Tropical Technology Journal
Volume 17, January – December 2014
ISSN: 1656-0264

SPA DAY VERSION 3: PROPOSED UPGRADED COURSE AND CLASSROOM SCHEDULING IN CTU FOR DAY CLASSES

Jeremy P. Mondejar
Cebu Technological University-Barili Campus
e-mail: mondejar.jeremy@gmail.com
mobile phone: +639154896821 or +639472280468

ABSTRACT

Schedule Programming Assistant (SPA) for day classes was upgraded to farther reduce memory requirement and enhance its capabilities. SPADay version 3 was restructured into six (6) workbooks mainly (1) Consolidated_Schedules_Database, (2) Faculty_Profiles, (3) Subject_codes, (4) SPADay_Department, (5) SPADay_consolidated, and (6) SPADay_INS. SPA_Department can be replicated according to the number of academic departments in the campus since class scheduling in CTU is done by department. SPADay version 3, an excel template, requires Windows operating system, at least with 1 GB RAM in Microsoft Excel 2007. These have VBA macros for checking data that will cause run-time error, determining conflicts in faculty loads and room utilization, determining over-loaded and under-loaded faculty, (d) number of preparations, units, hours of classes per week, and summary of schedules. Finally, SPA-INS templates were created in a separate workbook for producing pictographic chart of schedules by teacher, schedules by section, and room utilization as final outputs in course timetabling.

Keywords: information technology, VBA Excel programming, class-faculty or course scheduling

INTRODUCTION

Class programming problems are not new. They have been studied extensively and addressed by different approaches (Schaerf, 1999, Lewis, 2008). However, in practice, “only a small number of these studies have been implemented as decision support system (DSS)” (Miranda et al., 2012). There are underlying factors as to this scenario where some are negative perceptions about the application, resistance to change particularly in adoption of new technology (Cooper & Zmud, 1990), and organization’s lack of commitment (Ginzberg, 1981).Course scheduling is a construction of “efficient and desirable timetables” and are generally classified into four (4) methods such as sequential (graph colouring), clustering, constraint-based (integer programming), and meta-heuristics (Pongcharoen et al., 2008). Majority of scheduling problems are classified as “combinatorial problems, such as communications, industrial control, operational research and production planning” (Shiau, 2011). Several methods have been applied to course scheduling problems, like heuristic algorithm (Valdes et al., 2002; Pongcharoen et al., 2008; Shiau, 2011;). Shiau (2011) and Pongcharoen, et al. (2008) proposed for meta-heuristic algorithms having a capability of allowing instructors or lecturers to do lecture based on preferences.

CTU campuses have been adapting to SPA (Schedule Programming Assistant) which has day and evening versions SPADay and SPAEve. SPADay and SPAEve older versions have been tried

in CTU campuses; yet, there were issues like failure of the system to run smoothly on some computers, users need more training on the use of the technology and more needed capabilities should be added into the system. Nevertheless, some were successful in using SPA; particularly, in CTU-Barili Campus since problems are addressed immediately by the researchers. There are advantages in using SPA in course scheduling that led to better productivity such as: working time required that significantly reduced to a single plotting of program by section and the rest of the needed outputs like program by teacher, and room utilization that were successfully automated; early and immediate detection of conflict schedules and inconsistent information were corrected; the time required for the harmonization of class-faculty schedules and room utilization was also reduced; printing of outputs for the distribution of hard copies for program by teacher, program by section, and room utilization using the corresponding required formats have been accelerated. Finally, the database that is generated in SPA can be adapted with minor revision and fine-tuning in schedules for semesters. These results are an initial step in speeding up or establishing on-time preparation of course schedules every semester.

The main objective of the study was to have a progressive improvement of SPADay by achieving the specific objectives such as: (1) make SPADay run in personal computers with Windows operating system with at least 1 GB random access memory in Microsoft Excel 2007 by (1.1) creating external references of schedule database, faculty profile, prospectus and other needed data, and (1.2) speed up macro processing time by modifying VBA codes and procedures and turning off screen update; (2) improve user-forms; (3) improve listing of items in drop-down list; (4) determine data that will cause run time error; (5)detect conflicts in room utilizations viewed together with conflicts in faculty schedules; (6) determine over-loaded or under-loaded faculty in terms of the number of preparations, student contact hours per week and, number of units; (7) improve pictographic plotting of program by teacher, program by section, and room utilization; (8) determine subject codes not found in prospectus; (9) summarize schedules by block section; and (10) revise INS forms for the program by teacher and program by section.

