

Smart Timetable Organizer

By
Nem Sopha

Progress Report submitted in partial fulfillment of
the requirements for the
Bachelor of Technology (Hons)
(Business Information System)

MAY 2012

Universiti Teknologi PETRONAS
Bandar Seri Iskandar,
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

SMART TIMETABLE ORGANIZER

by

NEM SOPHA

A project dissertation submitted to the
Computer & Information Systems Programme
Universiti Teknologi PETRONAS
In partial fulfillment of the requirement for the
Bachelor of Technology (Hons)
(Business Information System)

Approved by,

MS. ROZANA BINTI KASBON

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK
MAY 2012

CERTIFICATION OF ORIGINALITY

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

NEM SOPHA

ABSTRACT

Universiti Teknologi PETRONAS (UTP) students are given more flexibility in term of how they want to arrange their classes. Each student tends to have a uniquely different timetable according to their preferred curriculum structure and personal preferences. Smart Timetable Organizer is going to assist students in arranging their own timetable, depending on the courses they wish to enroll for that particular semester. The system will give a trial-and-error platform that can give students all the possible combinations of classes they can have. This platform will be able to reduce the hassle of trying to manually arrange the lectures, labs, and tutorial sessions. It will help to save time and effort of fixing the clash for both students and lecturers, as well as to provide a more reliable way of noting down the timetable rather than practiced in the old times (pen & paper). The project has delivered a working prototype that is currently hosted online for testing and to obtain feedbacks from user. Based on the small scale testing from real potential users, Smart Timetable Organizer has obtained many good feedbacks and satisfaction. Hopefully, with the support from UTP management, the project would be implemented for students' use in the upcoming semester.

ACKNOWLEDGEMENT

First and foremost, I would like to express my gratitude to my supervisor, **Ms. Rozana Kasbon**, for her kind acceptance of me and my topic under her supervision. I really appreciate her advices, guidance, as well as her support for the whole project.

Secondly, I would like to thank Trinh Tuan Duong for his guidance and help with the technical attributes of the project, especially for being patient with me. Similarly, thanks also to my friend Sok Pichleap for her willing help and support whenever I need it. Without the two of you, I could not imagine how harder it would be for me to accomplish this project successfully.

Thirdly, my gratitude goes to my family, seniors and other friends who have always been there for me whenever I needed motivation and encouragement. There is nothing more valuable than the understanding, sharing, and their beliefs in me. I would not stand here today without all of them.

Last but not least, I would like to thank Computer and Information System Department, the lecturers who have given me knowledge and lightened up my path, and to my beloved Universiti Teknologi PETRONAS and PETRONAS for given me all the opportunities and chances to explore and prove myself.

Thank you very much; I hope that this project can benefit some of you – if not all.

LIST OF FIGURES

Figure 2.1: Wise Timetable – Student View	16
Figure 2.2: Mimos Timetabling System: Sample Interface	17
Figure 2.3: UniTime Timetable Screenshots	18
Figure 2.4 : Student’s Personal Timetable (KLIUC)	19
Figure 3.1: Development Methodology - Prototyping	20
Figure 3.2: Milestones for the Project	22
Figure 4.1: UTP Timetabling Business Processes	26
Figure 4.2: Questionnaire response Status by University	29
Figure 4.3: Questionnaire Response by Year of Study	29
Figure 4.4: Questionnaire response to whether the students follow suggested curriculum structure provided by the university	29
Figure 4.5: Questionnaire response on how students prepare their personal timetable	30
Figure 4.6: Questionnaire response on how often students facing clashes	30
Figure 4.7: Questionnaire response on how students deal with the clashes	30
Figure 4.8: Questionnaire response on the comments regarding automated personal timetabling	31
Figure 4.9: Questionnaire response on the preferred type of personal timetable	31
Figure 4.10: As-Is process of how students make their personal timetable	33
Figure 4.11: To-Be process of Smart Timetable Organizer	34
Figure 4.12: System Architecture of Smart Timetable Organizer	35
Figure 4.13: Class Diagram of Smart Timetable Organizer Database	36
Figure 4.14: Home Page	40
Figure 4.15: Result Page	44
Figure 4.16: Result Page: List of Courses Selected & Processed	45
Figure 4.17: Login Page for Administrators to CMS	46

Figure 4.18: View Page in CMS	47
Figure 4.19: Adding Courses in CMS	48
Figure 4.20: Updating Courses in CMS	48
Figure 4.21: Prompt message before deleting a record	48
Figure 4.22: User feedback respondent sort by year of study	48
Figure 4.23: Feedback regarding timetabling functionality	48
Figure 4.24: Feedback regarding user friendliness of the system	48
Figure 4.25: Feedback regarding CMS	48

LIST OF TABLES

Table 3.1: Project Activity List and their Status	21
Table 3.2: Project Gantt Chart throughout the two semester	23
Table 4.1: Summary of Subjects Covered	39
Table 4.2: Pseudo code for Smart Timetable Organizer	44
Table 4.3: Pseudo code for combination algorithm to generate all possible combinations for each group of course selection	45
Table 4.4: Functionality & user experience testing result sheet	54
Table 4.5: User's recommendations	55
Table 4.6: Load & System Performance Test Result Sheet	55

ABBREVIATIONS AND NOMENCLATURES

STO	Smart Timetable Organizer
UTP	Universiti Teknologi PETRONAS
CMS	Content Management System

TABLE OF CONTENT

CERTIFICATE OF APPROVAL	i
CERTIFICATE OF ORIGINALITY	ii
ABSTRACT	iii
ACKNOWLEDGEMENTS	iv
LIST OF FIGURE	v
LIST OF TABLE	vi
ABBREVIATIONS AND NOMENCLATURES	vi
CHAPTER 1: INTRODUCTION	8
1.1 Background of Study	8
1.2 Problem Statement	9
1.3 Objectives and Scope of Study	10
CHAPTER 2: LITERATURE REVIEW.	12
2.1 Introduction to Timetabling	12
2.2 Application and Challenges	12
2.3 Major Issues Encountered in Course Timetabling.	13
2.4 Similar Systems Available in Market.	15
CHAPTER 3: METHODOLOGY	20
3.1 Research & Development Methodology	20
3.2 Project Activities	21
3.3 Key Milestones	23
3.4 Gantt Chart	23
3.5 Tools	25
CHAPTER 4: RESULT AND DISCUSSION	26
4.1 Data Gathering & Analysis	26
4.2 Findings	29
4.3 Prototype Modeling	34
4.3.1 As-Is System & To-Be System	34
4.3.2 System Architecture	37
4.3.3 Database Design	38
4.3.4 Interface Design (User Interface)	42
4.3.5 Content Management System.	48
4.3.6 Testing.	51
CHAPTER 5: CONCLUSION AND FUTURE WORK CONTINUATION	57
5.1 Conclusion.	57
5.2 Future Work Continuation	57
REFERENCES.	59
APPENDICES.	59

CHAPTER 1

INTRODUCTION

1.1 Background of Study

Universities and colleges nowadays always give flexibility to students in term of how they want to arrange their classes, as well as their free time. Of course, those universities and colleges have to come up with the suggested curriculum structure and timetable that students can just pick up and follow, but each student tend to have a unique preferred curriculum structure and criteria, and that is what makes a student's timetable different from another one's. There are many cases to be considered such as diploma holder students, transfer students, students who got exempted for certain subjects, as well as students who wish to customize their curriculum structure to meet their different goals.

This customized structure should have brought so much trouble to both the management and students, but thanks to technology advancement, all of these are made simpler and faster through the usage of automated timetabling system. Timetabling system in this context includes not only the courses' module timetable; it also includes the staff timetable for lecturers and academic staffs, and the allocation of lecture theatres and classrooms. Those other than students' personalized timetables are widely seen implemented in academic institutions all around the globe, but the student's personalized one has not been introduced widely yet.

Smart Timetable Organizer is going to assist students in arranging their own timetable, depending on the courses they wish to enroll for that particular semester. The system will give a trial-and-error platform that can give students alternative timetables in the form of combinations. This platform will be able to reduce the hassle of trying to manually arrange the slots of each lecture, lab, and tutorial sessions.

1.2 Problem Statement

1.2.1 Problem Identification

Based on background study as stated in the previous section and the observation from UTP students' behavior as well as personal experiences every semester for the past four years, it is noticed that there are some common issues regarding the academic timetable as below:

- At the beginning of every semester, course registration and coming up with a final timetable is a hassle and generally consume a lot of students' time. Normally for complicated cases, it takes up to two weeks (until add/drop period is over) that the issues are solved or dropped.
- The case is valid to almost all students, as most of them cannot follow the suggested curriculum structure provided by the university/college due to background differences (diploma holder, transfer, exchange, or exemption) and some of them have their own goals and preferred ways to achieve them. All of these are the main causes of not being able to follow the suggested timetable.
- Timetable clashes are very common. It happens to almost anyone, and it is not easy to fix because most lectures do not have alternative slots, and lecturers have another staff timetable to follow. Plus, if the clashes are realized late, the probability of fixing is not high. (Fixing here means changing one or more of the classes time and venue)
- Some students have to waste the first two weeks to just go and see the situations in classes in order to come up with an arranged timetable. This causes the waste of time and effort, and they might not be able to catch up with the lectures later on.
- Self-written and drawing of timetable tend to come with mistakes and error (human error). The confusion of class venues and time is also very common during the first few weeks of the semester.

1.2.2 Significance of the Project

Smart Timetable Organizer will be able to:

- Automate the personal timetabling process for students
- Provide the timetables in a “combination” basis, so that students are able to see what the possible timetables they can adopt are. Since the system does not link to course registration site, therefore it has no effect at all over the courses registered.
- The system will be available online (web-based). There is no installation required.
- No username or password required to use the system.
- Students can save the timetable into PDF file for further use.

1.3 Objectives and Scope of Study

1.3.1 Objectives

The aim of this project is to:

- Automate personal timetabling process for students based on UTP timetables
 - o Reduce student’s hassle in preparing their personal timetable for each beginning of semester
 - o Implement a trial-and-error combination-based platform that will help students to see many options of timetable that they might be adopting

1.3.2 Scope of Study

The system will not require any log in username or password, and it does not attach to any of the existing university’s systems like e-learning or PRISM, as the purpose is to enable students to see more alternatives of timetables they can have, rather than sticking to those courses that they have registered but not sure if they can actually go to those

classes or not. Since we will be using UTP timetabling system as the main study focus in this project, we will look into it in details in the following chapter.

The project is intended to request and grab data directly from UTP's website, module timetable page. However, the first step of prototype development will be using manually key-in timetable in the database first. Users will be prompted to select the courses they wish to enroll in, and the system will provide all the possible combinations of timetable available, showing the one with least number of clashes first. If the system is to be implemented for real, a simple text file generated from Syllabus Plus (UTP timetabling software) would be then processed and it will update the whole database, which will make the system real-time and accurate.

1.3.3 Relevancy of the Project

In the Analysis part we will see that UTP is in need of this system because currently the PRISM (Student Portal) and the timetabling system (Syllabus Plus) are not able to interact with one another. Therefore, this system can be considered as a temporary platform to assist students as well as helping to reduce hassles for timetabling personnel.

1.3.4 Feasibility of the Project

This project is estimated to require approximately one month of literature view, background studies and design, and another four months to implement and come up with the working prototype.

As of August 2012, which is almost the end of project timeline, the development has covered all the basic functions mentioned earlier in the proposal. Even though there are still some small parts can be further enhanced and modified, the current prototype is fully functioning. With a bit more effort and time, the system would be ready to be implemented for UTP students to use.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction to Timetabling

Timetabling is a common example of scheduling problem, and can manifest itself in several different forms. According to [1], “*Timetabling* is the allocation, subject to constraints, of given resources to objects being placed in space time, in such a way as to satisfy as nearly as possible a set of desirable objectives”. In this research, we are focusing on course timetabling for individual students, which means that the master timetable already existed, but we need to derive only those that meet the student’s need and arrange them in a way that there is minimal clash.

According to [2], the particular form of a timetable required is specific to the environment or institutions in which it is needed. For example, a school and university timetabling system might appear to be the same problem at first glance, but actually these two have many minor differences.

2.2 Application and Challenges

The construction of exam timetable as well as course timetable is a common problem for all institutions of higher education. Quite often it is done by hand or with the limited help of a simple administration system and usually involves taking the previous year’s timetable and modifying so that it will work for the new year. It is important because more and more higher education institutions are now introducing the concept of modular degree [3], which gives the flexibility to students in what courses.

The application of computers to timetabling problems has a long a varied history. Almost as soon as computers were first built there were attempts to use them to solve these problems. The first generation of computer timetabling programs in the early 1960’s were largely an attempt to reduce the associated administration work, and soon they were presented with the aim of fitting classes and teachers to periods. Because of

the problems complexity, the classical algorithms are less or not efficient at all, despite that there are some initiations based on known algorithms as heuristic search or modified back-track. The Slovakian aSc group offers usable timetable arrangement software which uses direct search algorithm. During the timetable arrangement process, if numerous requirements are taken into consideration the problem's complexity can easily grow out of the manageable domain, and then direct search algorithm does not provide satisfactory results within reasonable time. Simulated annealing (SA) algorithm is used for timetabling problem by A. Abramson. Tabu search (TS) is introduced for a possible solution for timetabling problem by Hertz and de Werra. Unfortunately, they are not effective enough in such a big problem space and they work on just one timetable. Simulated annealing and tabu search are examined for timetable problem by Colomi, Dorigo and Maniezzo. The efficiency and the convergence speed of these algorithms are compared to the similar values of genetic algorithms (GA) and the result was that the values of genetic algorithm and a specialized version of tabu search are almost the same while the ones of simulated annealing much worse than the other two.

There are many algorithms and solutions being researched and implemented such as Hybrid Evolutionary Approach [4], Generic Algorithm and Local Search [5], Graph Coloring and Constraint Manipulation [2], Soft Computing Techniques [6], Object-oriented scheduler [7] and many others.

Technologies evolved from time to time, but the problem of constructing an automated system for use with timetabling is still a particularly well known one. Many programs exist for this task, but perform well only in particular isolated environments [8].

2.3 Major Issues Encountered in Course Timetabling

Although the basic processes behind course and exam scheduling are the same, there are subtle differences that need to be dealt with [9]. For exams, the major problem that needs to be overcome is the lacking of exams in rooms, the fitting of exams to periods, and another common requirement is to minimize the number of consecutive or same day exams a student must take. Course timetable is a little more complicated. Each course

usually involves several lecturers per week, which are repeated at the same times every week for a semester. However, a course does not contain only lectures. There are labs and tutorials, and all of these (lectures, labs, and tutorials) tend to have alternative slots to choose from relative to the number of students enrolled for that particular semester/course. Clashing of course schedules have to be fixed, as students cannot just skip any of the sessions. Attendance and performance of the students are largely dependent on those attributes.

Conventionally, the timetabling problem has always been solved by leveraging human resource in education institutions. During the process numerous aspects have to be taken into consideration. Almost a week of work of an experienced person is needed to produce a timetable for an average institution and the result is often not satisfactory; it does not meet all the requirements. What is more when the preconditions change, the whole work becomes unusable, and has to be restarted from scratch [10]. Below are the commonly accepted hard and soft constraints for class scheduling. Hard constraints refer to those constraints that cannot be overlooked, while soft constraints can be ignored [11].

Hard Constraints: includes:

- Class clash: student of a class should not participate in more than one lesson at the same time.
- Teacher clash: whereby a teacher should not give more than one lesson at a time.
- Teacher's (strict) availability: a teacher is not necessarily available in the whole educational period, so it is important to know when a teacher is available and arrange the lessons according to this.
- Finite room capacity

Soft Constraints: timetable that violates these constraints are still usable, but not convenient for either students or teachers. It may include:

- Lessons held at early morning and long breaks

- Multiple lessons: some subjects, the multiple lessons should be placed next to each other.
- Teacher's soft availability: it is possible that a teacher ask not to have lessons in some timeslots.
- Maximum number of classes/hours a day: it's important when considering student's productivity and concentration.

