klasi(SectionComplete sectionComplete, List<ScheduleWithData>
classSchedule, EditMode editMode, ScheduleWithData.PeriodAndUpdateId data)
        {
            this.sectionComplete = sectionComplete;
            this.classSchedule = classSchedule;
            this.editMode = editMode;
        //  if(data.update_id !=null)
            this.update_schedule_id = data.update_id.ToString();
            this.period_id = data.Periodid.ToString();
            this.period = data.period.ToString();
            this.IsMultiSchedule = data.IsMultiScheduleSubject;


        }
    }
}
*//about page//
using System;
```

```
using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Runtime.InteropServices.WindowsRuntime;

using Windows.Foundation;

using Windows.Foundation.Collections;

using Windows.UI.Xaml;

using Windows.UI.Xaml.Controls;

using Windows.UI.Xaml.Controls.Primitives;

using Windows.UI.Xaml.Data;

using Windows.UI.Xaml.Input;

using Windows.UI.Xaml.Media;

using Windows.UI.Xaml.Navigation;

// The Blank Page item template is documented at

https://go.microsoft.com/fwlink/?LinkId=234238

namespace HamburgerMenuDemo

{
```

```
    /// <summary>
    /// An empty page that can be used on its own or navigated to within a
Frame.
    /// </summary>
    public sealed partial class AboutPage : Page
    {
        public AboutPage()
        {
            this.InitializeComponent();
        }
    }
}
*//account setting set up//
using ClassScheduleOrganizer.Models;
using HamburgerMenuDemo.Models;
using HamburgerMenuDemo.Pages;
using System;
```

```
using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Runtime.InteropServices.WindowsRuntime;

using Windows.Foundation;

using Windows.Foundation.Collections;

using Windows.UI.Xaml;

using Windows.UI.Xaml.Controls;

using Windows.UI.Xaml.Controls.Primitives;

using Windows.UI.Xaml.Data;

using Windows.UI.Xaml.Input;

using Windows.UI.Xaml.Media;

using Windows.UI.Xaml.Navigation;

// The Blank Page item template is documented at
https://go.microsoft.com/fwlink/?LinkId=234238

namespace ClassScheduleOrganizer.Pages

{
```

```
    /// <summary>
    /// An empty page that can be used on its own or navigated to within a
Frame.
    /// </summary>
    public sealed partial class AccountSetUpPage : Page
    {
        public AccountSetUpPage()
        {
            this.InitializeComponent();
            warningBox.Visibility = Visibility.Visible;
        }


        private void Button_Click(object sender, RoutedEventArgs e)
        {


            bool okSave = true;
            warningBox.Visibility = Visibility.Collapsed;
```

```
okSave &= MiscUtils.isNotEmpty(txtName);
okSave &= MiscUtils.isNotEmpty(txtUserName);
okSave &= MiscUtils.isNotEmpty(txtPassword);
if (!okSave)
{
   warningBox.Visibility = Visibility.Visible;
}
else
{
   User user = new User()
   {
      Name = txtName.Text,
      UserName = txtUserName.Text,
      Password = txtPassword.Password
   };
   LogInPage.getAccountConn().Insert(user);
   MiscUtils.MsgBoxAsync("Saved!");
```

```
            AccountSetUpFrame.Navigate(typeof(LogInPage));
        }
    }
  }
}
*//add subject//*
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Runtime.InteropServices.WindowsRuntime;
using Windows.Foundation;
using Windows.Foundation.Collections;
using Windows.UI.Xaml;
using Windows.UI.Xaml.Controls;
using Windows.UI.Xaml.Controls.Primitives;
using Windows.UI.Xaml.Data;
```

```
using Windows.UI.Xaml.Input;
using Windows.UI.Xaml.Media;
using Windows.UI.Xaml.Navigation;
using HamburgerMenuDemo.Models;
using ExpertScheduling;
// The Content Dialog item template is documented at
https://go.microsoft.com/fwlink/?LinkId=234238
namespace HamburgerMenuDemo.Pages
{
   public sealed partial class AddSubjectContentDialog : ContentDialog
   {
      Klasi klasi;
      public AddSubjectContentDialog(Klasi k)
      {
         this.InitializeComponent();
         klasi = k as Klasi;
```

```
            PrepareRequest(klasi.sectionComplete);
        }
        bool do_override = false;
        string _update_id;
    public AddSubjectContentDialog(Klasi k, string update_field, string
new_value, string update_id )
        {
            this.InitializeComponent();
            klasi = k as Klasi;
            do_override = true;
            _update_id = update_id;
            PrepareRequest(klasi.sectionComplete);
        }
        public void PrepareRequest(SectionComplete sectionComplete)
        {
            if (klasi.classSchedule == null)
            {
```

```
            klasi.classSchedule =
DBManager.GetClassSchedule(klasi.sectionComplete);
        }
        List<Subject> subjects =
DBManager.GetClassSubjects(sectionComplete.level);
        List<SubjectRow> iListSubject = new List<SubjectRow>();
        foreach (var subject in subjects)
        {
            ScheduleWithData iObj = klasi.classSchedule.FirstOrDefault(s =>
s.Subject  == subject.subject);
             if (iObj == null)
                        {
                            iListSubject.Add( new SubjectRow() { id= subject.id, color =
subject.color, grade = subject.grade, subject = subject.subject, no_of_minutes =
subject.no_of_minutes });
                        }
        }
```