MATERIALS AND METHODS

A personal computer (PC), Microsoft Excel software, and class-faculty schedules for day classes were mainly the needed materials in developing SPADay version 3. Faculty profile (undergraduate to graduate programs taken, administrative designation, and other recent activities in research, extension, and production), prospectus of all degree offerings, and section advisories were the needed data to be incorporated in the program by teacher and section accordingly.

There are primarily two (3) kinds of SPADay workbooks, such as ***SPADay_Department***, ***SPADay_Consolidated***, and ***SPADay_INS*** that process respectively the schedules at department level, at consolidated level or all schedules of the campus (used in harmonizing faculty and room schedules), and in producing the final pictographic output schedules (by faculty, by section, by room) adapting the required format. From version 1, the database of schedules and other needed data are in same workbook (Figure 1). The approach in resolving the main problem that SPADay version 1 fails to run on personal computers with low specification or that basically has low random access memory (RAM) and processor is to restructure SPADay aiming on reducing

 ISSN: 1656-0264

memory requirement. Since SPADay is a workbook in Excel, reducing the size of workbooks and applying external reference for database of consolidated schedules, faculty profile and subject codes was the key to reduce memory requirement (Figure 1) in consideration that the memory requirement is directly proportional to size of workbook (Add-ins.com LLC, 2013). However, in order to reduce size of workbooks effectively, recreating SPADay from scratch was a better option.

In SPADay version 3, macro codes and procedures were modified, selecting sheet objects were avoided, conditional formats and formulas in worksheets were minimized, and screen updating is off until the code has finished executing to improve its performance (Balson, 1998).Since this is basically VBA programming, a personal computer with good specifications having licensed operating system and Microsoft office facilitated well in enhancing capabilities of SPADay. This can also enable running SPADay version 3 in a better way compared with unlicensed or starter version of Microsoft Excel.

Evaluation on the Use of SPA

Prior to further enhancing SPA, a survey was conducted in all CTU campuses to address usage problems and difficulties. Thus, after knowing the factors affecting usage of SPA, additional VBA codes and procedures were developed to enhance its capabilities applying the specific objectives mentioned above and based on suggestions and comments. Thereafter, the improved SPADay version 3 were tested using the consolidated schedules database collected from the earlier version. Testing was first done in CTU-Barili campus and after a successful test and results will be introduced on-site to all CTU campuses.

RESULTS AND DISCUSSION

Reducing the number of block section schedules from 24 to 12 (Figure 1) and creating separate workbooks for faculty profile and subject codes, the size of SPADay Department reduces by about 78 percent. Likewise, creating separate workbooks for Consolidated Schedules Database and applying external reference to same workbooks of faculty profile and subject codes, as shown in Figure 1, reduces the workbook size of SPADay Consolidated by about 83 percent. As VBA codes and procedures were also modified and screen update is turned off consequently improved macro processing time roughly by 5, 3, and 3 times faster than version 1 in processing schedule by teacher, section, and room utilization, respectively, both for SPADay Department and SPADay Consolidated.

User-forms in SPADay version 3 are designed to serve as window or dialog box. User-forms guide schedule programmers of the routinely procedures in processing the database of schedules. Schedules are stored in a ***DATABASE*** sheet where macro buttons in processing the database are labelled accordingly (Figure 4). Steps are easy to follow by the group of userforms as indicated with 1st and 2nd macro.

Adding capabilities in SPADay version 3 makes it more user friendly. First, there can be data that will cause run-time error and stop the calculation process undesirably of which the

 ISSN: 1656-0264

desired outcome of that particular macro will not be achieved. To avoid this scenario, where users got stuck and do not know what to do, a macro is added to determine data entries that may cause run-time error. A cell that is found to have error is highlighted for ease of correction.