2.4 Similar Systems Available in the Market

It is important to development a system that is new and able to meet users' requirements. Before development of Smart Timetable Organizer begins, I have conducted product search, and here are the few significant systems that are available out there in the market.

Wise Timetable

According to www.wisetimetable.com, the software provides two versions: one for universities, and one for primary and high schools. The system is quite powerful and complex, as it consists of many aspects of timetabling such as:

- Room scheduling
- Lecturer timetabling
- Lecture timetabling
- Individual timetable for each of the students

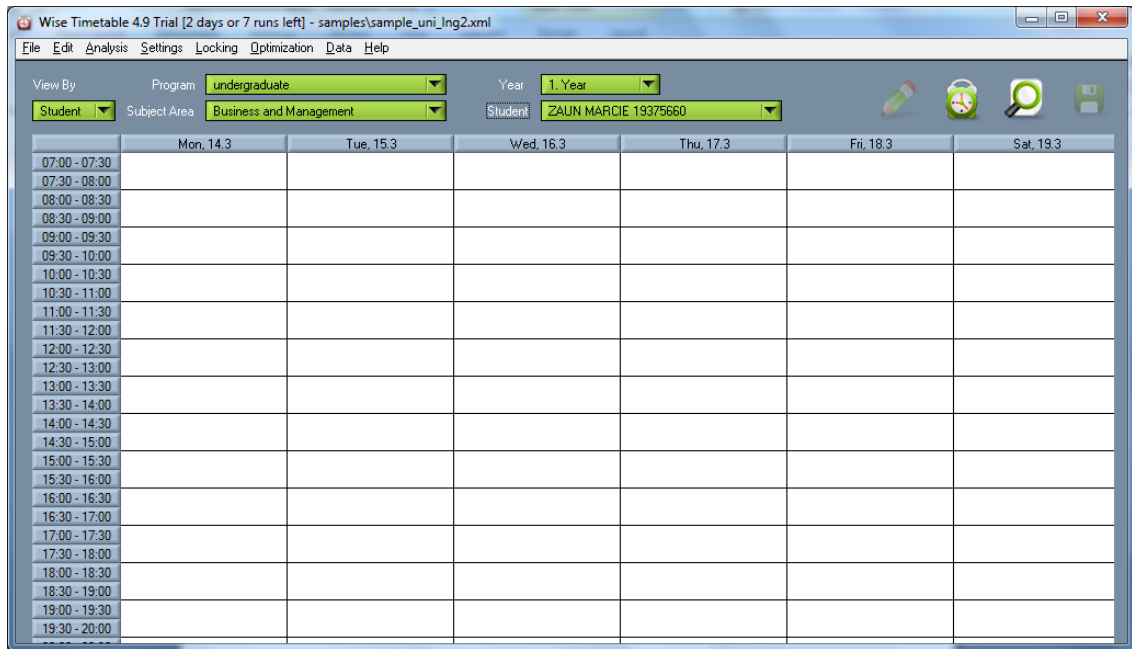


Figure 2.1: Wise Timetable – Student View

Characteristics: the software has no limit of size, provide interactive user interface, and user-friendly. There is notification by email, compatible with all popular database systems such as ORACLE, MySQL and MSSQL [12].

Drawbacks of the System:

This system is suitable for universities/schools that do not have existing timetabling system yet. The system do not link to any student portal that is able to let students to edit their profile or register for courses, therefore all those data input are solely from the university/school's responsibility. In case of a university, this means that the university does not entertain student's customization of their courses – have to follow strictly the university's curriculum structure – which is not applicable for universities that allows drag-and-drop of subjects like UTP.

All in all, the system above only caters the university/school's management needs, and does not offer options for students to arrange their personal curriculum.

Mimosa Timetabling System

The screenshot displays the Mimosa Timetabling System interface. It features a menu bar (File, Edit, View, Tools, Options, Window, Help) and a toolbar with icons for New, Open, Save, Print, Components, Courses, Weeks, Timetables, Optimise, Info, Colours, Font, Web, and Help. A banner at the top right says 'Click here to purchase full licensed version'. The main area shows two timetables side-by-side. The left timetable is for '14: FMAR FERNANDO MARIN' and the right is for '1: SOCE-12C SOCIALES ELECTIVO'. Both show a weekly grid with days (Lun, Mar, Mié, Jue, Vie) and rooms (R0, H1, H2, H3, H4, H5, H6, H7, H8, H9). Below the grids are summary tables for each timetable.

#	CODE	NAME	*	SUM	DONE	LNKS	LECT
1	12C	DECIMO SEGUNDO GRAL C	44	40	16x		
2	FMAR	FERNANDO MARIN	P	23	23	9x	
3	514	SALA 514	S	42	38	15x	

Figure 2.2: Mimos Timetabling System: Sample Interface

As stated in www.mimosoftware.com, the software provides scheduling and course planning for use in any kind of school and university varying type and size. Similar to Wise Timetable, Mimosa Timetabling System enables the creation of timetabling system by showing the available options how to assign, remove, move, and swap lectures. The system also stores the students who enrolled for the particular subjects, and we can view the number of students in the class so that the venue allocation can be done easily as well [13].

Drawbacks of the System:

Not much different from Wise Timetable, this system only caters for those universities who do not have existing timetabling system yet. Also, this system do not emphasize on student's clashes in the timetabling. So there is no solution provided in term of this.

UniTime Timetable

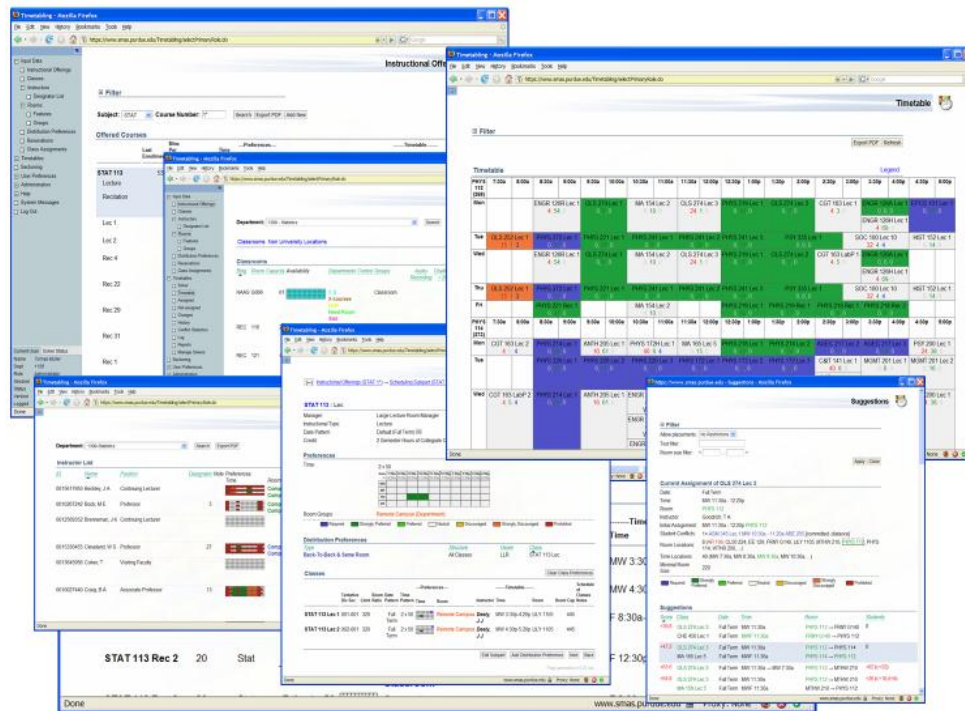


Figure 2.3: UniTime Timetable Screenshots

This system from www.unitime.org focuses on matching the sets of classes required by each student to the available class spaces, so that all (or as many as possible) of the students' education requirements are met. The main idea is that: if all courses are offered at only one time and students are unable to attend the courses they need, then we need a better course timetable [14].

Drawbacks of the System:

As stated above, the system only concerns with the venue clashing. This system is application to universities whereby venues are strictly scarce, which is not a big concern here in UTP, for instance. We have plenty of lecture room and lecture theatres available for conducting lectures and tutorials as well as lab session for now, to an extent that there should be no problems regarding this any time soon.

In conclusion, even if there is quite a number of timetabling system out there in the market, there is not really a system that caters only the need of student's preference of courses they want to take. Practically in Malaysia, there are a number of universities like UUM (Universiti Utara Malaysia) and KLIUC (Kuala Lumpur Infrastructure College), have already implemented student's personal timetable attaching with the student portal. However, these two universities' curriculum structure is not so flexible and customized like the one we are having here in UTP [14]. In UTP, students are facing more clashing problems than anywhere else, and we would be looking into this in details in Data Gathering section – where we will see all the response from students in these universities regarding their existing timetabling system and comments regarding Smart Timetable Organizer. Figure 2.4 is an example of student's personalized timetable provided by KLIUC in Student Portal:

Figure 2.4: Student's Personal Timetable (KLIUC)

Another need of Smart Timetable Organizer in UTP is that: we have timetabling systems for lecturers, staffs, course's module timetable, semester's master timetable, and exam timetable as well. However, what we are still lacking even up today is a system that can assist students to prepare their **personal timetable** despite all those hard and soft constraints as discussed earlier. We can also implement student's personal timetable that relates to the student's course registration directly, but somehow it is not very useful at the beginning of the semester whereby students still are in add/drop period.

CHAPTER 3

METHODOLOGY

3.1 Research & Development Methodology

Research Methodology

Before the designing and implementation of the project, intensive study regarding literature review and background studies will be carried out via various available sources such as internet sources, books, journals, papers, etc. These sources are accessible personally as well as from our IRC (Information Resource Center). Interviews and questionnaires regarding the proposed topic will also be conducted both at students' and lecturers' side in UTP as well as other universities and colleges. Data gathering and analysis is to be performed intensively to support the study before the project is to be implemented. Last but not least, after the implementation of the system, feedbacks and comments are to be collected from users also, to ensure that the system is beneficial.

Development Methodology

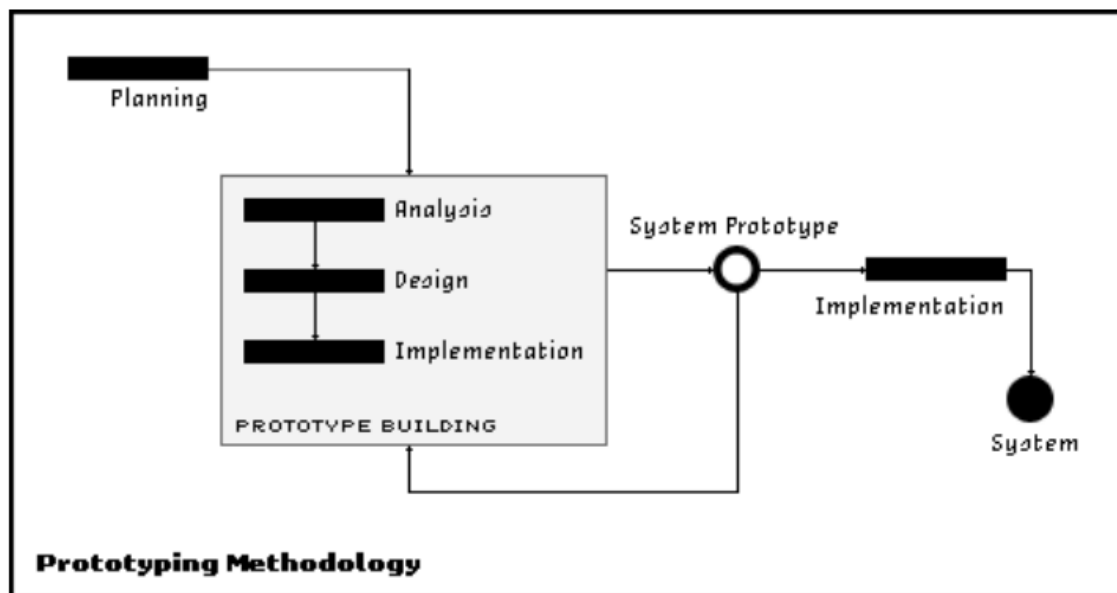


Figure 3.1: Development Methodology - Prototyping

The methodology which is most suitable for this system is Prototyping Development Methodology. Figure 3.1 shows the common phases of Prototyping Development Methodology. When the business process is likely to be changed as the project proceeds or when the project sponsor has little idea of what system to be built, which is in this case the developer is having very high uncertainty, it is the most suitable methodology to pursue. The Analysis, Design, and Implementation phase are performed only once, yet they will be reviewed from time to time as the cycle repeat. The cycle repeated continually based on the sponsor comments until the prototype successfully meets the requirements. The last version of the prototype will then be called the final system. Prototyping development needs only initial basic analysis and design, but as the result important system functions may not be recognized until somewhere in the middle of project timeline. This process goes on and on until all the initial functional requirements have fully met.

3.2 Project Activities

The list of tasks that should be completed for the project and their status are described as in Table 3.1.

Task No	Task Name	Duration (in days)	Status
1	Identify a topic area and define title	5	Completed
2	Discuss with supervisor on the next step	3	Completed
3	Prepare for literature view, background studies, objectives, and methodology	7	Completed
4	Create work plan and Gantt Chart	2	Completed
5	Analyze as-is process and define to-be process	4	Completed

6	Gather requirements and data necessary for the analysis	7	Completed
7	Finalize the functions (system specification)	3	Completed
8	Create functional, structural and behavioral models	7	Completed
9	Develop Design Strategy	3	Completed
10	Architecture and Interface Design	5	Completed
11	Program Design	5	Completed
12	Development of the system	90	Completed
13	Testing	5	Completed
14	Provide Feedback to and fro Users and Modify requirement if any	5	Completed
15	System Implementation and Documentation	14	Completed

Table 3.1: Project Activity List and their Status

As Table 3.1 describes, the project implementation was carefully planned by allocating a suitable time period to work on each of them. Smaller tasks are given less time, whereby the bigger and more complicated tasks are given more time. This list of tasks provides convenience for development, as it can be crossed out and next task is shown clearly.

The summary of the activities is depicted in the Figure 3.2 in the next page. The figure basically grouped the activities into main ones following the timeline. The project timeline is divided into two semesters, January and May 2012. In short, the project planning up to design phase were done during January semester, while implementation, testing, report and presentation will be done and concluded in May semester (September 2012)

3.3 Key Milestones

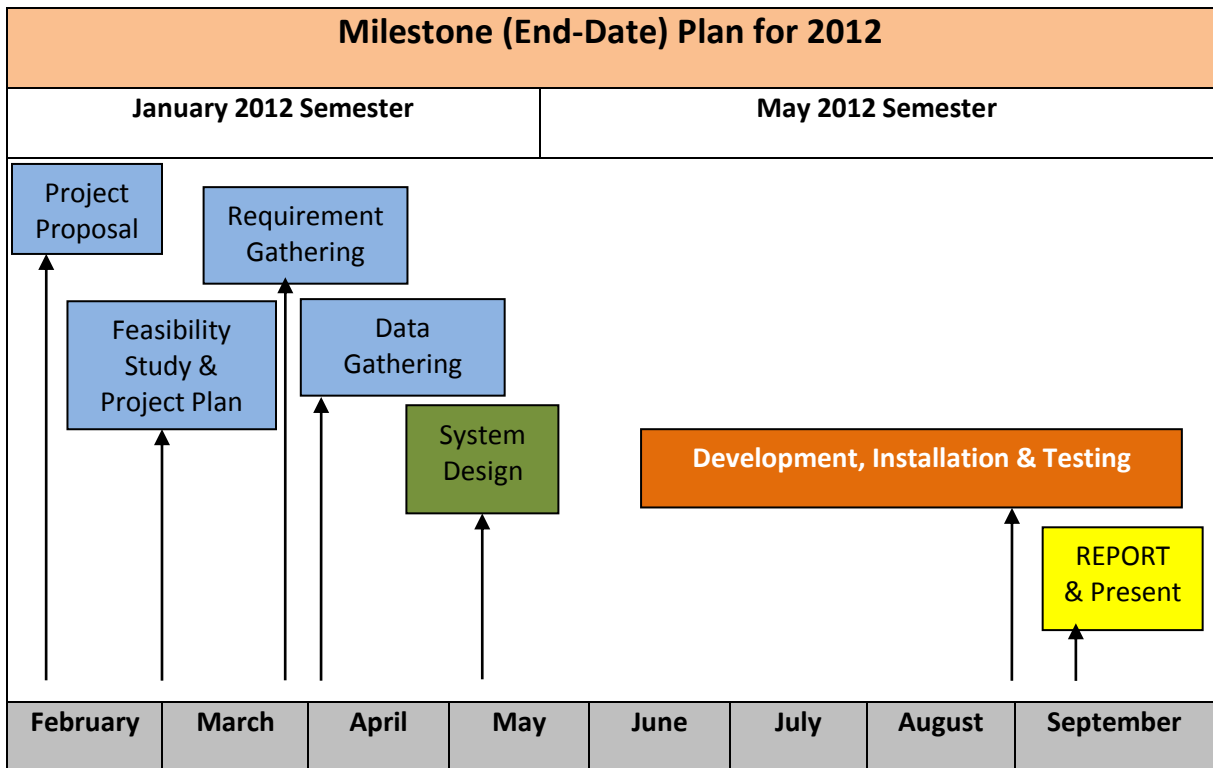


Figure 3.2: Milestones for the Project

3.4 Gantt Chart

The Gantt chart as exhibited in Table 3.2 shows the estimation of time needed for each tasks in order to complete the project. It is similar to the list of project activities above, but the difference is that Gantt chart provides a clearer view of the timeline as in calendar-view. The chart also shows the activities that can be done in parallel, and what activities needed to be done separately.