```
            subject_listView.ItemsSource = iListSubject.OrderBy(sbj =>
sbj.subject).ToList(); ;
        }
        private void Select_Button_Click(object sender, RoutedEventArgs e)
        {
            var btn = sender as Button;
            String id = btn.DataContext.ToString();


            DBManager.UpdateSchedule("subject_id", id, _update_id);


            //DBManager.UpdateClassSchedule( klasi.fieldToUpdate, klasi, id);
            this.Hide();
            //MainPage.refreshActiveFrame();
        }
        public int Result { get; private set; }
```

```
        private void ContentDialog_CloseButtonClick(ContentDialog sender,
ContentDialogButtonClickEventArgs args)
        {
            // User clicked Cancel, ESC, or the system back button.
            this.Result = -1;
        }
        private void Button_Click(object sender, RoutedEventArgs e)
        {
            this.Hide();
        }
        private void Button_Click_1(object sender, RoutedEventArgs e)
        {
        //    var btn = sender as Button;
            String id = "0";
           // DBManager.UpdateClassSchedule(klasi.fieldToUpdate, klasi, id);
            DBManager.UpdateSchedule("subject_id", id, _update_id);
            this.Hide();
```

```
            MainPage.refreshActiveFrame();

        }

    }

}

*//print manager//*

using ClassScheduleOrganizer.Pages;

using HamburgerMenuDemo.Models;

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Runtime.InteropServices.WindowsRuntime;

using Windows.Foundation;

using Windows.Foundation.Collections;

using Windows.Graphics.Printing;

using Windows.Graphics.Printing.OptionDetails;

using Windows.UI.Xaml;
```

```
using Windows.UI.Xaml.Controls;

using Windows.UI.Xaml.Controls.Primitives;

using Windows.UI.Xaml.Data;

using Windows.UI.Xaml.Input;

using Windows.UI.Xaml.Media;

using Windows.UI.Xaml.Navigation;

using Windows.UI.Xaml.Printing;

using ExpertScheduling;

// The Blank Page item template is documented at
https://go.microsoft.com/fwlink/?LinkId=234238

namespace HamburgerMenuDemo.Pages

{

   /// <summary>

   /// An empty page that can be used on its own or navigated to within a
Frame.

   /// </summary>

   public sealed partial class AkonPrintManager : Page

   {
```

```
public AkonPrintManager()
{
    this.InitializeComponent();
    try
    {
    printmgr.PrintTaskRequested += Printmgr_PrintTaskRequested;


    }
    catch (Exception e)
    {
        //throw;
        String ee = e.Message;
    }
    AutoSuggest_ItemsSource = DBManager.GetSections();
    AutoSuggestSectionGradeList.ItemsSource = AutoSuggest_ItemsSource;
}
protected override void OnNavigatingFrom(NavigatingCancelEventArgs e)
```

```
{
    printmgr.PrintTaskRequested -= Printmgr_PrintTaskRequested;
    if (printDoc != null)
    {
        printDoc.GetPreviewPage -= OnGetPreviewPage;
        printDoc.Paginate -= PrintDic_Paginate;
        printDoc.AddPages -= PrintDic_AddPages;
    }
}
SectionComplete sectionComplete;
Page printTeachingLoadTemplate;
private string titleText;
protected override void OnNavigatedTo(NavigationEventArgs e)
{
    base.OnNavigatedTo(e);
    if(e.Parameter != null)
    {
```

```
            sectionComplete = (SectionComplete)e.Parameter;


printPreviewFrame.Navigate(typeof(PrintClassHomeProgramTemplate),
sectionComplete);
          }



      }
  PrintManager printmgr = PrintManager.GetForCurrentView();
      PrintDocument printDoc = null;
      PrintTask task = null;
      private List<SectionComplete> AutoSuggest_ItemsSource;
      private void Printmgr_PrintTaskRequested(PrintManager sender,
PrintTaskRequestedEventArgs args)
      {
          var deferral = args.Request.GetDeferral();
          task = args.Request.CreatePrintTask( titleText ,
OnPrintTaskSourceRequrested);
```

```
        PrintTaskOptionDetails printDetailedOptions =
PrintTaskOptionDetails.GetFromPrintTaskOptions(task.Options);
        deferral.Complete();
    }
    private async void
OnPrintTaskSourceRequrested(PrintTaskSourceRequestedArgs args)
    {
        try
        {
            var def = args.GetDeferral();
            await
Dispatcher.RunAsync(Windows.UI.Core.CoreDispatcherPriority.Normal,
                () =>
                {
                    args.SetSource(printDoc?.DocumentSource);
                });
            def.Complete();
```

```
    }
    catch (Exception e)
    {
        String ee = e.Message;
    }


}
internal void setTargetPrint(Page allSchedulesPage)
{
   // allSchedulesPage.initTimeAsync();
    printTeachingLoadTemplate = allSchedulesPage;
    titleText = "xxxxxxxx";
    if (MainPage.getTimeSet() == "1")
    {
        titleText =   "Grade 1 to 3";
    }
```