Secondly, resolving conflicts in course schedules are the most challenging task on the part of schedule programmers. Conflicts are not only on faculty loads, as some faculty are given load from several departments, but conflicts on the use of rooms are also encountered in CTU. Thus, the newer version of SPADay can determine conflicts both faculty loads and room utilization.

Thirdly, due to the limited number of faculty, they can be loaded from several departments without a prior knowledge on the status one's faculty load. Hence, a macro that determines overloaded or under-loaded faculty (in terms of number of preparation, faculty-student contact hours per week, and number of units) is added to aid in practicing fair and balance faculty loading. CTU adapts pictographic presentation of class schedules by teacher, section, and room utilization. The pictographic presentation of class schedules is further enhanced in SPADay version 3. CTU offers a number of degree programs with several majors particularly in industrial technology. Consequently, subject codes of each degree program should be entered at subject codes database. SPADay version 3 has a macro to check whether a particular subject code(s) assigned to class schedules are found in subject codes database or not. Finally, the database of consolidated schedules can be summarized in tabulated format by block section where days with same schedules are fused.

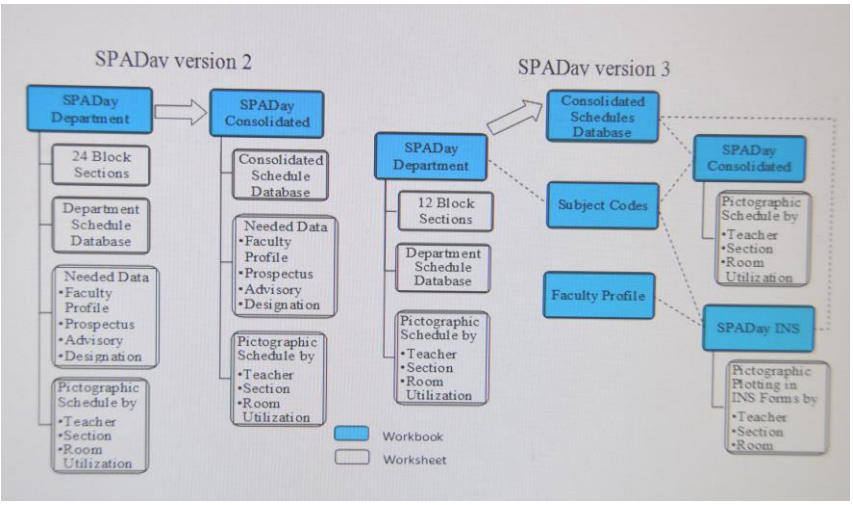

Figure 1. Restructuring SPADay

User's Manual for SPADay Version 3

The SPADay template was created using Microsoft Excel 2007 and the macro were coded with Microsoft Visual Basic for Applications. The template for ***SPADay_Department*** consists of 17 worksheets: Sched1 to Sched3, Database, Faculty1 to Faculty4, Section1 to Section4, Room1 to Room3, Subject, and day-time code. While the template for ***SPADay_Consolidated*** consists of 14 worksheets: Faculty1 to Faculty4, Section1 to Section4, Room1 to Room3, Subject, and day-time code, and SecSubFacRoom. The data of class-faculty section block schedules are entered into

 ISSN: 1656-0264

Sched1 to Sched3 worksheets of which every ***SPADay_Department*** can handle 12 block schedules. The different macros can be activated by using the buttons and userforms. The general procedure in using ***SPADay_Department*** are as follows:

Step 1. Naming list of degree offerings, faculty, and rooms
Step 2. Pictographic plotting of degree-year-section-major block schedules

2.1. Tabulate or Re-Tabulate schedules
2.2. Send tabulated schedules to the DATABASE worksheet

Step 3. Processing Schedules at DATABASE sheet.

The ***SPADay_Department*** and ***SPADay_Consolidated*** can be named or renamed to the desire of the schedule programmers from the different departments applying names that can be easily recognized. However, the ***Consolidated_Schedules_Database, Faculty_Profile*** and, ***Subject_codes*** workbooks filenames and names of worksheets should be retained to avoid failure in submitting schedules to consolidated database and getting information from these workbooks.