No.	Task	Duration	Week (estimated to require 7 months - 28 weeks to complete the project)																											
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28
Planning Phase																														
1	Identify Topic Area and Define Title	1 Week																												
2	Project Feasibility Study	1 Week																												
3	Project Plan	2 Weeks																												
Analysis Phase																														
4	Methodology Analysis	2 Weeks																												
5	Intensive Literature Review and Information/ Data Gathering	2 Weeks																												
6	Result and Analysis	2 Weeks																												
7	Requirements Gathering	2 Weeks																												
Design																														
8	System Design	4 Weeks																												
9	Architecture and Interface Design	4 Weeks																												
10	Database Design	4 Weeks																												
11	Program Design	4 Weeks																												
Implementation																														
12	System Construction	13 Weeks																												
13	Installation	13 Weeks																												
14	Testing and obtain users' feedback	1 weel																												
Report and Presentation																														
15	Support Plan and Documentation	2 Weeks																												

* dates are subjected to change

Table 3.2: Project Gantt chart throughout the two semesters

3.5 Tools

The prototype of Smart Timetable Organizer which is in the form of a website has been developed using the following set of hardware, software, and programming languages:

Hardware

- Personal Computer, Sony VAIO SR12G
 - Intel® Core™ 2 Duo 2.26GHz
 - 4GB RAM
 - 200 GB Hard Disk (Internal)
 - 1 TB External Hard Disk & 320 GB External Hard Disk (Back-up Purposes)

Software

- Notepad ++
- WampServer 2.2D:
 - PHP 5.3.10
 - Apache 2.2.21
 - MySQL 5.5.20
 - PhpMyAdmin 3.4.10.1

Programming Language

- MySQL for databases
- HTML, CSS for webpage formatting
- jQuery for website interaction
- PHP as the main language of development

CHAPTER 4

RESULT AND DISCUSSION

In this chapter, all the result and discussion will be presented and discussed. Since the development of the system is based on Prototyping Methodology, the analysis and design phase will be revised from time to time until the prototype is finalized as the final system.

4.1. DATA GATHERING & ANALYSIS

There are several methods that have been used to gather data. In this project, we have been using two methods of data gathering: interview and questionnaire.

4.1.1. Interview

The aim of the interview is to get technical data and information from UTP's management regarding course timetable. The first personnel interviewed is as below:

- Person Interviewed 1: **Mr. Nor Razmi B Ahmad Suhami**

Department:	Academic Central Service (ACS), UTP (Academic Administrator Unit)
Date/Time:	19 March 2012, 3:00 PM

From the interview, some of very useful information about UTP timetabling system was obtained. Academic Central Services department is the body that prepares all timetabling for UTP such as Course Timetabling, Venue Timetabling and Arrangement, Lecturer Timetabling, and to cater all the issues involved in these types of timetables. Referring to the diagram below, the department is currently using "Syllabus Plus", which is a system that arranges all the timetables needed.

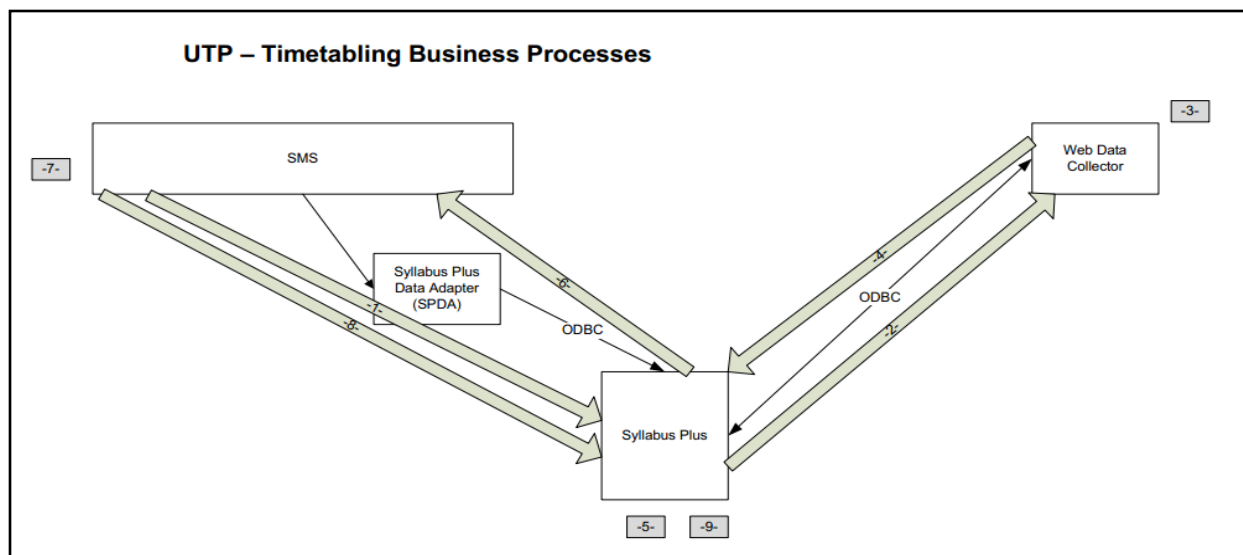


Figure 4.1: UTP Timetabling Business Processes

The process of UTP Timetabling is to be explained as follows:

- (1) Each department will need to provide ACS the Timetable Workload and Courses Offered. The document consists of courses offered for the semester (for the whole year – in this new trimester system) and each course would have lecturer(s) responsible for the delivery. All of the said information is then loaded to Syllabus Plus.
- (2) Syllabus Plus processes all the information and export the data to WDC (Web Data Collector) which serves as the main database for storage.
- (3) WDC collect all the activity data and store them accordingly.
- (4) Syllabus Plus then import from WDC the Activity Templates, Activities, Programs, Courses, and Staffing required.
- (5) After getting all the necessary information, WDC will create Draft Timetable that UTP is currently using as Course Timetable for students, provided in the form of Master Timetable (for each batches of student according to curriculum structure) and Module Timetable (for each courses).

The process currently stops at the above step, but that was not the intentional ending. The purpose of the whole process is to continue to the next few steps as follows:

- (6) Syllabus Plus will import all the draft timetabling information to PRISM (UTP's Student Portal).
- (7) PRISM will therefore collect all student registration data.
- (8) PRISM will match the student registration data with the draft timetabling information acquired from Syllabus Plus, in order to suggest to students the possible clashes that might occur.
- (9) Taking into consideration about clashing based on the import of all information from PRISM together with draft timetable, Syllabus Plus is able to create the final timetable that caters for minimal clashes.

According to Mr. Nor Razmi, the above step 6-9 is still in construction. There are many technical problems that up until now do not allow Syllabus Plus and PRISM to be able to fully interact and generate full output. *This is the reason that clashing in UTP is a really big issue and unavoidable at the moment, and this is also the reason why Smart Timetable Organizer will exist to assist to minimize the clashes and hassle from student's side.*

- Person Interviewed: **Mr. Azizi B Ahmad**

Department: IT & Media Services, UTP
(IT Application Development & Support)

Date/Time: 28 March 2012, 10:30 AM

(With the presence of Mr. Nor Razmi too)

In timetabling team, there is only two person involved: Mr. Nor Razmi and Mr. Azizi. Mr. Nor Razmi handles the creation and maintaining of the raw timetables itself, while Mr. Azizi takes over the set of timetables and publish it to the server of UTP Website.

Mr. Azizi kicks start his work when he gets the .spf file from Mr. Nor Razmi. This .spf file is the working file from Syllabus Plus that contains all the information of timetables. Then he uploads the file, configure it, create databases and views, and inform Mr. Nor Razmi when it is done. The detailed process is much more complicated. It includes the admin assignation, connection to databases, test connection, create services, window components, and configuration of the pages,

etc. Mr. Azizi mentioned that even though the work is only done every beginning of the semesters, but it is quite tough and complicated. Usually in timetabling unit in other universities, there are at least ten people handling this issue, whereas in UTP there are only two people. Currently IT department is still trying to figure out how to integrate PRISM with Syllabus Plus. They are actually still in the process of trying to understand fully how it works, and so far there is not much luck to realize when the experiment will succeed yet.

Mr. Azizi has expressed his support in the development of Smart Timetable Organizer. He agrees that the system would be able to save a lot of troubles for him and Mr. Nor Razmi, and allow the finalization of timetables to be done quicker. He has also mention that he would be more than happy to suggest to UTP management to employ the website as soon as possible given that the website itself functions properly.

4.1.2. Questionnaire

This initial stage of questionnaire distribution was done online by using Google Spreadsheet. So far I have received **86 responses** from four different universities such has UTP (Universiti Teknologi PETRONAS), UiTM (Universiti Technology Mara), UUM (Universiti Utara Malaysia), and KLIUC (Kuala Lumpur Infrastructure University). However, the main respondents are from UTP, due to the convenience of distribution and networking. Nonetheless, this questionnaire process will be extended to the rest of universities, and the final result would be described in the final report of the project.

4.2. Findings

Below is the result from the questionnaire:

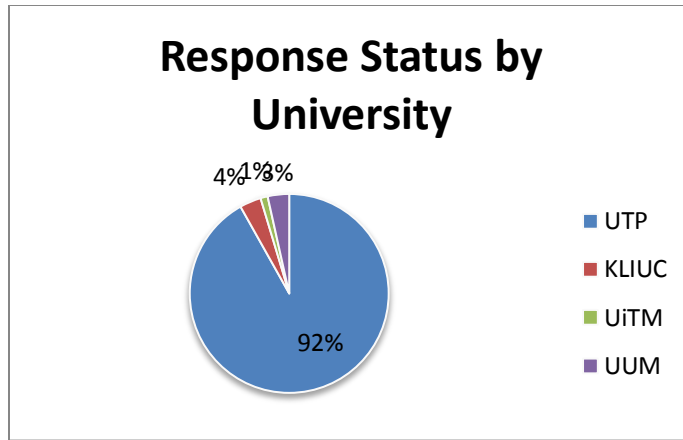


Figure 4.2: Questionnaire response Status by University

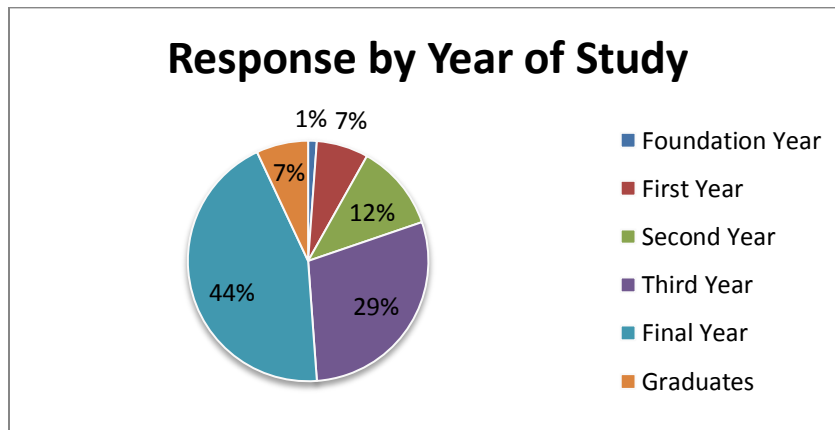


Figure 4.3: Questionnaire Response by Year of Study

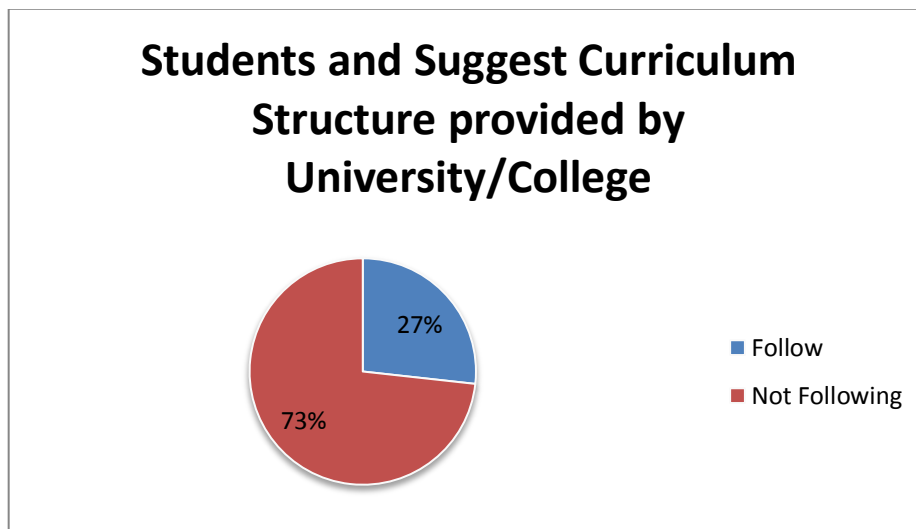


Figure 4.4: Questionnaire response to whether the students follow suggested curriculum structure provided by the university

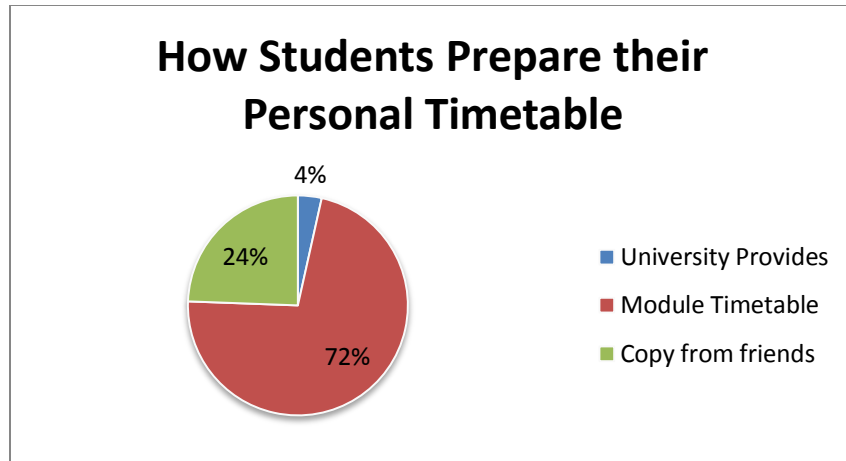


Figure 4.5: Questionnaire response on how students prepare their personal timetable