```
            if (MainPage.getTimeSet() == "4")
            {
                titleText =   "Grade 4 to 6";
            }


        }
        private void PrintDic_AddPages(object sender, AddPagesEventArgs e)
        {
            printDoc.AddPage( printTeachingLoadTemplate );
            printDoc.AddPagesComplete();
        }
        private void PrintDic_Paginate(object sender, PaginateEventArgs e)
        {
            PrintTaskOptions opt = task.Options;
            PrintTaskOptionDetails printDetailedOptions =
PrintTaskOptionDetails.GetFromPrintTaskOptions(e.PrintTaskOptions);
            printDoc.SetPreviewPageCount(1, PreviewPageCountType.Final);
```

```
}

private void OnGetPreviewPage(object sender, GetPreviewPageEventArgs e)

{

   try

   {

      Page page = printTeachingLoadTemplate;

      Grid printableArea = (Grid)page.FindName("PrintArea");

      printDoc.SetPreviewPage(e.PageNumber, printableArea);

   }

   catch (Exception)

   {



   }



}
```

```
        private void PreparePrintContent(Page pageToPrint)
        {
            var canvas = (Canvas)this.FindName("MyCanvas");
            canvas.Children.Clear();
            canvas.Children.Add(pageToPrint);
        }
        private async void printClasshome_program_Click(object sender,
RoutedEventArgs e)
        {
            if (sectionComplete != null)
            {
                var d = new PrintClassHomeProgramTemplate();
                d.initSomething(sectionComplete);
                printTeachingLoadTemplate = d;
                titleText = sectionComplete.grade_and_section;
                preparePrintDoc();
            }
```

```
    }

    private void AutoSuggestBox_TextChanged(AutoSuggestBox sender,
AutoSuggestBoxTextChangedEventArgs args)

    {

        try

        {

            var tmp = AutoSuggest_ItemsSource.Where(c =>
c.grade_and_section.ToLower().Contains(AutoSuggestSectionGradeList.Text.To
Lower()));

            AutoSuggestSectionGradeList.ItemsSource = tmp;

        }

        catch (Exception e)

        {

            String ee = e.Message;

        }

    }
```

```
    private void AutoSuggestBox_QuerySubmitted(AutoSuggestBox sender,
AutoSuggestBoxQuerySubmittedEventArgs args)
    {
        sectionComplete = args.ChosenSuggestion as SectionComplete;
        if (sectionComplete != null)
        printPreviewFrame.Navigate(typeof(PrintClassHomeProgramTemplate),
sectionComplete);
        //preparePrintDoc();
    }
    private void AutoSuggestBox_SuggestionChosen(AutoSuggestBox sender,
AutoSuggestBoxSuggestionChosenEventArgs args)
    {
        sectionComplete = args.SelectedItem as SectionComplete;
        if (sectionComplete != null)

printPreviewFrame.Navigate(typeof(PrintClassHomeProgramTemplate),
sectionComplete);
       // preparePrintDoc();
```

```
}

public async void preparePrintDoc()

{


   if (printDoc != null)

   {

      printDoc.GetPreviewPage -= OnGetPreviewPage;

      printDoc.Paginate -= PrintDic_Paginate;

      printDoc.AddPages -= PrintDic_AddPages;

   }

   this.printDoc = new PrintDocument();

   printDoc.GetPreviewPage += OnGetPreviewPage;

   printDoc.Paginate += PrintDic_Paginate;

   printDoc.AddPages += PrintDic_AddPages;


   PreparePrintContent(printTeachingLoadTemplate);
```

```
            bool showPrint = await PrintManager.ShowPrintUIAsync();

        }

    }

}

*//all schedule//*

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Runtime.InteropServices.WindowsRuntime;

using Windows.Foundation;

using Windows.Foundation.Collections;

using Windows.UI.Xaml;

using Windows.UI.Xaml.Controls;

using Windows.UI.Xaml.Controls.Primitives;

using Windows.UI.Xaml.Data;

using Windows.UI.Xaml.Input;
```

```
using Windows.UI.Xaml.Media;

using Windows.UI.Xaml.Navigation;

using HamburgerMenuDemo.Models;

using System.Threading.Tasks;

using Windows.Graphics.Printing;

using Windows.UI.Xaml.Printing;

using Windows.Graphics.Printing.OptionDetails;

using ClassScheduleOrganizer.Pages;

using Syncfusion.XlsIO;

using Windows.Storage.Pickers;

using Windows.Storage;

using Windows.UI;

using System.Reflection;

using RFloraClassScheduler.Pages;

using ExpertScheduling;

// The Blank Page item template is documented at

https://go.microsoft.com/fwlink/?LinkId=234238
```

```
namespace HamburgerMenuDemo.Pages
{
    /// <summary>
    /// An empty page that can be used on its own or navigated to within a
Frame.
    /// </summary>
    public sealed partial class AllSchedulesPage : Page
    {
        private List<ClassPeriodWithInfo> _ItemsSource;
        public AllSchedulesPage()
        {
            this.InitializeComponent();


            initTimeAsync();
```

```
    }

    public static List<String> SubjectToAvoid = new List<string> { "Remedial
Instruction","Flag Ceremony", "Supervised Recess", "Noon Break" };



    bool codeTriggered = false;