1. Naming list of degree offerings, faculty, and rooms

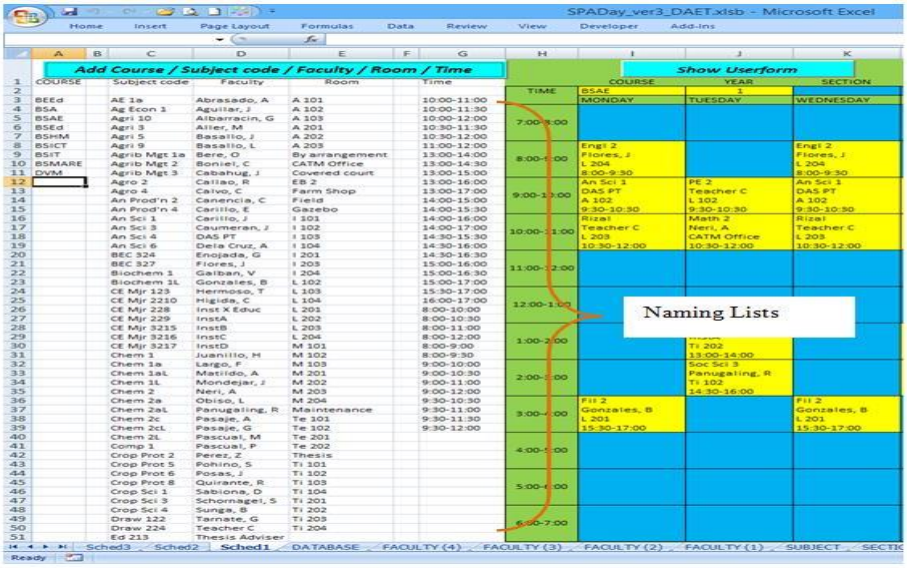

Figure 2. Naming of degree offerings, faculty, and rooms as agreed by schedule programmers

In order to be consistent with all entries, the naming of degree offerings, faculty, and rooms should be agreed upon by all departments' schedule programmers so that every entry related to a particular name will be accounted. Information should be entered first and only in Sched1 worksheet (Figure 2) before plotting the block schedules. The lists provided will become items to be selected in the drop-down list. Adding item(s) on the list can be done at the last row of the particular list, then click on ***Add Course/Subject code/Faculty/Room/Time*** button to sort list.

 ISSN: 1656-0264

Pictographic Plotting of Degree-Year-Section-Major Block Schedules.

All input data for block schedules are entered in Sched1, Sched2, and Sched3. As shown in Figure 3, information about the degree, year, section, major, and number of students are entered at the top portion of a block schedule. Ensuring valid data entry particularly on the subject code, faculty, room, and time is an important task; thus, to prevent invalid data entry in a worksheet, a drop-down list for these items are provided making the data entry go smoothly. In other words,