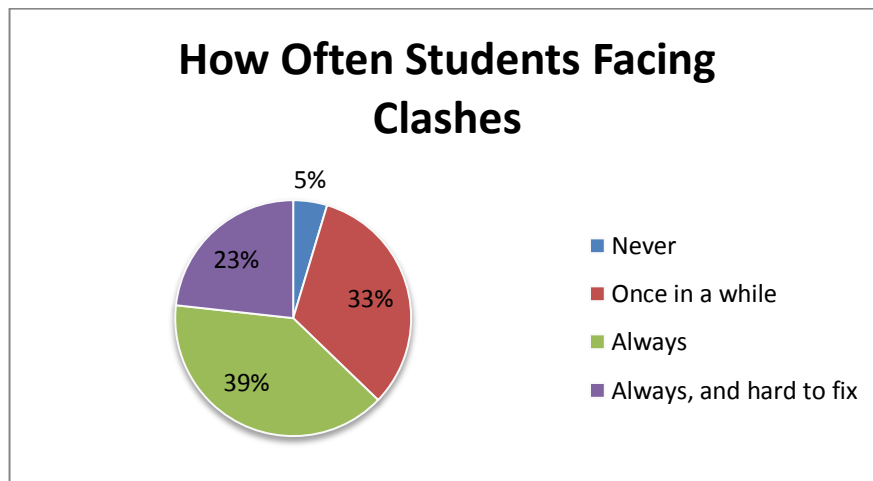


Figure 4.6: Questionnaire response on how often students facing clashes

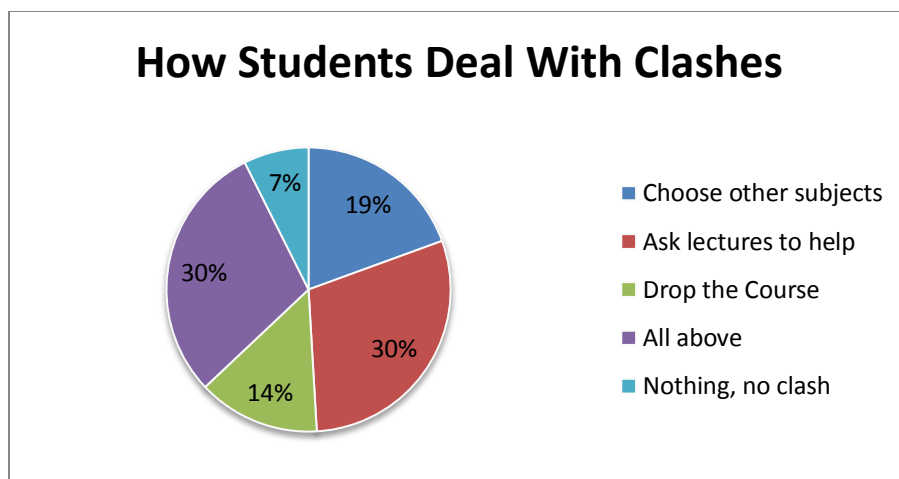


Figure 4.7: Questionnaire response on how students deal with the clashes

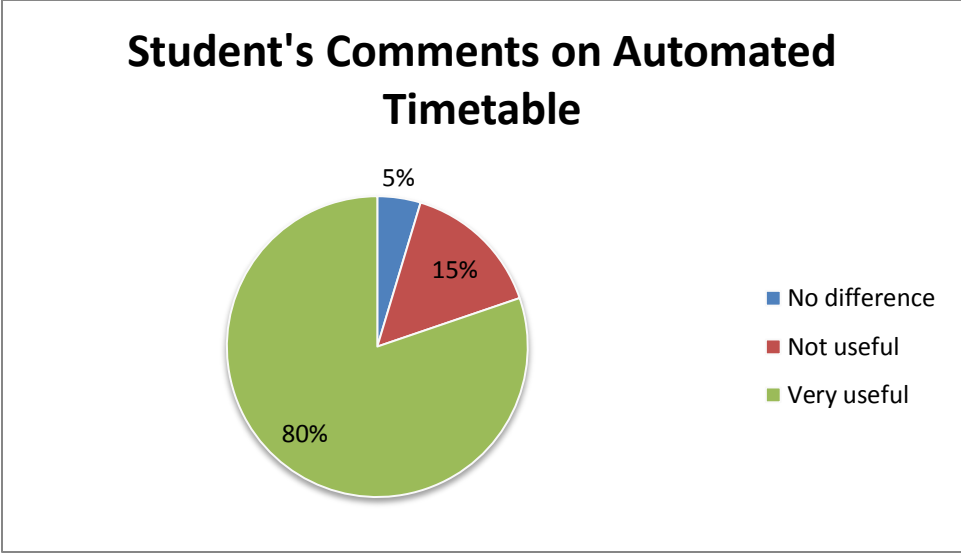


Figure 4.8: Questionnaire response on the comments regarding automated personal timetabling

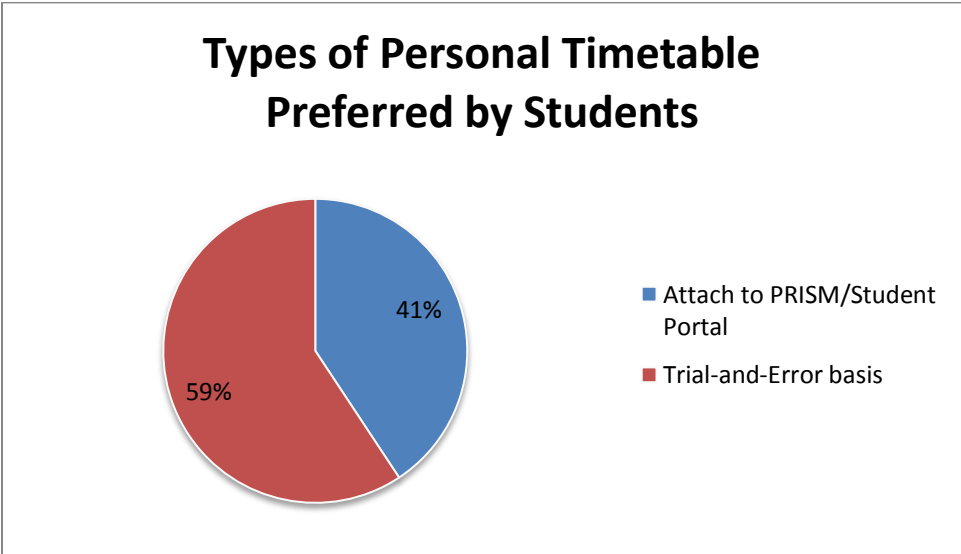


Figure 4.9: Questionnaire response on the preferred type of personal timetable

Some of the comments from the questionnaires (quoted directly):

- This is a good system. I heard some other universities have implemented, but link to their courses registered. I prefer it trial-and-error.
- Personal timetable that is made automatically? It sounds superb to me. Preparing timetable always gives me a lot of headache at the beginning of the semester and for the first few weeks, it's always about dealing with all the clashes and that's very troublesome indeed!
- This would be really helpful. I wish someone were building this since I was first year :(
- As I've mentioned to you before, your idea is good, try and get the current method of sorting from ACS and see what they do, perhaps it might aid you in your prototype :)
- Questions... How the system works when there is no internet or maybe server down? Because you can see, when the internet is down students are face difficulties whether to add drop or view their time table.
- it will be better if the timetable can cater clashes automatically without need to meet lecturer and discuss with classmate
- Please implement the idea soon! it would be great for students in future

4.3 Prototype Modeling

4.3.1. As-Is System & To-Be System

- As-Is Process

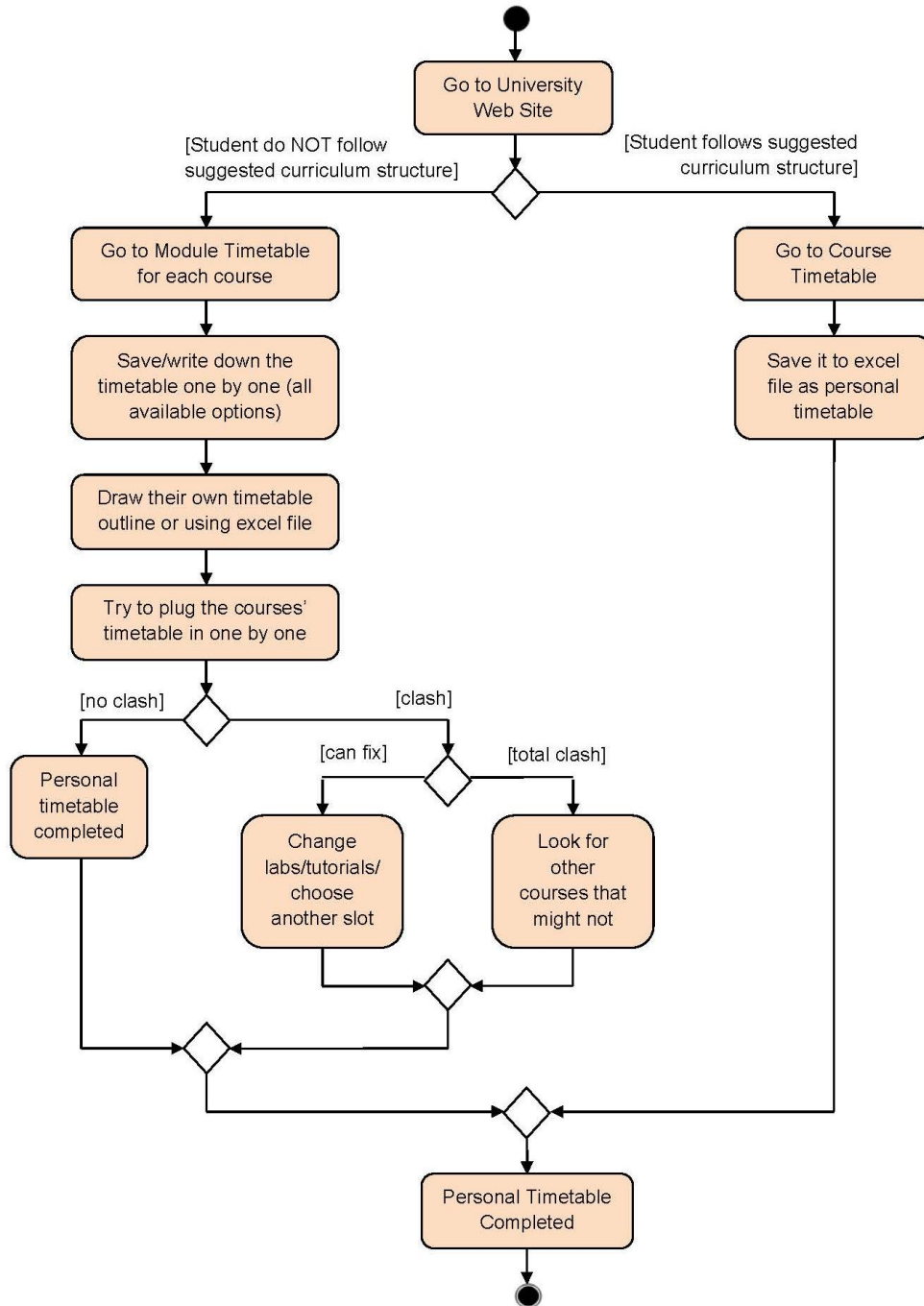


Figure 4.10: As-Is process of how students make their personal timetable

Figure 4.10 depicts the as-is process of how UTP students normally make their personal timetable at the beginning of every semester. There are two types of processes involved. First, if the students always follow the suggested curriculum structure provided by the university, then they can just go to UTP Timetabling center and download the course timetable based on their year and programme of study. In this part, Smart Timetable Organizer does not come in play much, because course timetable usually has no clash, unless students drag and drop the subjects from another semester.

In the case that students do not follow the suggested curriculum structure – as most of the students do not – students will start by going to Module timetable, check the timetable for each course one by one, save it to their personal computer or write the details down on a piece of paper or copy the cell contents to Excel/Word. Once they have finished with all the subjects, they will then try to plug in the details of the classes (day and time) to their personal timetable by putting the lectures first, then select the most convenient time for their labs and tutorials.

If there is no clash, then they can just call it done and their personal timetable is finalized. However, that is not always the case. Sometimes there are clashes, so they would see whether they can go to other lab or tutorial sessions, or whether there are other groups for lectures available to choose from. If they still cannot fix it this way, usually they will just attend the first class, meet the lecturers and discuss the problem. Lecturers would try their best to find new slots (depends on the lecturers too), but it all subjected to their availability and decision. Worse come to worst, students have to drop the course, take other subjects maybe and yes – have to rearrange their timetable again starting from the beginning.

- **To-Be Process (Smart Timetable Organizer)**

The to-be process, which is depicted in Figure 4.11, is the process of Smart Timetable Organizer. There are two modules available for this website: users and administrators.

User is student who wishes to check/get their timetables arranged. Different from the manual way above, all they have to do is select the number of courses they wish to take, and then select the subjects, and click “PROCEED”, the best possible timetable they can adopt will be arranged and displayed for them. They can choose to display more combinations if they want to, and they can save the timetable to PDF for their future use. For administrator side, they have to log-in with administrator account. The purpose of having this site is for the administrators to easily update/make changes to the database. They can add/edit/delete administrators, subjects, groups, and session table in the database without having to go manually to phpMyAdmin.