    Double height;

    public async Task initTimeAsync()

    {

       height   = Window.Current.Bounds.Height;

       height = height - 20;

       height = height / 14;

       codeTriggered = true;

       if (MainPage.getTimeSet() == "1")

       {

          comboYear.SelectedIndex = 0;

          txtTimeToDisplay.Text = "Grade 1 to 3";

       }
```

```
    if ( MainPage.getTimeSet()=="4")

    {

        comboYear.SelectedIndex = 1;

        txtTimeToDisplay.Text = "Grade 4 to 6";

    }

    codeTriggered = false;



    loadingBorder.Visibility = Visibility.Visible;

    gridList.Visibility = Visibility.Collapsed;



     await Task.Delay(500);

    await ComputeNextMove();

}



private async Task ComputeNextMove()

{
```

```
        var tset = MainPage.getTimeSet();
        _ItemsSource = DBManager.getClassPeriodSample( tset );

        // time_list_listView.ItemsSource = _ItemsSource;

        ActiveGradeText.Height = height;
        foreach (var item in _ItemsSource)
        {
            RowDefinition rowDefinition = new RowDefinition();
            rowDefinition.Height = new GridLength(height);
            time_list_listView.RowDefinitions.Add(rowDefinition);
            TimeGridControl timeGridControl = new TimeGridControl(item,
height);
            time_list_listView.Children.Add(timeGridControl);
            Grid.SetRow(timeGridControl,
time_list_listView.RowDefinitions.Count - 1);
```

```
        }


        var whe = MainPage.getWhere();
        var collection = DBManager.GetSections("SELECT * FROM SECTION
WHERE " + whe);
        st.Children.Clear();
        foreach (var item in collection)
        {
            st.Children.Add(new SectionSchedulePage(item, height));
        }
        loadingBorder.Visibility = Visibility.Collapsed;
        gridList.Visibility = Visibility.Visible;


    }
    string where,TimeSet;


    private async void Button_Click(object sender, RoutedEventArgs e)
```

```
{
    /*
    //PreparePrintContent(new emptyPage());
    printMasterScheduleTemplate = new PrintMasterScheduleTemplate();
    await  printMasterScheduleTemplate.initContents();


    preparePrintDoc();
    */
    try
    {
       ExportToExcelAsync();
    }
    catch (Exception ex)
    {
       MiscUtils.MsgBoxAsync(ex.Message);
    }
```

```
        private void ComboBox_SelectionChanged(object sender,
SelectionChangedEventArgs e)
        {
            if(!codeTriggered)
            {
                var i = sender as ComboBox;
                var item = i.SelectedItem as ComboBoxItem;
                var toView = item.Content.ToString();
                if (toView.Equals("Grade 4 to 6"))
                {
                    TimeSet = "4";
                    where = " grade_level >= 4 and grade_level <=6; ";
                }
                else
                {
                    TimeSet = "1";
                    where = " grade_level >= 1 and grade_level <=3; ";
```

```
            }

            MainPage.setTimeSet(TimeSet);

            MainPage.setWhere(where);

            initTimeAsync();

        }

    }

 string[] letters = { "A", "B", "C", "D", "E", "F", "G", "H", "I", "J", "K", "L", "M", "N",
"O", "P", "Q", "R", "S", "T", "U", "V", "W", "X", "Y", "Z", "AA", "AB", "AC", "AD",
"AE", "AF", "AG", "AH", "AI", "AJ", "AK", "AL", "AM", "AN", "AO", "AP", "AQ",
"AR", "AS", "AT", "AU", "AV", "AW", "AX", "AY", "AZ" };

    async Task ExportToExcelAsync()

    {

        //Create an instance of ExcelEngine.

        using (ExcelEngine excelEngine = new ExcelEngine())

        {

            IApplication application = excelEngine.Excel;



            application.DefaultVersion = ExcelVersion.Excel2016;
```

```
            //Create a workbook with a worksheet
            IWorkbook workbook = application.Workbooks.Create(1);


            //Access first worksheet from the workbook instance.
            IWorksheet worksheet = workbook.Worksheets[0];



            Assembly executingAssembly =
typeof(MainPage).GetTypeInfo().Assembly;
            Stream inputStream =
executingAssembly.GetManifestResourceStream("SplashScreen.scale-200.png");
            String fileName = "Master Schedule School Year " +
MainPage.getActiveSchoolyear().SchoolYear;


           string rowindex = "";
            string text = "";
```

```
        //    Grid GridArea = new Grid();

            var tset = MainPage.getTimeSet();

            var _ItemsSource = DBManager.getClassPeriodSample(tset);

            var collection = DBManager.GetSections("SELECT * FROM SECTION
WHERE 1");

            //    PrintArea.Children.Clear();



            worksheet.Range["A2"].CellStyle.Font.Bold = true;

            worksheet.Range["A2"].CellStyle.HorizontalAlignment =
ExcelHAlign.HAlignCenter;

            worksheet.Range["A2"].CellStyle.VerticalAlignment =
ExcelVAlign.VAlignCenter;