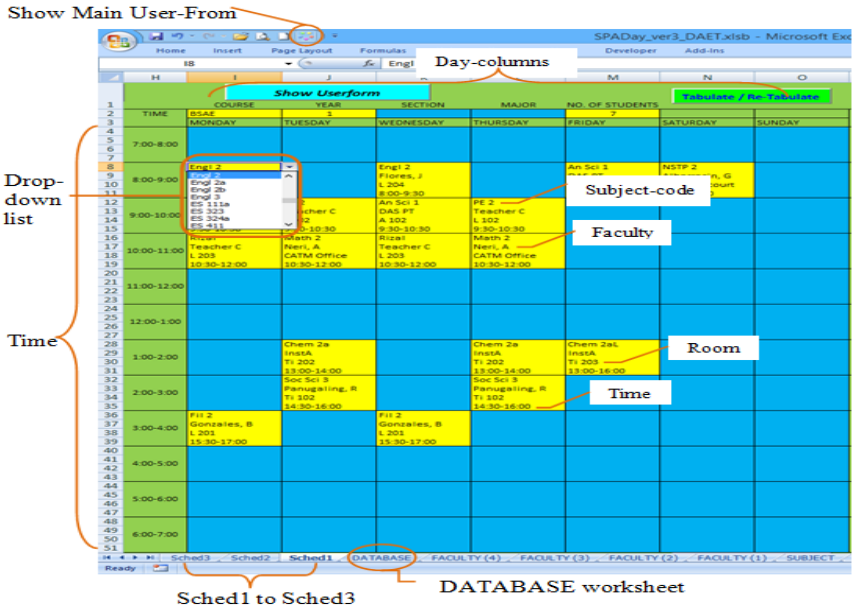

Figure 3. SPA Day Template at Schedule (Sched1, Sched2, Sched3) Worksheets

only those lists provided as agreed upon by the schedule programmers (e.g. Figure 2) are the valid choices. If a particular subject code, faculty, room or time is not among the list, such data can be added in the list applying the agreed way of naming as mentioned in *Naming list of degree offerings, faculty, and rooms*. For every subject timetabling, four (4) items should be completely entered such as subject code, faculty, room, and time entered at a corresponding day-column and time-row. In selecting which time-row, just take note of start-time. For example, if it starts at 8:00 or 8:30AM, subject timetabling should be entered at 8:00-9:00 time-row or that if it starts at 1:00 or 1:30 PM, timetabling should be at 1:00-2:00 time-row (Figure 3). Take note that SPA is adopting 24-hour format.

After pictographic plotting of degree-year-section-major block schedules, schedules have to be tabulated or re-tabulated by clicking on the ***Tabulate/Re-Tabulate*** button making them ready to be sent to DATABASE worksheet. Click on ***Show Userform*** to view the main userform (Figure 3). Click on ***To Internal DB*** to submit tabulated schedules of Sched1 to Sched3 sheets to DATABASE sheet.

Processing Schedules at DATABASE Sheet

 ISSN: 1656-0264

Having the user-form guides, the user follows the steps in processing schedules at DATABASE sheet (Figure 4a-c). The macros are grouped as 1st and 2nd as the repetitive steps follow accordingly. It is repetitive since the user can add schedules at schedule worksheets (Sched1, Sched2, Sched3) anytime and repeat the procedure from step 2 or step 1 if necessary. At this stage, the different degree-year-section-major block schedules at DATABASE sheet can now be examined in terms of errors in data entries; list of faculty, sections, rooms, and subject codes with schedules; conflicts in faculty load and room utilization; subject codes not found in ***Subject_codes*** workbook; details of faculty load in terms of class hours per week, number of preparations and units, and over-loaded or under-loaded status.

- SPADay version 3 is provided with a macro that checks data entries that may cause run-time error. Run ***Correct first: Data Entries that may cause run-time error*** to avoid stopping the calculation process undesirably of which the desired outcome of that particular macro will not be achieved.

- There is no strict sequence to follow in running the different macro buttons at the different group of macros (1st and 2nd). However, the most important macro to be performed before proceeding to the 2nd group is to click on ***Prior to running Program by Faculty, Program by Section, Room Utilization, and Subject***. This creates the list of faculty, sections, rooms, and subjects with schedules. Thus, this macro should be run again whenever schedules are added or changed to make sure that all items and schedules are accounted for accordingly.

Once data entries are corrected and list of faculty, sections, rooms, and subjects with schedules are generated; the schedules by teacher, by section, and room utilization can be plotted pictographically by choosing an item on the list and running the 2nd group of macros (Figure 4c).

The Faculty Profile and Subject Codes Workbooks

The ***Faculty_Profile*** workbook contains the faculty profile (undergraduate to graduate programs taken, administrative designation, and other recent activities in research, extension, and production), designation and advisory assignment, whereas the ***Subject_codes*** contains subject codes with their corresponding descriptive titles and number of units.