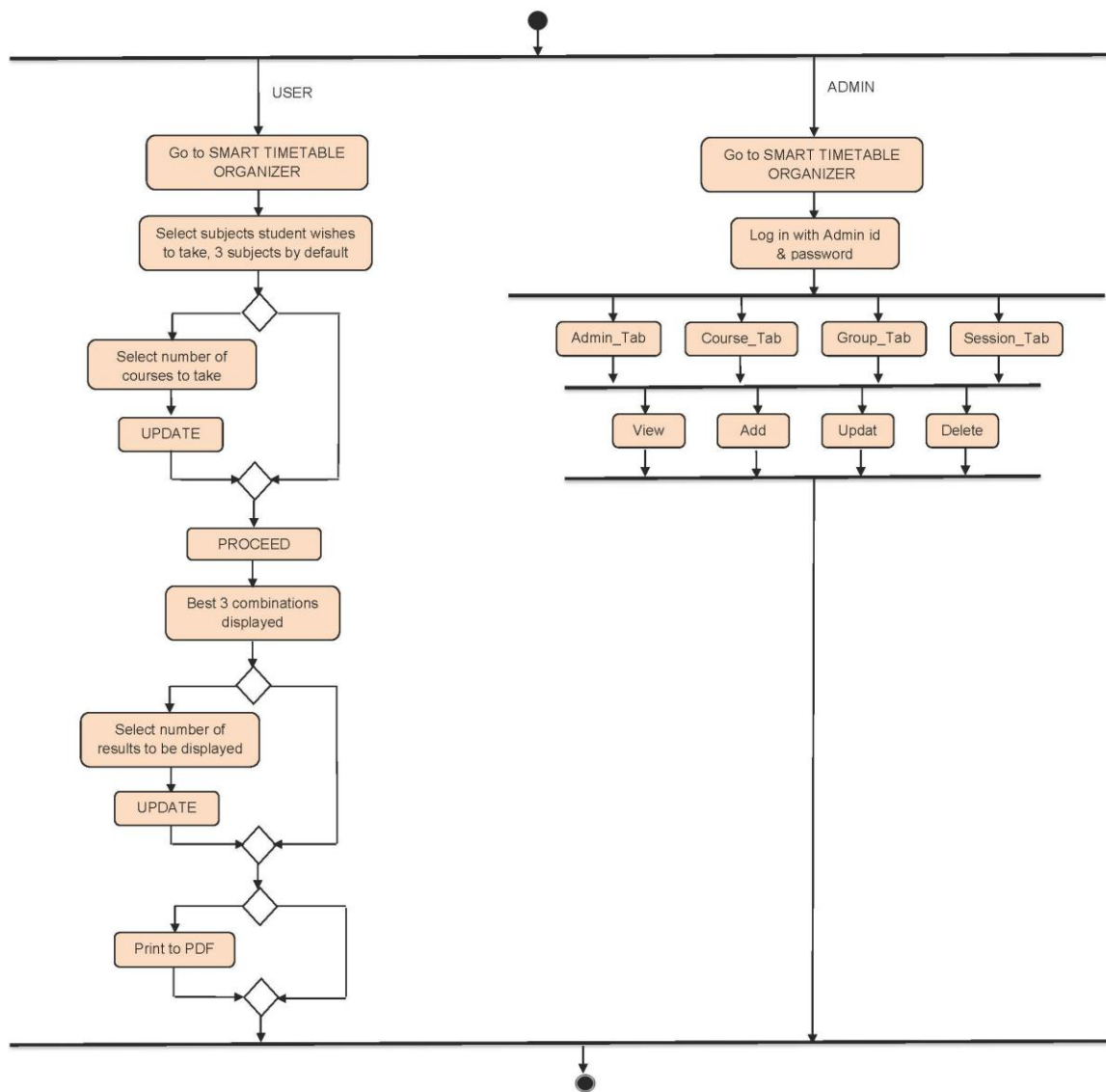


Figure 4.11: To-Be process of Smart Timetable Organizer

4.3.2 System Architecture

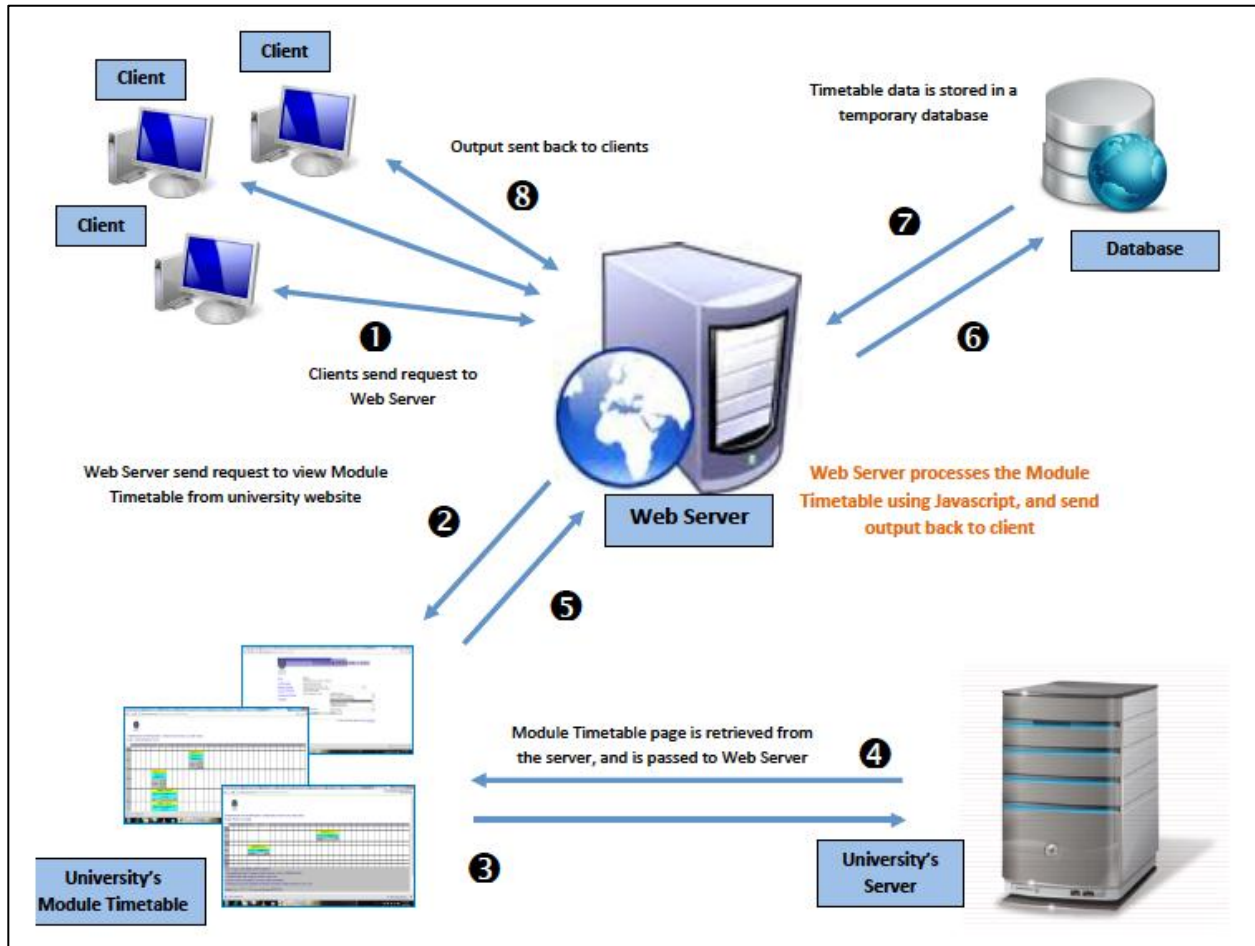
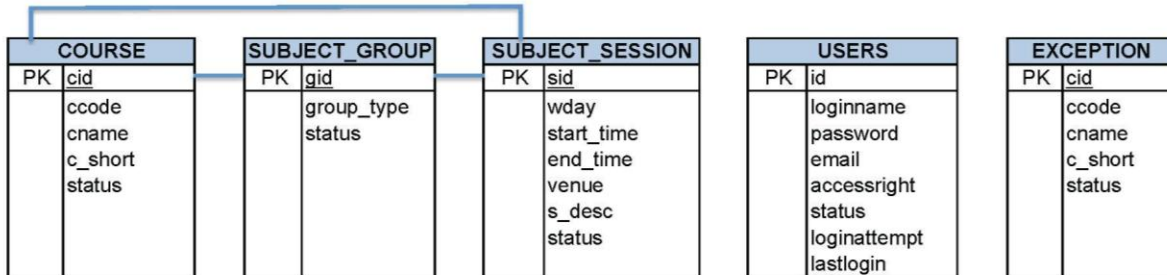


Figure 4.12: System Architecture of Smart Timetable Organizer

- User (client) sends request to webserver via Smart Timetable Organizer webpage, by inputting the courses they wish to take
- Web server (the system) will process the request, by obtaining the data of the timetables from the university's module timetable page. The module timetable page is retrieved from the university's server.
- Once the web server receives all the data required, it will store all those information in database, realizing the subject's particulars such as lab/lecture, time and venue, etc.
- The system will then generate all possible combinations of the subjects and display it to the user.

4.3.3 Database Design

TABLES:



VIEWS:

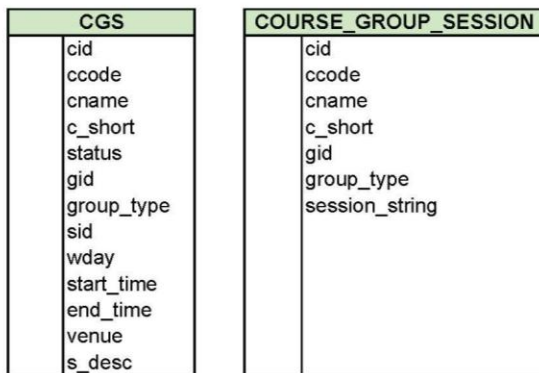


Figure 4.13: Class Diagram of Smart Timetable Organizer Database

There are four main tables in the database of Smart Timetable Organizer. Table “Users” stores all the administrators’ particulars like username, password, access rights, and log-in attempts. Table “Course” stores course code, course name, and course short name, while the table “Subject_Session” stores the details of lectures, labs, and tutorials such as day, time and venue. In other to avoid confusion of whether the session is lab or lecture or tutorial, we have created another table as a bridge entity “Subject_Group” that stores the group type whether it is **main** (1) or **sub** (2). This mechanism will provide a way for the system to realize which sessions are compulsory, and which are alternatives. We also need this mechanism to generate the combination of the timetables. These three main tables consist of a field called “status”. This field is for the benefit of the Content Management System (CMS) to know whether the data is

active (1) or inactive (2). It simply means that the modification and viewing of table data is only for those with the status of “1”. By default, the status is “1”.

Since the system works better with views, two views have been created: CGS and COURSE_GROUP_SESSION to reduce the hassle of joining tables during extraction of data.

Assumption: below is the assumption of the data input in the database:

(a) Each subject consists of groups, and each group consists of sessions.

(b) A session can be a lecture, a lab, or a tutorial.

(b) The sessions can be categorized into:

- main : for example: lecture

- sub : for example: lab

(c) A subject can only consist of:

- group(s) of lectures OR

- group(s) of labs OR

- group(s) of tutorials OR

- group(s) of lectures & group(s) of labs, OR

- group(s) of lectures & a group of tutorials & group(s) of labs (alternate), OR

- group(s) of lectures & a group of labs & group(s) of tutorials (fixed), OR

** Therefore, the exception case in this study:

group(s) of lectures & groups of tutorials & groups of labs

According to the assumption above, we can see that the system’s weakness is when it comes to a subject that requires the student to choose sessions from three categories: lectures, labs, and tutorials. If there is only one tutorial or one lab to choose from, there would be no problem, but if there are groups for all the sessions to choose from, that is where the exception occurs. In this Smart Timetable Organizer 1.0, there is not yet a solution for the above exception.

These are the subjects that fall in the exception category and the current prototype cannot cover yet (many groups of lectures + many groups of labs + many groups of tutorials) [15]:

- 1- Academic Writing LCB1042
- 2- Analytical Chemistry CCB2073
- 3- Chemical Analysis CBB4032
- 4- Chemical Process Dynamics & Control CBB3013
- 5- Communication Systems ECB2163
- 6- Control Systems (ME) MBB2094
- 7- Digital Electronics I ECB1053
- 8- Digital Electronics II ECB2133
- 9- Digital Signal Processing EBB3043
- 10- Drilling Engineering PCB2033
- 11- Electrical Power & Machines MCB1033
- 12- Facilities Eng Transportation & Storage MBB4513
- 13- Fluid Flow & Transport Processes CCB1033
- 14- Introduction to Material Science MCB1012
- 15- Mechanical Eng Thermodynamics II MCB2063
- 16- Microprocessor ECB2143
- 17- Modern Control Engineering EBB4213
- 18- Offshore Structures VBB4423
- 19- Organic Chemistry CCB1013
- 20- Physical Chemistry CCB1023
- 21- Power Systems I EBB3013
- 22- Power Systems II EBB4333
- 23- Principles of Chemical Engineering ECB1064
- 24- Probability & Statistics FCM2063
- 25- Process Safety & Loss Prevention CBB2093
- 26- Reaction Engineering CBB2074
- 27- Reservoir Modeling & Simulation PAB3053
- 28- Reservoir Rock & Fluid Properties PCB1033
- 29- Solid Mechanics (ME) MCB2034
- 30- Structured Programming & Database Systems (ENG) TCB2073
- 31- Vibrations MBB4023

32- Engineering Drawing (CHE) CBB1052

Another minor weakness of the current prototype is that it cannot recognize the subjects that have many groups according to programme yet. For example, for Malaysian Studies (MPW2133), there are several groups allocated specifically for ICT, BIS, ME, CV and so on so forth. But the current system does not take into account which programme the user belongs to. It only can recommend the slot that best fit in well with the other subjects. Those subjects in this category are:

- 1- Computational Method FCM2043 – *Lecture got two groups*
- 2- Engineering Economics & Entrepreneurship GBB3173 – *Lecture got 3 groups*
- 3- Fluid Mechanics – *Lecture got 2 groups*
- 4- Islamic Studies MPW2143 – *Lecture got 5 groups*
- 5- Introduction to Management GBB2053 – *Lecture got 3 groups*
- 6- Introduction to Oil & Gas Industry & Sustainable Development PCB1012 – *Lecture got 2 groups*
- 7- Ordinary Equations FCM1023 – *Lecture got 6 groups*
- 8- Malaysian Studies MPW2133 – *Lecture got 5 groups*

In summary, in our current database we have:

Total subjects covered:	181
Total Groups associated:	533
Total Sessions associated:	807

Table 4.1 Summary of Subjects Covered

** Please see the appendix at the back of the report for the list of subjects covered

4.3.4 Interface Design (User Interface)

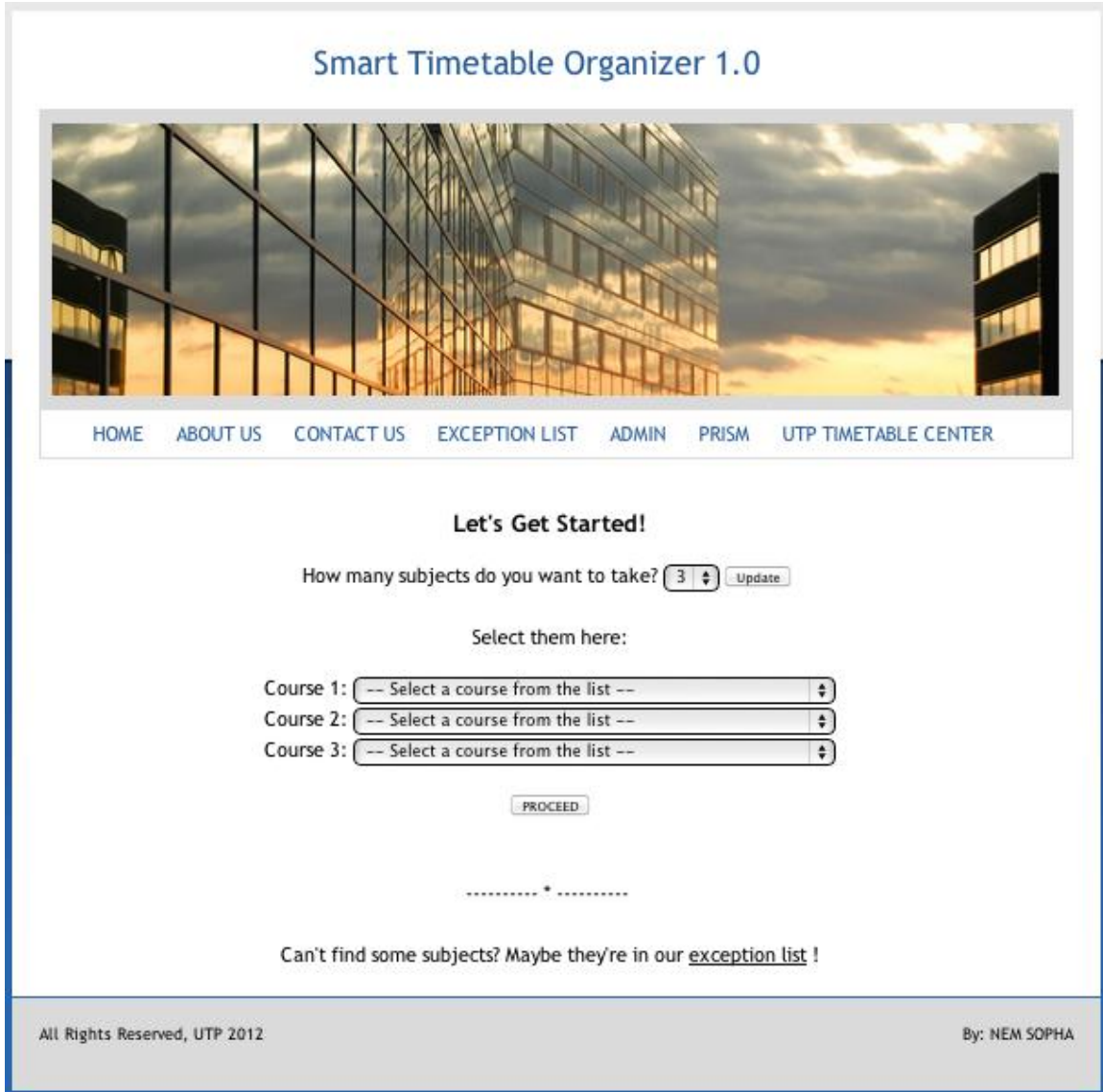


Figure 4.14: Home Page

This home page is basically to allow users to select the courses they wish to enroll in. The default number of courses to be selected is 3, which is the minimum. Users can choose to select up until 10 subjects by selecting from the dropdown list and click “UPDATE”. The lists of subjects are generated from the database, sorted by alphabetically ascending. Before submitting the courses, the system will first check for duplicates. The website also provides links to About Us, Contact Us, Exception List, PRISM, and UTP Timetabling Center, just for users’ convenience to related sites.