            // Time text

            //worksheet.Range["A2"].ColumnWidth = 20;

            //worksheet.Range["A2"].Text = "Time";

            //worksheet.Range["A2:A3"].Merge();
```

```
            //worksheet.Range["A2"].CellStyle.Font.Bold = true;

            //worksheet.Range["A2"].CellStyle.HorizontalAlignment =
ExcelHAlign.HAlignCenter;

            //worksheet.Range["A2"].CellStyle.VerticalAlignment =
ExcelVAlign.VAlignCenter;



            int row_start = 5;

            int column_start = 0;

            int _rowHeight = 20;



            row_start = 5;



            _ItemsSource = DBManager.getClassPeriodSample("1");

            AddColumnHeader(worksheet, column_start, row_start, _ItemsSource,
"Grade 1 - 3");
```

```
            column_start++;



            string current_grade = "1";

            bool grade_4_to_6_has_colunm = false;

            foreach (var _section in collection)

            {

                if( int.Parse(_section.level) >= 4)

                {

                    if(!grade_4_to_6_has_colunm)

                    {

                        row_start = 5;

                       column_start++;

                        _ItemsSource = DBManager.getClassPeriodSample("4");

                        AddColumnHeader(worksheet, column_start, row_start,
_ItemsSource, "Grade 4 - 6");

                        grade_4_to_6_has_colunm = true;

                    column_start++;
```

```
                }

                }


            column_start++;


            row_start = 2; // Grade and Section

            rowindex = letters[column_start] + (row_start);

            worksheet.Range[rowindex].Text = _section.grade_and_section;

            worksheet.Range[rowindex].RowHeight = _rowHeight;

            worksheet.Range[rowindex].CellStyle.Font.Bold = true;

            worksheet.Range[rowindex].CellStyle.HorizontalAlignment =
ExcelHAlign.HAlignCenter;

            worksheet.Range[rowindex].CellStyle.VerticalAlignment =
ExcelVAlign.VAlignCenter;

            worksheet.Range[rowindex].CellStyle.Color = Colors.Beige;

            worksheet.Range[rowindex].CellStyle.Font.Color=
ExcelKnownColors.Grey_80_percent ;
```

```
            row_start++; // Class Adviser

            rowindex = letters[column_start] + (row_start);

            worksheet.Range[rowindex].RowHeight = _rowHeight;

            worksheet.Range[rowindex].Text = _section.Adviser;

            worksheet.Range[rowindex].CellStyle.HorizontalAlignment =
ExcelHAlign.HAlignCenter;

            worksheet.Range[rowindex].CellStyle.VerticalAlignment =
ExcelVAlign.VAlignCenter;

            worksheet.Range[rowindex].CellStyle.Color = Colors.Beige;

            worksheet.Range[rowindex].CellStyle.Font.Color =
ExcelKnownColors.Grey_80_percent;



            worksheet.Range[rowindex].ColumnWidth = 40;



            var Schedules = DBManager.GetClassSchedule(_section);
```

```
                Schedules = Schedules.OrderBy(d => d.Period).ToList();

                var sbox = new GridScheduleBox(_section.grade_and_section,
_section.Adviser, new SolidColorBrush(Colors.Goldenrod));


                foreach (ScheduleWithData schedule in Schedules)

                {


                  if(SubjectToAvoid.IndexOf(schedule.Subject) < 0)

                  {

                  row_start++; // Subject

                  rowindex = letters[column_start] + (row_start);

                  worksheet.Range[rowindex].RowHeight = _rowHeight;

                  worksheet.Range[rowindex].Text = schedule.Subject;

                  worksheet.Range[rowindex].CellStyle.HorizontalAlignment =
ExcelHAlign.HAlignCenter;

                  worksheet.Range[rowindex].CellStyle.VerticalAlignment =
ExcelVAlign.VAlignBottom;
```

```
                worksheet.Range[rowindex].CellStyle.Color =
schedule.SubjectColor.Color;

                worksheet.Range[rowindex].CellStyle.Font.Bold = true;

                worksheet.Range[rowindex].CellStyle.Font.Color =
ExcelKnownColors.White;

                row_start++; // Teacher

                rowindex = letters[column_start] + (row_start);

                worksheet.Range[rowindex].RowHeight = _rowHeight;

                worksheet.Range[rowindex].Text = schedule.Teacher;

                worksheet.Range[rowindex].CellStyle.HorizontalAlignment =
ExcelHAlign.HAlignCenter;

                worksheet.Range[rowindex].CellStyle.VerticalAlignment =
ExcelVAlign.VAlignTop;

                worksheet.Range[rowindex].CellStyle.Color =
schedule.SubjectColor.Color;

                worksheet.Range[rowindex].CellStyle.Font.Color =
ExcelKnownColors.White;
```

```
                }
                else
                {
                    row_start++; // Avoid
                    rowindex = letters[column_start] + (row_start);
                    worksheet.Range[rowindex].RowHeight = _rowHeight;
                    worksheet.Range[rowindex].Text = schedule.Subject;
                    worksheet.Range[rowindex].CellStyle.HorizontalAlignment =
ExcelHAlign.HAlignCenter;
                    worksheet.Range[rowindex].CellStyle.VerticalAlignment =
ExcelVAlign.VAlignCenter;
                    worksheet.Range[rowindex].CellStyle.Color = Colors.Gainsboro;
                    worksheet.Range[rowindex].CellStyle.Font.Bold = true;
                    worksheet.Range[rowindex].CellStyle.Font.Color =
ExcelKnownColors.Grey_80_percent;
                    worksheet.Range[rowindex + ":" + letters[column_start] +
(row_start + 1)].Merge();
                    row_start++;
```