 ISSN: 1656-0264

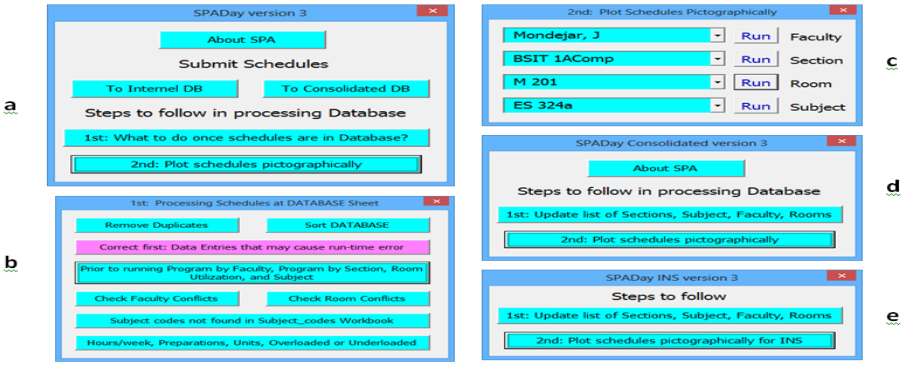

Figure 4. SPA Day version 3 User-forms

The Consolidated Schedules Database Workbook

The Consolidate Schedules Database workbook contains all the block schedules of all degree programs offered by the campus. The procedure of using this macro enabled workbook is exactly the same as processing schedules at DATABASE sheet of ***SPADay Department*** as explained above.

Using SPADay Consolidated

There is only a little difference in using ***SPADay_Consolidated*** compared to ***SPADay_Department***. ***SPADay_Consolidated*** processes the data of ***Consolidate_Schedules_Database*** workbook. The most important macro to be performed before proceeding to the 2ndgroups is to click on ***1st: Update list of Sections, Subject, Faculty, Rooms***(Figure 4d) button and the rest is the same as ***SPADay_Department***.

Using SPADay INS

Consequently, the final output of SPADay, are the standard instructional form adapted by CTU (INSFAC, INSSEC, and INSROOM) can be obtained by running macros of ***SPADay_INS***. Whichever is processed at FACULTY, SECTION, and ROOM worksheets are the corresponding items to be processed at INSFAC, INSSEC, and INSROOM, respectively as shown in Figure 5. The user has the option to use the userforms or directly use the combo list and macro buttons on FACULTY, SECTION, ROOM, INSFAC, INSSEC, and INSROOM worksheets. For INSFAC, INSSEC, and INSROOM just click on ***Update Load and Faculty Details, Update Section Load,*** and ***Update Room Utilization***, respectively.