ABOUT US

The purpose of this page is to inform the users what is the purpose of the project, what our objective is, as well as inform users how to use the system.

CONTACT US

This page will list down the three main personnel involved in timetabling process. The first one would be the developer of this website, in the event of any technical issues and inquiries. Second and third personnel would be the two key people that handle UTP timetabling process, from Academic Central Services (ACS) Department and Information Technology Media and Services Department (ITMS) respectively.

EXCEPTION LIST

Since the system haven't yet covered all the subjects in every cases yet, those subjects will fall into this exception list which means that they would not be able to find these subjects in the list in main page.

PRISM

After seeing the timetable, students may want to add or drop courses, and this is just a link that directly goes to PRISM. The purpose is to give convenience to the user.

UTP TIMETABLING CENTER

Similar to the PRISM link, this one provides direct link to UTP Timetabling Center in case users want to clarify and check whether the provided timetable is accurate or not. Users might also want to compare and see whether Smart Timetable Organizer has obtained the latest updates from the timetabling center or not.

Once the user has selected the subjects correctly and clicked “PROCEED”, the system will process using the following pseudo code in Table 4.2.

```
Begin

1.0  Courses are selected and stored as $course_array[];

2.0  Get from database all groups in both category: main & sub
      (separately) associated with $course_array stored as
      $group_collection[];

3.0  Generate a collection of all possible combinations from
      $group_collection[] and stored as $combination_arr_int[];

4.0  For each combination:

      4.1  Get all session id associated and stored in
            $final_session_id_array[];

      4.2  Get all sessions' details from database and use function
            getclash_number() to count the number of clashes and stored
            as $clash_count;

      4.3  All $clash_count generated from each combination is stored
            in $clash_array[];

4.4  Sort $clash_array[] ascending;

4.5  Use function print_table() to print the session details out in
      HTML Table format, by printing out 3 first combinations according
      to $clash_array[];

End
```

Table 4.2: Pseudo Code for Smart Timetable Organizer

For the step 3.0 whereby the combinations are generated, a simple explanation using pseudo code would be as the following:

```

Begin

    Count the number of categories involved as n=2*no_of_subjects
    selected

    Count the number of groups in Category 1 of Subject 1 as m

    Function find_comb (n categories)

        If n=1, no recursive of find_comb required

        Else, recursive of find_comb required

            //Find all combination for each group in Category 1
            of Subject 1

            1stGroup.find_comb (n-1 categories)

            until

            mthGroup.find_comb (n-1 categories)

End

```

Table 4.3: Pseudo Code for Combination Algorithm to generate all possible combinations for each group of course selection

Explanation:

After the user successfully and correctly chosen the subjects, those subjects' codes will be used to retrieve all groups associated and find all possible combinations from those groups. Each combination will be checked for clashes and the number of clashes for each combination will be stored in an array which is sorted ascending later on. Once the sorting done, we will be able to know which combination is the best combination (index 0), so we will display – in default – three best combinations to the user. Users can of course select the number of combinations they wish to see, which is further explained in the following screen shot.

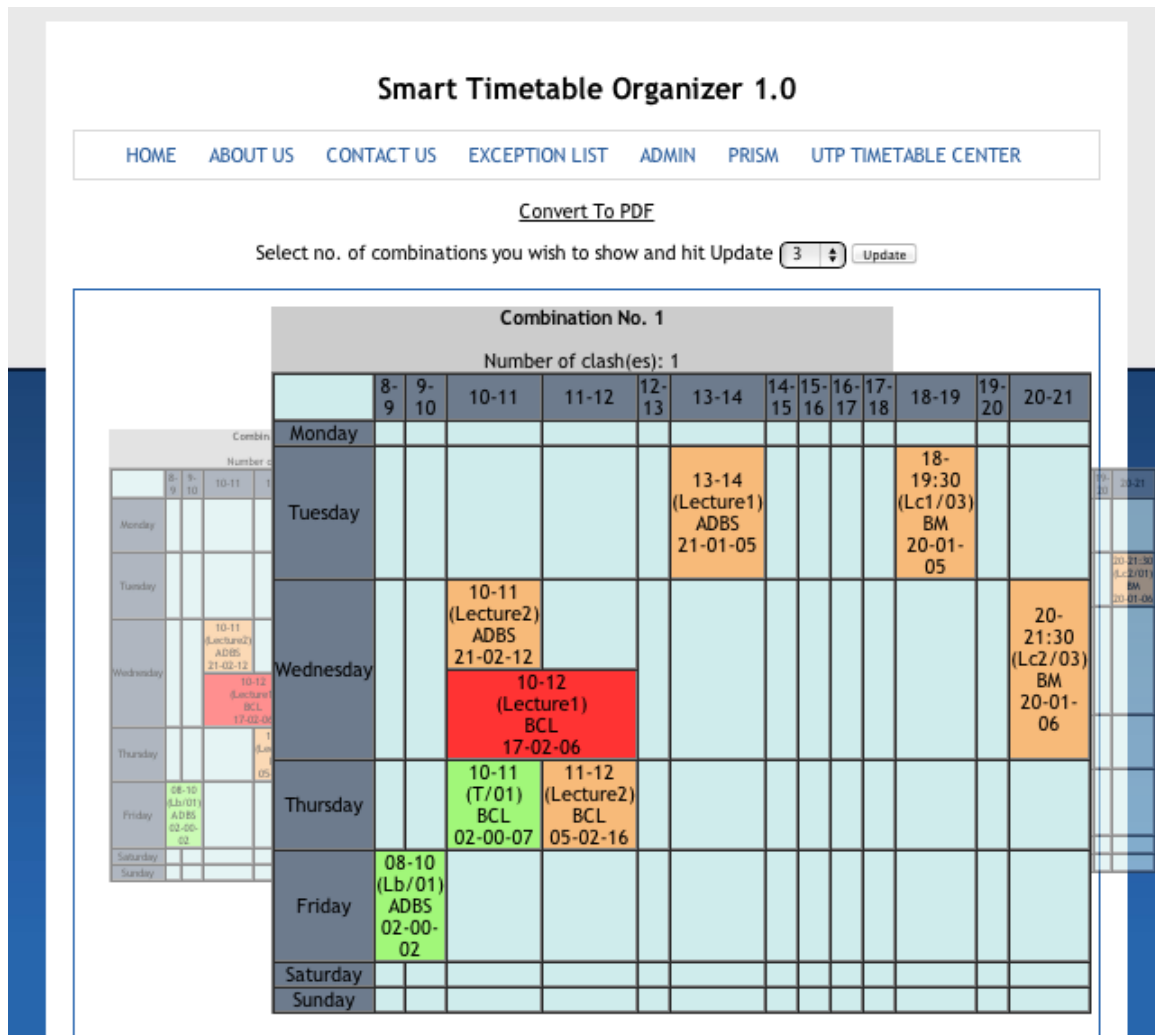


Figure 4.15: Result Page

As Figure 4.15 illustrates, this is the result page. The three best combinations are displayed in table format using RoundAbout, together with their id (Combination Number), and the number of clashes identified. As we can see, the first and best combination to come up consists of one (1) clash, which is highlighted in red. Lecture (main) group would be colored in pale orange, while Lab & Tutorial (sub) group will be in pale green color. All details are featured such as time, venue, and course's short name.

The purpose of having this course's short name is to make it easier for users to quickly realize which subjects the table is for. In UTP module timetable, they display the course code, which is not very convenient for users as most people do not remember the course

codes. However, if the user is not sure what short name belong to what course, they can always scroll down to the bottom of the page, they will see this detail of course selection as in Figure 4.16.

Lists of Courses Selected & Processed:		
No.	Course Selected	Status
1	SBB3013 : Advanced Database Systems (ADB5)	(SELECTED)
2	MPW2123 : Bahasa Kebangsaan B (BM)	(SELECTED)
3	GBB3113 : Business & Cyber Law (BCL)	(SELECTED)

Figure 4.16: Result Page – List of Courses Selected & Processed

This section is for the purpose of helping users to double check whether they have selected the right courses. Course code, course name, and the short name will be displayed for references.

Selecting Preferred Number of Combinations to Display

Back to the top of the result page, users can select the number of combinations they wish to see. The reason why we only display 3 combinations is because we want the system to be smarter which it is able to display best of the best combinations, and also the RoundAbout display would not look too stacked. But if users want to see all other combinations available, they can do so by just selecting the number from drop down list and click “UPDATE” then the number of combinations displayed in the roundabout will be updated accordingly.

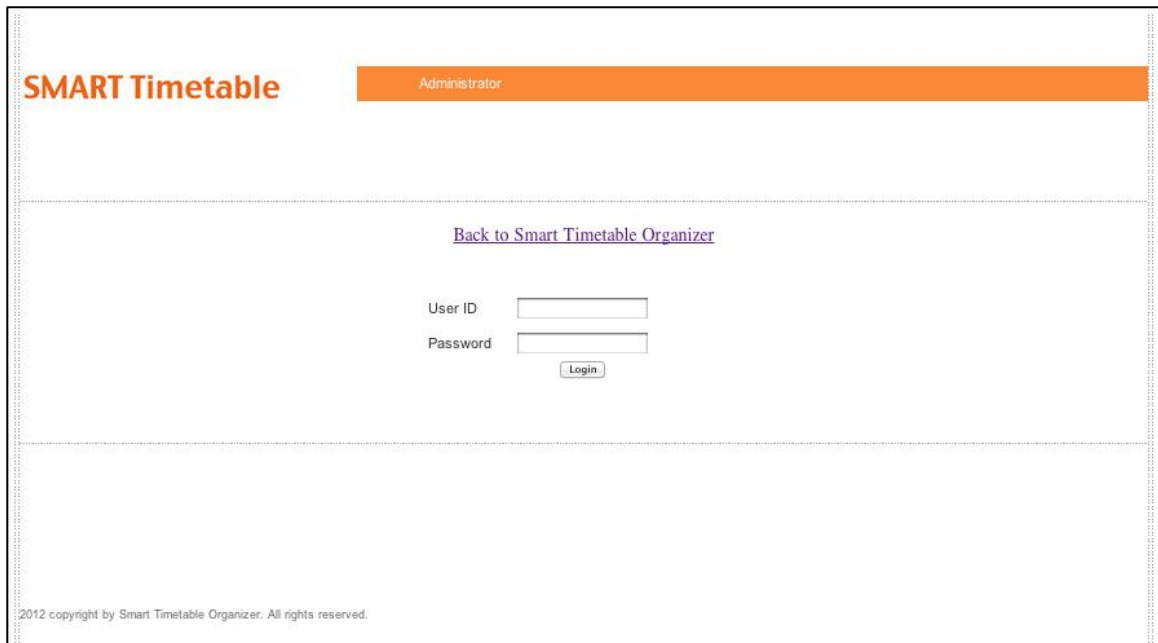
Print to PDF

Users can capture the timetable for their further use. All they have to do is click on “Convert to PDF” on the top of the page; a PDF file will be generated. That file will contain all the combinations displayed. For example, if users chose to display five combinations, the PDF file will contain those five combinations. In default mode it would print out the best three only. Users may find the downloaded timetable in their “Downloads” Folder.

4.3.5 Content Management System for Administrators

Basically, Content Management System (CMS) is a webpage for administrators of a website/system to make changes to the content or the database of that website/system. Without CMS, administrators have to go straight to the raw database structure like for example: phpMyAdmin page in order to make changes to the database. This practice is not favorable because in order to make changes straight from phpMyAdmin, users need to be familiar with the technical attributes of the system like where and how to access the page and so on. However, with CMS, we provide a simple webpage that allows the administrator to easily make changes to the database records like add, edit, or delete and view the records.

For Smart Timetable Organizer, we have developed a simple CMS as the following.



The screenshot shows a web interface for the 'SMART Timetable' system. At the top left, the logo 'SMART Timetable' is displayed. To its right, an orange horizontal bar contains the word 'Administrator'. Below this bar, there is a link that says 'Back to Smart Timetable Organizer'. In the center of the page, there is a login form consisting of two input fields: 'User ID' and 'Password'. Below the 'Password' field is a 'Login' button. At the bottom left corner of the page, there is a small copyright notice: '©2012 copyright by Smart Timetable Organizer. All rights reserved.'

Figure 4.17: Login Page for Administrators to CMS

Before being able to access and make changes to the records, administrators have to log into the system using provided username and password. There are two levels of access right, one is Full Access (Manager) and the other one is Limited Access (Operator). The

difference between these two access rights is that a Manager is able to add and delete operators, but operators cannot delete other operators or managers.

Once the administrator successfully logs in, they will be diverted to Course Page which will view all the courses stored in the database.

SMART Timetable Administration Course Group Session

Welcome, admin LOGOUT

View Course Add Course

Course

Course ID	Course Code	Course Name	Course Short Name	Action
1	SBB3013	Advanced Database Systems	ADBS	edit del
12	PAB4333	Advanced Drilling Engineering	ADE	edit del
13	MBB4433	Advanced Manufacturing Technology	AMT	edit del
15	PAB4233	Advanced Reservoir Simulation	ARS	edit del
14	PAB4213	Advanced Well Test Analysis	AWTA	edit del
164	CBB4213	Air Pollution Engineering	APE	edit del
16	TCB1063	Algorithm & Data Structure	ADS	edit del
17	MBB4213	Alternative Engery	AE	edit del

Figure 4.18: View page in CMS

As we can see in Figure 4.18, there are four main tabs: Administrator, Course, Group, and Session. These four tabs are for the four main tables in the database. When user clicks the tab, the page will view the content of that table.

On the left hand side for each tab, there is an “Add” function whereby administrators can add records to the database. When users want to add a new administrator, they have to fill in the entire details required then click “Save”. If all the information required is filled in correctly, a new record will be inserted into the database and the system will be diverted to the “view” page. However, user can always click “Cancel” to abort the adding process. An example of adding a new subject is as below.

Administration - Add

Login Name

Password

Retype Password

Email Address

Access Right

Status

Figure 4.19: Adding Course in CMS

In each record viewed, at the end of each line we can see two small buttons: “Update” and “Delete”. When user clicks on “Update” button, they will be able to make changes to the record once they hit “Save”.

Course - Update

Course Code

Course Name

Course Short Name

Figure 4.20: Updating Course in CMS

Similarly, when users hits the button “Delete”, the system will ask for confirmation before deleting, as below:

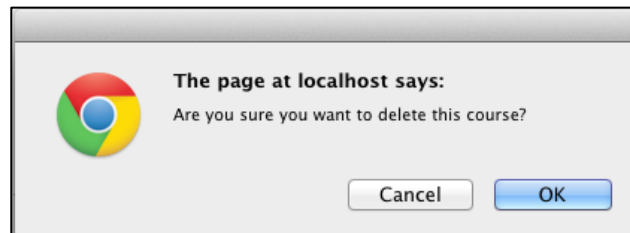


Figure 4.21: Prompt message before deleting a record

For each update and add, the system will have the error checking mechanism that will check whether the data input is of the correct format required by the database or not. If not, the system will prompt the user to input in correct format.

4.3.6 Testing

Testing is an important part of any system development. It helps to find the bugs and unwanted events as well as to realize the limitations of the system itself. For Smart Timetable Organizer, we have done testing as the following. In order to do the testing officially and easily obtain users' feedback, we have currently hosted it online at www.pluzme.net/sto/ , and that is how we make it convenience for the users to test.

- Functional Testing: is to make sure that the main functionalities of the system are working. Main functionalities of the system tested and verified are:
 - Combination generation
 - Clash checking
 - Table displaying
 - Printing to PDF
 - CMS: viewing, deletion, updating, and adding of records
- Usability Testing: is to check the level of user-friendliness of the system. We have requested some users to try the system and see how they like the system, and we can conclude that the level of usability is acceptable.