```
            }
        }
    }

    // header na
    worksheet.Name = fileName;

    worksheet.Range["A1"].Text = fileName;
    worksheet.Range["A1"].RowHeight = 35;
    worksheet.Range["A1"].CellStyle.Font.Size = 20;
    worksheet.Range["A1"].CellStyle.Font.Color =
ExcelKnownColors.White;
    worksheet.Range["A1"].CellStyle.Color = Colors.DimGray;
    worksheet.Range["A1"].CellStyle.VerticalAlignment =
ExcelVAlign.VAlignCenter;

    worksheet.Range["A1:" + letters[collection.Count + 2] + "1"].Merge();
```

```
            rowindex = letters[0] + (row_start);



            // Save the Workbook
            StorageFile storageFile;
           if
(!(Windows.Foundation.Metadata.ApiInformation.IsTypePresent("Windows.Pho
ne.UI.Input.HardwareButtons")))
            {
               FileSavePicker savePicker = new FileSavePicker();
               savePicker.SuggestedStartLocation = PickerLocationId.Desktop;
               savePicker.SuggestedFileName = fileName;// "Output";
               savePicker.FileTypeChoices.Add("Excel Files", new List<string>()
{ ".xlsx" });
               storageFile = await savePicker.PickSaveFileAsync();
            }
            else
            {
```

```
                StorageFolder local =
Windows.Storage.ApplicationData.Current.LocalFolder;
                storageFile = await local.CreateFileAsync(fileName + ".xlsx",
CreationCollisionOption.ReplaceExisting);
            }
            //Saving the workbook
            await workbook.SaveAsAsync(storageFile);

            // Launch the saved file
            await Windows.System.Launcher.LaunchFileAsync(storageFile);
            MiscUtils.MsgBoxAsync("Done");
        }
    }

    void AddColumnHeader(IWorksheet worksheet, int column_start, int
row_start,List<ClassPeriodWithInfo> _ItemsSource , string timeText)
    {
```

```
        string break_column = letters[column_start] + "4:" + letters[column_start]
+ (((_ItemsSource.Count + 1) * 2) + 1);


        worksheet.Range[letters[column_start] + "4"].ColumnWidth = 10;

        worksheet.Range[letters[column_start] + "4"].Text = "Time Period " +
timeText;

        worksheet.Range[letters[column_start] + "4"].CellStyle.Rotation = 90;

        worksheet.Range[letters[column_start] + "4"].CellStyle.Font.Size = 18;

        worksheet.Range[letters[column_start] + "4"].CellStyle.VerticalAlignment
= ExcelVAlign.VAlignCenter;

        worksheet.Range[letters[column_start] +
"4"].CellStyle.HorizontalAlignment = ExcelHAlign.HAlignCenter;

        worksheet.Range[letters[column_start] + "4"].CellStyle.Color =
Colors.Gainsboro;

        worksheet.Range[letters[column_start] + "4"].CellStyle.Font.Color =
ExcelKnownColors.Grey_80_percent;
```

```
            worksheet.Range[break_column].Merge();

            worksheet.Range[break_column].CellStyle.Color = Colors.Gainsboro;

            worksheet.Range[break_column].CellStyle.Font.Color =
ExcelKnownColors.Grey_80_percent;



            column_start++;

            String rowindex = "";

            worksheet.Range[letters[column_start] + "2"].ColumnWidth = 20;

            worksheet.Range[letters[column_start] + "2"].Text = timeText;

            worksheet.Range[letters[column_start] + "2"].CellStyle.Font.Size = 30;

            worksheet.Range[letters[column_start] + "2"].CellStyle.Font.Bold = true;

            worksheet.Range[letters[column_start] + "2"].CellStyle.VerticalAlignment
= ExcelVAlign.VAlignCenter;

            worksheet.Range[letters[column_start] +
"2"].CellStyle.HorizontalAlignment = ExcelHAlign.HAlignCenter;

            worksheet.Range[letters[column_start] + "2"].CellStyle.Color =
Colors.Gainsboro;
```

```
        worksheet.Range[letters[column_start] + "2"].CellStyle.Font.Color =
ExcelKnownColors.Blue_grey;


        worksheet.Range[letters[column_start-1] + "2:" + letters[column_start] +
"3"].Merge();
        foreach (ClassPeriodWithInfo item in _ItemsSource)
        {


            rowindex = letters[column_start] + (row_start - 1) + ":" +
letters[column_start] + row_start;
            worksheet.Range[rowindex].Text = item.TimeText;


            worksheet.Range[rowindex].Merge();
            if (!item.IsRegularTime)
            {
                worksheet.Range[rowindex].CellStyle.Color = Colors.Gainsboro;
                worksheet.Range[rowindex].CellStyle.Font.Bold = true;
```