 ISSN: 1656-0264

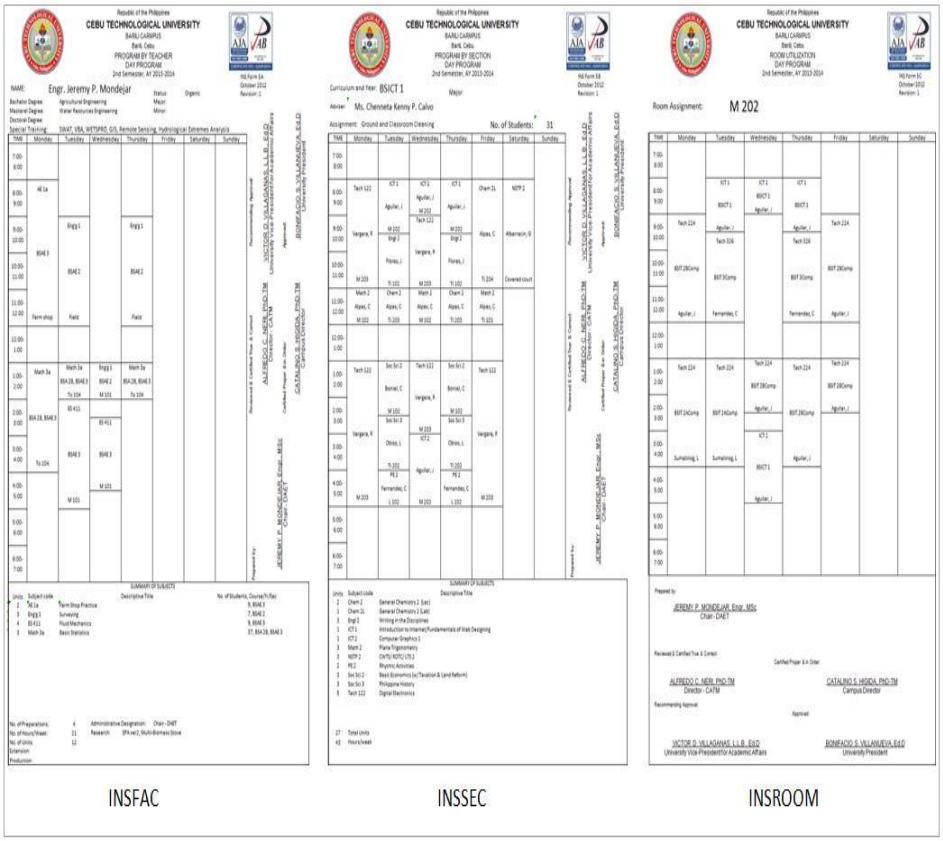

Figure 5. Samples of Proposed Revision of Instructional Forms in CTU for Day Classes (INSFAC, INSSEC, and INSROOM)

CONCLUSIONS

This study has significantly improved the VBA codes and procedures, and the structure of SPADay making it able to run in computers with Windows operating system, at least with 1 GB memory having Microsoft Excel 2007, consequently, increasing its efficiency and performance. However, with the advent of high-end personal computers nowadays, SPADay could run without difficulties. SPADay has been very important application software in speeding up the generation of class-faculty schedules every semester initiated at CTU-Barili campus while it is also introduced to all other CTU campuses since 2012. This enhanced version will further facilitate class-faculty schedule programmers and boost their interest on using this application software. SPADay has offered to make a stop on purely manual class-faculty scheduling or course timetabling while further research is needed by introducing an option of fully automated class programming; particularly, in resolving conflict schedules.

ACKNOWLEDGMENT

The author acknowledges Cebu Technological University Research Office for providing financial support during the conduct of this research.

 ISSN: 1656-0264

LITERATURE CITED

Add-ins.com LLC. (2013). What to do if you get an out of memory or not enough resources message. Retrieved September 26, 2013, from Add-Ins.com LLC: http://www.add-ins.com/support/out-of-memory-or-not-enough-resource-problem-with-microsoft-excel.htm

Balson, D. (1998, April). Excel VBA: Efficiency and Performance. Retrieved June 2, 2013, from Australian Visual Developers Forum Archive (ADVF): The Magazine for Visual Programmers: http://www.avdf.com/apr98/art_ot003.html

Boland, N., Hughes, B. D., Merlot, L. T., & Stuckey, P. J. (2008). New Integer linear programming approaches for course timetabling. Computers & Operations Research, 35(7), 2209-2233. doi:10.1016/j.cor.2006.10.016.

Cooper, R. B., & Zmud, R. W. (1990). Information Technology Implementation Research: A Technological Diffusion Approach. Management Science, 36(2) 123-139. doi: 10.1287/mnsc.36.2.123.

Ginzberg, M. J. (1981). Early Diagnosis of MIS Implementation Failure: Promising Results and Unanswered Questions. Management Science, 27(4) 459-478.

Lewis, R. (2008). A survey of metaheuristic-based techniques for University Timetabling problems. OR Spectrum, 30(1), 167-190. doi: 10.1007/s00291-007-0097-0 .

Miranda, J., Rey, P. A., & Robles, J. M. (2012). udpSkeduler: A Web architecture based decision support system for course and classroom scheduling. Decision Support System, 52(2), 505-513. doi:10.1016/j.dss.2011.10.011.

Pongcharoen, P., Promtet, W., Yenradee, P., & Hicks, C. (2008). Stochastic Optimisation Timetabling Tool for university scheduling. International Journal of Production Economics, 112(2), 903-918. doi:10.1016/j.ijpe.2007.07.009.

Schaerf, A. (1999). A survey of automated timetabling, Artificcial Intelligence Review. Artificial Intelligence Review, 13(2) 87-127. doi: 10.1023/A:1006576209967.

Shiau, D. F. (2011). A hybrid particle swarm optimization for a university course scheduling problem with flexible preferences. Expert Systems with Applications, 38(1), 235-248. doi:10.1016/j.eswa.2010.06.051.

Valdes, R. A., Crespo, E., & Tamarit, J. M. (2002). Design and implementation of a course scheduling system using Tabu Search. European Journal of Operation Research, 137(3), 512-523.