We have obtained detailed functionality testing from 21 users, which ranges from first year to final year, and both engineering and technology students. There were also a few PG students who did their undergraduate studies in UTP. The result in details for the two tests above is as the following (scale from worst to best):

No	Name	Year of Study	Functionality & User Experience				Remarks* (see below the table)
			Timetabling generation & clash checking	PDF Generation	Interface	CMS	
1	Tony Trinh	Final	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	
2	Pichleap Sok	Final	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
3	Noor Zahirah	Final	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
4	Prum Kosnak	Final	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
5	LimVicheaka	Y3	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
6	Chhim Sothea	Y3	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
7	Melissa Lee	Final	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
8	Hing Ratana	PG	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
9	Chan Piseth	PG	<input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	
10	Aris Chema	Final	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
11	Aga Prima	Final	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
12	Sry Thearith	Y1	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	
13	Chhorn Daro	Y1	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
14	TheaSokheang	Y3	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
15	Siti Maisarah	Final	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
16	Chin Shu Wei	Final	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
17	Nyoman P.	Y3	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
18	Nurudeen K.	Y3	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
19	Chanakan Keo.	Final	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
20	Lee SiewKwan	Final	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	
21	Alison Wong	Y2	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input checked="" type="checkbox"/> <input type="checkbox"/>	

Table 4.4: Functionality & user experience testing result sheet

According to the above, all of the users are satisfied with the main functionality of the system which is the timetable generation. Anyhow, for short illustration of the result, please refer to Figure 4.22, 4.23, 4.24, and 4.25.

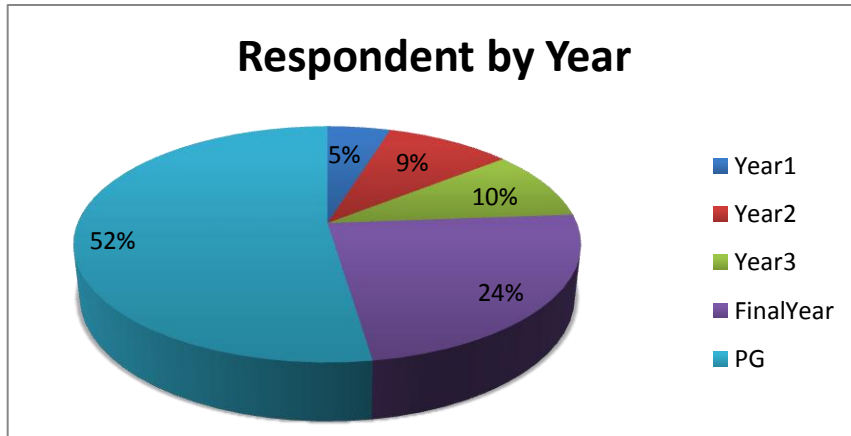


Figure 4.19: User feedback respondents sort by year of study

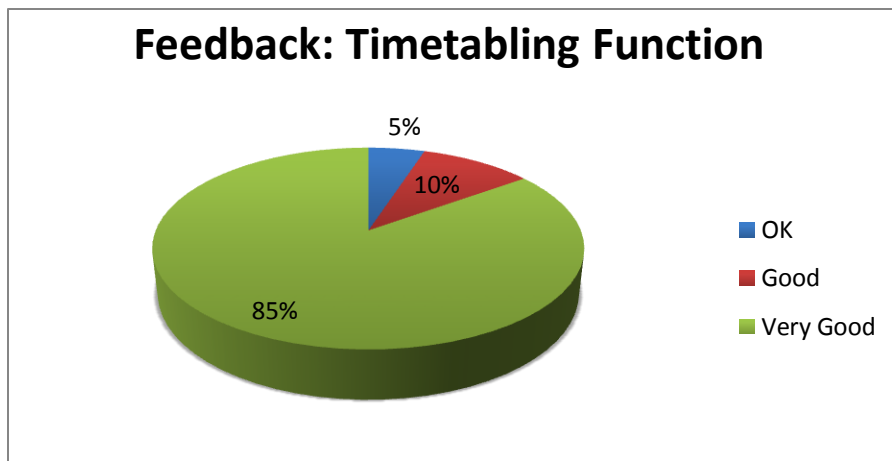


Figure 4.20: Feedback regarding timetabling function

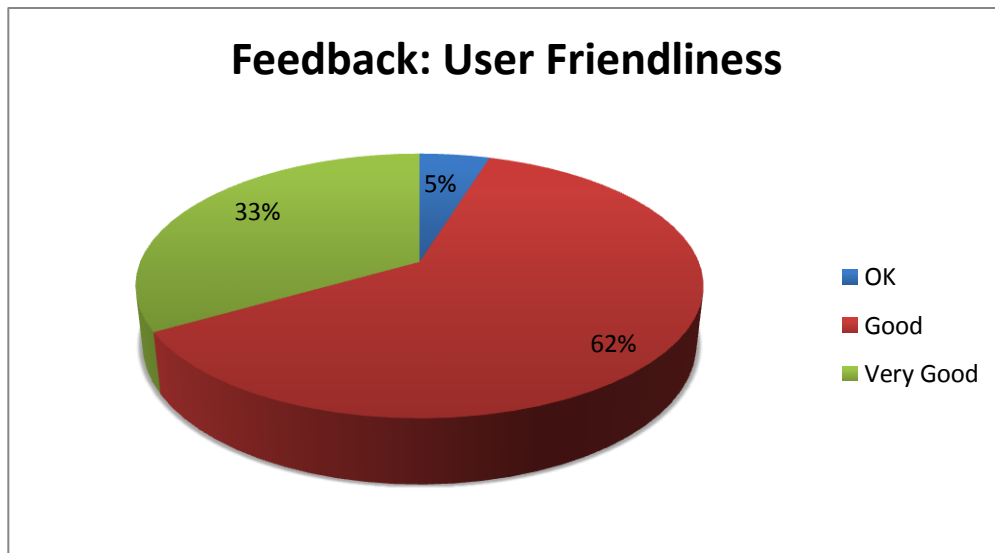


Figure 4.21: User feedback regarding user friendliness of the system

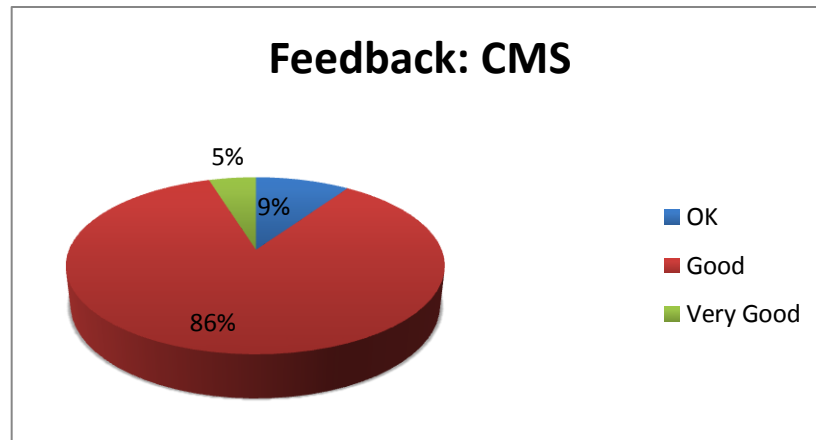


Figure 4.22: Feedback regarding Content Management System (CMS)

On the CMS side, most of them are quite okay, but they are not really concerned about that. However, in term of PDF generation and especially the interface (user experience) there were a number of Recommendations/requests as below:

No	Recommendations/Suggestions	Actions To be taken by developer	
		To be changed	To be kept in consideration
1	Please change to a better banner picture	✓	
2	Put legends for the color codes of the table	✓	
3	Should have a BACK button to go to index page	✓	
4	Print to PDF only selected combination	✓	
5	Users should be able to select current view or normal view (less clicking required and can compare differences)	✓	
6	Space between the table and the list of subjects	✓	
7	Change color code, maybe different color for each subjects		✓
8	Indicate how a combination differs from the other one		✓

9	Able to save as Excel		✓
10	Minimize clicking and scrolling in any way possible		✓
11	Disable the “Update” button in result page if cannot update (no more combination)	✓	
12	Provide default courses for students according to year, or according to programme (do not mix all subjects together)		✓
13	Put instructions how to use in home page rather than About Us page.	✓	

Table 4.5: User’s recommendations

- **Load Testing:** is to find out how many subjects that the system can arrange flawlessly. As result, the system is able to process accurately up to 8 subjects. More than that, the system will take longer time and would give errors from time to time. Note while testing is as follows. (This is summary result, since real data output produced by the system is too large)

Case				Remarks
	Course Code correctly passed?	Combinations currently generated?	Clash Found gives correct result?	
Case 1: IT/IS Subjects				
3 subjects	✓	✓	✓	Ok, Good
4 subjects	✓	✓	✓	Ok, Good
5 subjects	✓	✓	✓	Ok, Good
6 subjects	✓	✓	✓	Ok, Good
7 subjects	✓	✓	✓	Ok, Good
8 subjects	✓	✓	✓	Ok, Good
9 subjects	✓	✓	✓	Ok, but takes longer time
10 subjects	Sometimes couldn’t execute	Sometimes couldn’t execute	Sometimes couldn’t execute	Slow& sometimes error

Case 2: Engineering Subjects				
3 subjects	✓	✓	✓	Ok, Good
4 subjects	✓	✓	✓	Ok, Good
5 subjects	✓	✓	✓	Ok, Good
6 subjects	✓	✓	✓	Ok, Good
7 subjects	✓	✓	✓	Ok
8 subjects	✓	✓	✓	Ok, but takes longer time
9 subjects	Sometimes couldn't execute	Sometimes couldn't execute	Sometimes couldn't execute	Slow & sometimes error
10 subjects	Sometimes couldn't execute	Sometimes couldn't execute	Sometimes couldn't execute	Slow & sometimes error

Table 4.6: Load & System Performance Test Result Sheet

According to the result, it shows that both for Engineering and IT/IS subjects, the upper limit of subjects to ensure satisfactory performance is eight (8). Therefore, Smart Timetable Organizer uses 8 as the limit for subject selection.

CHAPTER 5

CONCLUSION AND FUTURE WORK CONTINUATION

5.1. Conclusion

Smart Timetable Organizer 1.0 is the working prototype that covers all the scopes that has been set earlier in the project initiation. It is able to automatically arrange the timetable by giving best-possible combinations upfront in a fast and accurate manner. The prototype is able to provide a PDF format for user's preferences, and also provide a basic Content Management System (CMS) to assist administrators in maintaining the system. The prototype itself is actually very close to being "implementable". We only need the input (updated data) from UTP management, some more time and configuration, then the system will be up and running.

In conclusion, the project has achieved its objectives in benefiting UTP students with a smart system that can help them to save time and effort in arranging their timetables. The system also makes it more convenient for the management as it will bring about fewer changes to be made for each semester's timetable. Since at the moment PRISM is not able to interact with Syllabus Plus yet, Smart Timetable Organizer is certainly the best possible temporary solution, in which students are strongly looking forward for the project to be implemented.

5.2. Future Work Continuation

There are some additional features that can be enhanced additional to the current features:

- a) Cater the exception subjects (Subjects with multiple groups of lectures, labs, and tutorials): as we have discussed earlier, the system left out 32 subjects that fall into this category. Therefore, some enhancements to the combination algorithm and data arrangement would be able to cater this group of subjects too, and that would make the system real useful for all students of any programme and year.

- b) Plug in data directly from Syllabus Plus and develop an auto-update schedule: currently, due to the inability to access UTP's database, we are using manually key-in data that is static. In order to implement this system and make sure that it can be used anytime later on, the management should cooperate by giving the access to UTP database so that the system will be updated real-time accordingly. Another solution would be that the management extracts the timetable into a text file (or similar format) and input to the system to process and update the database frequently, so that the system will stay updated if any changes with UTP timetabling system. All in all, in order to be implementable, we need UTP management to be cooperative and give their input. If this were to happen, an enhancement to CMS will also be required.
- c) Link to PRISM: the system can be configured to work with PRISM. In general idea, the subjects that students registered will be passed to the algorithm and students can view the possible timetables that they have.
- d) Create accounts for users: users can create accounts and save their preferred timetables, so that later on anytime they can log in and see their timetables.
- e) Mobile Apps: since nowadays smart devices are pretty much in trend, being available both in website and in mobile application would be an added advantage that ensures the usability of the system. It will provide convenience and users will be more willing to use it.

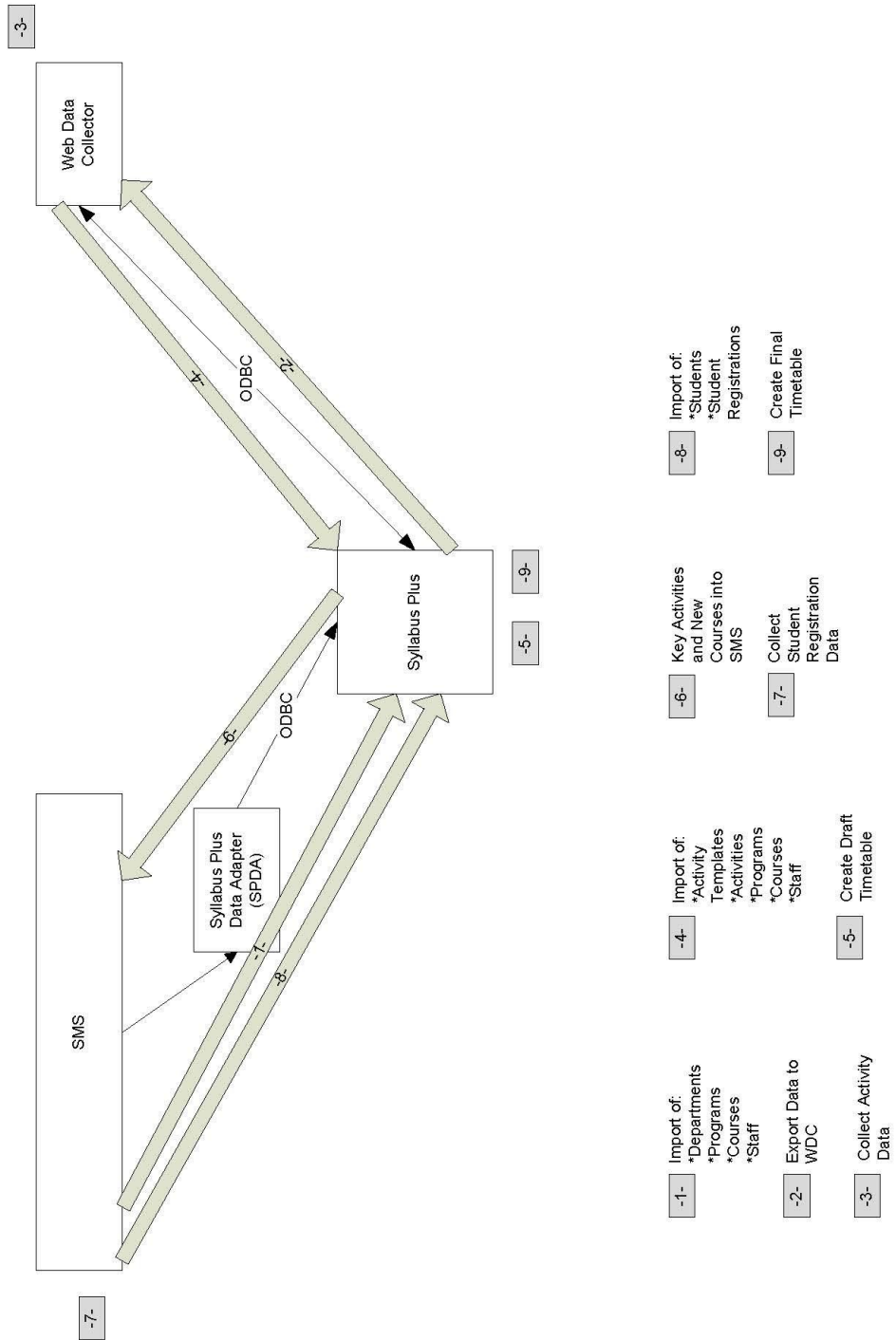
REFERENCES

- [1] Burke, E. K., McCollum, B., Meisels, A., Petrovic, S., & Qu, R. (2007). A graph-based hyper-heuristic for educational timetabling problems. *European Journal of Operational Research*, 176(1), 177-192.
- [2] Burke, E. K., & Petrovic, S. (2002). Recent research directions in automated timetabling. *European Journal of Operational Research*, 140(2), 266-280.
- [3] Padmini, M. V., & Athre, K. (2010). Efficient design of university timetable. *Electro/Information Technology (EIT), 2010 IEEE International Conference on*, pp. 1-5.
- [4] Abdullah, S., Burke, E. K., & McCollum, B. (2007). A hybrid evolutionary approach to the university course timetabling problem. *Evolutionary Computation, 2007. CEC 2007. IEEE Congress on*, pp. 1764-1768.
- [5] Abdullah, S., & Turabieh, H. (2008). Generating university course timetable using genetic algorithms and local search. *Convergence and Hybrid Information Technology, 2008. ICCIT '08. Third International Conference on*, , 1. pp. 254-260.
- [6] Sabri, M. F. M., Husin, M. H., & Chai, S. K. (2010). Development of a timetabling software using soft-computing techniques with a case study. *Computer and Automation Engineering (ICCAE), 2010 the 2nd International Conference on*, , 5. pp. 394-397.
- [7] Gunadhi, H., Anand, V. J., & Wee Yong, Y. (1996). Automated timetabling using an object-oriented scheduler. *Expert Systems with Applications*, 10(2), 243-256.
- [8] Dimopoulou, M., & Miliotis, P. (2001). Implementation of a university course and examination timetabling system. *European Journal of Operational Research*, 130(1), 202-213.
- [9] Turabieh, H., & Abdullah, S. (2011). An integrated hybrid approach to the examination timetabling problem. *Omega*, 39(6), 598-607.
- [10] Karova, M. (2004). Solving timetabling problems using genetic algorithms. *Electronics Technology: Meeting the Challenges of Electronics Technology Progress, 2004. 27th International Spring Seminar on*, 1. pp. 96-98 vol.1.