```
            worksheet.Range[rowindex].CellStyle.Font.Color =
ExcelKnownColors.Grey_80_percent;
            worksheet.Range[rowindex].CellStyle.VerticalAlignment =
ExcelVAlign.VAlignCenter;
        }
        else
        {
            worksheet.Range[rowindex].CellStyle.Font.Bold = true;
            worksheet.Range[rowindex].CellStyle.HorizontalAlignment =
ExcelHAlign.HAlignCenter;
            worksheet.Range[rowindex].CellStyle.VerticalAlignment =
ExcelVAlign.VAlignCenter;
            worksheet.Range[rowindex].CellStyle.Color = Colors.DimGray;
            worksheet.Range[rowindex].CellStyle.Font.Color =
ExcelKnownColors.White;
        }

        row_start += 2;
```

```
            }


        }


    }

}
```

Appendix B

LETTER TO THE ELEMENTARY PRINCIPAL

Republic of the Philippines
SAMAR COLLEGES, INC.
Catbalogan City

January 19, 2020

EVANGELINE B. MIRANDA, MAT
Elementary Principal
Samar Colleges, Inc.
Catbalogan, City

Madam:

Greetings!

I would like to request permission to conduct an interview and conduct an on-site observation on your current approach on Classroom Management. This is in connection with my research study entitled "Classroom Management in Elementary & Junior High school Department using Expert System." Rest assured that the entrusted data will be handled with extreme confidentiality and will be used only for this study.

Your kind and favorable consideration and preferential attention to this request would be highly appreciated.

Thank you & more power

Very truly yours,

(Sgd)JOHANNA C. FLORA
Researcher

Approved by:

(Sgd)EVANGELINE B. MIRANDA, MAT
Elementary Principal

Appendix C

LETTER TO THE JUNIOR HIGH SCHOOL PRINCIPAL

Republic of the Philippines
SAMAR COLLEGES, INC.
Catbalogan City

January 19, 2020

JACQUELYN B. MONTALES, MA,Ed
Junior High School Principal
Samar Colleges, Inc.
Catbalogan, City

Madam:

Greetings!

I would like to request permission to conduct an interview and conduct an on-site observation on your current approach when it comes to Classroom Management. This is connection with my research study entitled "Classroom Management in Elementary & Junior High school Department using Expert System." Rest assured that the entrusted data will be handled with extreme confidentiality and will be used only for this study.

Your kind and favorable consideration and preferential attention to this request would be highly appreciated.

Thank you & more power.

Very truly yours,

(Sgd)JOHANNA C. FLORA
Researcher

Approved by:

(Sgd) JACQUELYN B. MONTALES, MA,Ed
JHS Principal

Appendix D

LETTER TO THE VICE PRESIDENT FOR ACADEMIC AFFAIRS

Republic of the Philippines
SAMAR COLLEGES, INC.
Catbalogan City

January 19, 2020

PEDRITO G. PADILLA, PhD
Senior Vice-President for
Academic Programs
Samar Colleges, Inc.
Catbalogan, City

Sir:

Greetings!

I would like to request permission to conduct an interview and conduct an on-site observation on your current approach related to classroom management. This is in connection with my research study entitled Classroom Management in Elementary & Junior High school Department using Expert System. Rest assured that the entrusted data will be handled with extreme confidentiality and will be used only for this study.

Your kind and favorable consideration and preferential attention to this request would be highly appreciated.

Thank you & more power

Very truly yours,

(Sgd)JOHANNA C. FLORA
Researcher

Approved by:

(Sgd) PEDRITO G. PADILLA, Ph.D.
Senior Vice-President for Academic Programs

Appendix E

LETTER TO THE VICE PRESIDENT FOR BASIC EDUCATION

Republic of the Philippines
SAMAR COLLEGES, INC.
Catbalogan City

January 19, 2020

LETICIA GUIRRA, PhD
Vice-President for
Basic Education
Samar Colleges, Inc.
Catbalogan, City

Madam:

Greetings!

I would like to request permission to conduct an interview and conduct an on-site observation on your current approach when it comes to classroom management. This is in connection with my research study entitled Classroom Management in Elementary & Junior High school Department using Expert System. Rest assured that the entrusted data will be handled with extreme confidentiality and will be used only for this study.

Your kind and favorable consideration and preferential attention to this request would be highly appreciated.

Thank you & more power

Very truly yours,

(Sgd)JOHANNA C. FLORA
Researcher

Approved by:

(Sgd) LETICIA GUERRA, Ph.D.
Vice-President for Basic Education

Appendix F

LETTER TO THE RESPONDENTS (END-USERS)

Date: _____

Dear Ma'am/Sir,

Greetings!

The undersigned is doctorate student taking up Master of Science in Information Technology. She is currently working on his dissertation titled **Elementary and Secondary Classroom Management Using Expert System.**

In connection to the said research undertaking, the undersigned would like to ask your available time as the end user of the developed system so he can facilitate the beta-testing as output of his study, there will be initial orientation of the said system at your most convenient time just let him know.

She also takes this opportunity to ask permission to accommodate her in your respective department on the months of February to March as her data collection period. Rest assured that all the data provided by you will be treated with utmost confidentiality, and after having used the data, it will be disposed properly.

Looking forward to grant this request. Thank you so much.

Respectfully yours,

(Sgd.) JOHANNA C. FLORA
MSIT, Candidate Student

Noted:

(Sgd.) ESTEBAN A. MALINDOG, JR., Ph.D.
Dean, Graduate School

Appendix G

SYSTEM EVALUATION QUESTIONNAIRE

Name (Optional): _____ Date: _____

Direction: Please check (√) the column that corresponds to your rating on the developed system as to several quality system attribute indicators. Read each item carefully and check the most appropriate rating using the 5-point Likert scale as follow:

5	Excellent	(E)
4	Good	(G)
3	Fair	(F)
2	Poor	(P)
1	Very poor	(V)

	System Quality Attribute Indicators	**5 (E)**	**4 (G)**	**3 (F)**	**2 (P)**	**1 (V)**
FUNCTIONALITY						
1	The proposed system has available all function required for its execution.					
2	The proposed system is precise in its results.					
3	The proposed system interacts with					

	specified modules					
4	The proposed system complies with standards, laws, etc.					
5	The proposed system has secured access through password					

System Quality Attribute Indicators		5 (E)	4 (G)	3 (F)	2 (P)	1 (V)
RELIABILITY						
1	The proposed system reacts appropriately when failure occurred					
2	The proposed system informs user concerning invalid data entry					
3	The proposed system is capable of recovering data in the event of failure					

System Quality Attribute Indicators		5 (E)	4 (G)	3 (F)	2 (P)	1 (V)
USABILITY						
1	It's easy to understand the concept and					

	application					
2	It's easy to learn how to use					
3	It's easy to operate and control					

System Quality Attribute Indicators		5 (E)	4 (G)	3 (F)	2 (P)	1 (V)
EFFICIENCY						
1	The system response time is appropriate					
2	The system execution time is appropriate					
3	The resources used are appropriate					

System Quality Attribute Indicators		5 (E)	4 (G)	3 (F)	2 (P)	1 (V)
MAINTAINABILITY						
1	It's easy to find failure when it occur.					
2	It's easy to modify and adopt					
3	Changes are easy to test					

System Quality Attribute Indicators	5 (E)	4 (G)	3 (F)	2 (P)	1 (V)

PORTABILITY						
1	It's easy to adopt with other environment					
2	It's easy to install in other environment					
3	It is in agreement with portability standard					
4	It's easy to use to replace another program					

Comments and Suggestions (if there's any):

_____

_____

_____

Thank you for your cooperation!

Appendix H

SAMAR STATE UNIVERSITY
Arteche Blvd., Catbalogan City, Philippines 6700
Office of the University President

We Innovate. We Build. We Serve.

JAS-ANZ
Certificate No. AJA18.1089
SSU-OPRES-EA-007
01-FEB-2020 REV. 001

CERTIFICATE OF ETHICS APPROVAL

This is to certify that the Samar State University Institutional Research Ethics Review Committee (IRERC) has reviewed and approved a study entitled:

Title : **Classroom Management using Expert System for Elementary and Secondary Department**
Name of Researcher/s : **Johanna P. Cabalhin**
Reference No : **IRERC EA-0024**
Date of Application : 02/24/2020
Date Reviewed : 02/28/2020

It is hereby mandated that in the implementation of the aforementioned study, the subject researcher shall adhere to International ethical guidelines, national guidelines and all other pertinent requirements prescribed by the SSU-IRERC.

The Researcher can now commence to the data gathering process and the study shall be valid for two (2) years from the date of issuance hereof.

DATE OF ISSUANCE: May 21, 2020 VALID UNTIL: May 21, 2022

RHEAJANE A. ROSALES, D.M.
Director, IRERC

MARILYN D. CARDOSO, Ph.D.
University President

Telephone No. (055) 251 – 2139 | Fax: (055) 543 - 8394 | Website: www.ssu.edu.ph

ETHICAL APPROVAL CERTIFICATE

CURRICULUM VITAE

CURRICULUM VITAE

JOHANNA P. CABALHIN-FLORA

P2 Brgy. Bunu-anan Catabalogan City Samar

(+639283768593)

jflora2018@gmail.com

Personal Vitae

Date of Birth	:	February 19,1986
Place of Birth	:	Catbalogan City
Citizenship	:	Filipino
Marital Status	:	Married

Professional Qualifications

GRADUATE STUDIES

University	:	Samar State University
Address	:	Catbalogan city
Degree	:	Master of Science in Information Technology
Inclusive year:		Summer 2016- Present

TERTIARY

University	:	Eastern Visayas State University
Address	:	Tacloban city
Degree	:	Bachelor of Science in Information Technology
Year Graduated:		2008

Eligibilities

Licensure Exam:	Professional teacher
Date of Conferment:	June 26, 2018
License Number:	1657018

Professional Experience

June 2010 - Present:	College Instructor Samar Colleges, Inc. Catbalogan City

---

Skills

Computer Literate

Languages

English, Filipino, and Waray-waray

LIST OF TABLES

LIST OF TABLES

Table **Page**

LIST OF FIGURES

LIST OF FIGURE