- [11] Duong Tuan Anh, Vo Hoang Tam, & Nguyen Quoc Viet Hung. (2006). Generating complete university course timetables by using local search methods. *Research, Innovation and Vision for the Future, 2006 International Conference on*, pp. 67-74.
- [12] UniTimetable <http://unitime.org/>, Retrieved on March 10th, 2012
- [13] Mimosa Timetabling System <http://mimosasoftware.com/>, Retrieved on March 13th, 2012
- [14] Wise Timetable System <http://wisetimetable.com/>, Retrieved on March 15th, 2012
- [15] Comprehensive University Timetabling System <http://www.unitime.org/>, Retrieved on February 03rd, 2012
- [16] Universiti Teknologi PETRONAS Course Timetabling Center (Module Timetable) <http://utptt.utp.edu.my/swsevenprd/Default.aspx>, Retrieved on February 15th, 2012,

APPENDIX 1

UTP – Timetabling Business Processes



APPENDIX 2

LIST OF COURSES OFFERED & TIMETABLE WORKLOAD FOR UNDERGRADUATE PROGRAMMES

UNIVERSITI TEKNOLOGI PETRONAS																				
CHEMICAL ENGINEERING DEPARTMENT																				
LIST OF COURSES OFFERED & TIMETABLE WORKLOAD FOR UNDERGRADUATE PROGRAMMES																				
JANUARY 2011 SEMESTER																				
Department : CHEMICAL ENGINEERING																				
NO	COURSE NAME	COURSE CODE	YR.	SEM	Programmes Involved							Lecturers / Coordinators / Leaders (if any)	Tutor	Method of Delivery		Spec. Req./Remarks	Expected Number of Students			
					CVE	CHE	EE	ME	PE	ICT	IBS			Lecture	Tutorial			Lab		
Structure C																				
1	Physical Chemistry	CEB 1023	1,1		x								AP Dr. Muzam Bustan		3 Lec(1+1+1) @ 2grp (70)	11(T) @ 2grp (70)	3 Lab(3) @ 2grp (45)		140	
2	Engineering Drawing	CEB 1023	1,1		x										2 Lec(1+1) @ 2grp (75)	11(T) @ 2grp (75)	4 Lab(4) @ 2grp (45)		130	
Structure B																				
1	Organic Chemistry	CEB 1013	1,1		x								Prof. Wafiqur-Rahm		3 Lec(1+1+1) @ 2grp (70)	11(T) @ 2grp (70)	3 Lab(3) @ 2grp (45)		140	
2	Chemical Engineering Fluid Mechanics	CEB 1033	1,2		x								Dr. Luluan Ismail		2 Lec(1+1) @ 2grp (75)	11(T) @ 2grp (75)	4 Lab(4) @ 2grp (45)		130	
3	Health Safety & Environment	CEB 2012	2,1		x	x	x	x	x				Dr. Azuadin-Abdullah (2 grps) / Dr. Mohamad (1 grp) / Dr. Fauziah (1 grp) / Dr. Daryng Rahayu (CT) (grps)		2 Lec(1+1) @ 4 grps(150)	12(T) (1+1+1+1+1+1+1+1+1+1+1) @ 2grps(50)			600	
Structure A																				
1	Air Pollution Engineering	CEB 4213	4		x								Dr. Lee Kok Leng		3 Lec(3) @ 1grp(40)	11(T) @ 1grp(40)	4 Lab(4) @ 1grp(40)		Block 5 (Pollution Ctrl Lab)	40

Prepared by :
 Name : Suhana bt. Sharudin
 Designation: Academic Executive
 Department: Chemical Engineering Department
 Date :

Endorsed by :
 Name : Dr. Suhaimi b. Mohd. Azir
 Designation: Head of Department
 Department: Chemical Engineering Department
 Date :

APPENDIX 3
SURVEY QUESTIONS SAMPLE

SMART TIMETABLE ORGANIZER

We aim to implement a web-based Smart Timetable Organizer, which is a website that will assist students in preparing their personal timetable. We would really appreciate if you could spend a few minutes to answer this questionnaire.

* Required

1- What university/college are you from? *

- | | | |
|-------------------------------|--------------------------------|-----------------------------|
| <input type="checkbox"/> UTP | <input type="checkbox"/> KLIUC | |
| <input type="checkbox"/> UiTM | <input type="checkbox"/> UUM | <input type="checkbox"/> UM |

2- In what year are you in? *

- | | | |
|-------------------------------------|-------------------------------------|--------------------------------------|
| <input type="checkbox"/> Foundation | <input type="checkbox"/> First Year | <input type="checkbox"/> Second Year |
| <input type="checkbox"/> Third Year | <input type="checkbox"/> Final Year | <input type="checkbox"/> Graduates |

3- Do you follow exactly the curriculum structure (what subjects to take for each semester) suggested by the university? *

- Yes, I always follow exactly the curriculum structure
- No, I sometimes drag courses from other semesters.

4- How do you prepare your personal timetable at the beginning of each semester? *

- No need to prepare anything, my university provides it in the student portal.
- I go to university website/student portal, check module timetable for each course one by one then write it down on a piece of paper or create an excel file
- I copy from my friends who are taking the same course
- Others: _____

5- How often do you come across clashes throughout your past experiences in university life? *

- Never experience any clashes
- Once in a while
- Always have clash every semester, but not much and can be fixed
- Clashes always happen, and it is very hard and troublesome to fix
- Others: _____

6- How did you deal with the clashes? *

- Try to choose/enroll in other subjects that do not clash
- Try to ask lecturers to change the slot to another time, or open new slots
- Drop the course
- Try all above
- No need to do anything, I never have any clashes

6- What do you think if the process of arranging your personal timetable is made automatic? *

- No difference for me
- Not useful for me, I'm fine with conventional way
- It would be very useful for me
- Others: _____

7- Which one would you prefer? *

- Personal Timetable that links to courses registered (PRISM or Student Portal)
- Personal Timetable that is on trial-and-error basis (not fixed, and we can add/drop courses and see different combinations of courses)

9- Any suggestions or remarks?

APPENDIX 4
LISTS OF SUBJECTS AVAILABLE IN SMART TIMETABLE
ORGANIZER DATABASE
(BASED ON MAY 2012 UG UTP TIMETABLE CENTER)

1. Advanced Drilling Engineering PAB4333
2. Advanced Manufacturing Technology MBB4433
3. Advanced Reservoir Simulation PAB4233
4. Advanced Well Test Analysis PAB4213
5. Air Pollution Engineering CBB4213
6. Algorithm and Data Structure TCB1063
7. Alternative Energy MBB4213
8. Analogue Electronics 1 ECB2113
9. Analogue Electronics 2 ECB2153
10. Badminton KSP1031
11. Basic Arabic Communication LBB2013
12. Basic Dance & Movement I KKB1031
13. Basic Dance & Movement II KKB2031
14. Basic Analysis & Petroleum System QAB3033
15. Business & Cyber Law GBB3113
16. Caklempong I KBB1051
17. Carbonate Sedimentology & Petrography QAB2033
18. Chemical Engineering Thermodynamics CCB2024
19. Circuit Theory ECB1014
20. Clastic Sedimentology & Petrography QAB2023
21. Computational Method FCM2043
22. Computer Aided Engineering Design MBB2022
23. Computer Graphics & Animation TBB4263
24. Computer Organization TCB1043
25. Concrete Technology VCB2023
26. Construction Project Management VBB4063

27. Control Systems EBB3053
28. Corporate Ethics GBB3013
29. Corrosion Engineering MBB4423
30. Cultural Anthropology HBB2013
31. Data & Computer Network EBB4023
32. Data & Information Management SCB1033
33. Data Mining & Recovery SBB4343
34. Deepwater Technology VBB4433
35. Design of Reinforced Concrete Structures VBB3023
36. Discrete Mathematics FCM1013
37. Distributed Computing TBB3023
38. Drama & Theatre I KKB1041
39. Drilling & Production Technology (GeoSc) QAB3012
40. Drilling & Production Technology PAB4533
41. Dynamics MCB2043
42. E-Business Security SBB4283
43. Economics GBB2023
44. Electronics & Energy System EBB4043
45. Electrical Machine I EBB3053
46. Electronics Machine II EBB4313
47. Electromagnetics Theory ECB2173
48. Energy Conversion & Mgmt. MBB4223
49. Engine Design MBB4113
50. Engineering Design I (CVE) VBB4084
51. Engineering Drawing (ME) MCB1023
52. Engineering Drawing (CVE) VCB1012
53. Engineering Economics & Entrepreneurship GBB3173
54. Engineering Fluid Mechanics VCB1053
55. Engineering Materials MBB3023
56. Engineers in Society VBB4043
57. Entrepreneurship I KAM1061

58. Environmental Engineering VBB2092
59. Environmental Management System (EMS) VBB4223
60. Essential of Knowledge Mgmt. SBB4363
61. Facilities Engineering PAB2073
62. Fieldwork Management and Geofield Camp QAB3044
63. Fluid Mechanics I MCB2013
64. Fluid Mechanics II MCB2053
65. Foundation & Earth Structures VBB3013
66. French I LBB2033
67. Gamelan I KKB1011
68. Gamelan II KBB2011
69. Gas Field Engineering PAB4013
70. Gas Process Engineering CBB4413
71. Geo-Environmental Engineering VBB4613
72. Geochemistry QAB2013
73. Geodynamic & Basin Development QAB4053
74. Geological Mapping Techniques QAB1023
75. Geology for Engineers VCB1033
76. Geomatics VCB1043
77. Geotechnical Engineering VCB2033
78. Health, Safety & Environment CBB2012
79. Heat Integration CBB4313
80. Heat Transfer MBB3033
81. Heat Transfer Design CCB2033
82. Highway Engineering VBB3042
83. Human Computer Interaction TCB2013
84. Hybrid Vehicles MBB4123
85. Hydraulics VCB2033
86. Hydrology VBB2063
87. Industrial Effluent Engineering CBB4233
88. Introduction to Oil & Gas Industry and Sustainable Development PCB1012

89. Introduction to Social & Political Science HBB2053
90. Introduction to Management GBB2033
91. Introduction to Petroleum Geoscience PCB1023
92. Islamic Knowledge & Sciences HBB2023
93. Islamic Studies MPW2143
94. Knowledge Mgmt Application & Measurement SBB4373
95. Mandarin LBB2023
96. Manufacturing Systems & Operations MBB3054
97. Manufacturing Technology I MCB1043
98. Mass Transfer Design CCB2053
99. Mechanical Engineering Design MBB3043
100. Mechanical Systems Design MBB4034
101. Microelectronics Physics & Devices ECB1033
102. Minerology & Petrography QAB1034
103. Modeling & Simulation for Computer-based Systems TBB4363
104. Modern Music I KKB1021
105. Modern Music II KKB2021
106. Moral Studies MPW2153
107. Multivariable Process Control CBB4523
108. Network Analysis ECB1024
109. Network Security TBB2063
110. Operating Systems TCB2023
111. Ordinary Differential Equations FCM1023
112. Paleontology & Stratigraphy QAB1044
113. Personal Finance GBB3133
114. Petroleum Economics PAB3023
115. Petroleum Eng. Thermodynamics PCB1043
116. Petroleum Exploration PAB4513
117. Physical Geology QAB1013
118. Plan Design I CBB4013
119. Plant Process Control System EBB4233

120. Plant Utility Systems CBB4323
121. Polymer Process Engineering CBB4423
122. Power Electronics I EBB3023
123. Power Electronics II EBB4323
124. Principles of Finance GCB1053
125. Principles of Marketing GBB1043
126. Process Optimization CBB4333
127. Process Plant Design CBB3024
128. Professional Communication Skills HBB2033
129. Reactor Design for Petrochemicals CBB4433
130. Real-Time Microcontroller Systems EBB4433
131. Recreation & Adventure I KAM1051
132. Reservoir Characterization & Geological Modeling QAB3043
133. Reservoir Engineering 1 PCB2023
134. Reservoir Engineering 2 PCB2053
135. Reservoir Geoscience PCB2013
136. Rock Physics & AVO Analysis QAB3023
137. Seismic Data Processing QAB4083
138. Seismic Interpretation QAB2054
139. Seismic Methods QAB2044
140. Separation Processes CCB2063
141. Signals & Systems ECB2123
142. Small Business & Entrepreneurship GBB3093
143. Soccer KSP1071
144. Software Engineering TCB2043
145. Solid Hazardous & Wastes Management CBB4243
146. Solid Waste Engineering VBB4543
147. Statics MCB1053
148. Statics and Dynamics PCB1013
149. Statistical Process Control CBB4533
150. Statistics and Empirical Method GCB2013

151. Strategic Management GBB3103
152. Structural Geology QAB1053
153. Structured Programming & Interfacing ECB1063
154. Structured Programming (ICT/BIS) TCB1013
155. System Analysis & Design SCB2053
156. Table Tennis KSP1111
157. Technical & Professional Writing LBB1013
158. Tennis KSP1131
159. Theory of Structures VCB2053
160. Transport Phenomena CBB3033
161. Transport Planning & Engineering VCB2063
162. Unit Operation Lab I CCB2092
163. Vector Calculus FCM1033
164. Virtual Reality & Augmented Reality TBB4273
165. Volume Interpretation QAB4093
166. Wastewater Engineering VBB3033
167. Well Logging & Formation Evaluation PCB2044
168. Well Stimulation Techniques PAB4323
169. Well Test Analysis PAB3013
170. Wireless Technology TBB2083
171. Advanced Database Systems SBB3013
172. E-Business SBB4263
173. International Financial Management GBB3073
174. Investment & Portfolio Management GBB3083
175. Sport Sciences KSP1011
176. Corporate Finance GBB3023
177. Bahasa Kebangsaan B MPW2123
178. Object-Oriented Programming TCB2053
179. Malaysian Studies MPW2133
180. Data Communication & Networking TCB1083
181. Web Application Development TCB1033

APPENDIX 5
TECHNICAL PAPER

