International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:7, No:11, 2013

iCCS: Development of a Mobile Web-Based Student Integrated Information System Using Hill Climbing Algorithm

Maria Cecilia G. Cantos, Lorena W. Rabago, Bartolome T. Tanguilig III

***Abstract*—**This paper describes a conducive and structured information exchange environment for the students of the College of Computer Studies in Manuel S. Enverga University Foundation in. The system was developed to help the students to check their academic result, manage profile, make self-enlistment and assist the students to manage their academic status that can be viewed also in mobile phones. Developing class schedules in a traditional way is a long process that involves making many numbers of choices. With Hill Climbing Algorithm, however, the process of class scheduling, particularly with regards to courses to be taken by the student aligned with the curriculum, can perform these processes and end up with an optimum solution. The proponent used Rapid Application Development (RAD) for the system development method. The proponent also used the PHP as the programming language and MySQL as the database.

***Keywords*—**Hill climbing algorithm, integrated system, mobile web-based, student information system.

I. Introduction

STUDENT Integrated Information System is now a facility whereby universities and colleges manage the records for their students. The convenience of accessing the educational resources online makes the programs ideal for working professionals and students alike. With the widespread of distance learning education program, even most conventional colleges and universities are now offering online education. Along with certificate, associates and bachelor's degree, master's degree, and doctoral degree are what most online. The main advantage is to be able to have a place for the students in the appropriate instructional environment, which is in an online application.

Furthermore, student advising is an essential component of a successful academic experience. Academic advisors are exposed to a variety of opportunities, enhancements, problems, and choices as technology becomes more prevalent on university campuses [1]. Various universities and institutions around the world use automated advising systems. They are helpful and beneficial for both advisors and advisees in that they contribute to assisting in making better-informed decisions and improved services [2].

Mobile computing is a versatile and potentially strategic technology that improves information quality and accessibility, increases operational efficiency, and enhances management effectiveness [3]. This is because of its capability to enable users to remain connected while on the move. High end users can opt for satellite based networking which provides wireless connectivity anywhere in the world [4].

Aware that in today's trend, quality education with essential systems being used is a necessity. Since, Manuel S. Enverga University Foundation has already an online student information portal whereas viewing of grades is the main focus of the system and can be accessed only through intranet. Thus, the proponent influenced to look into the workability of developing a mobile web-based student integrated information system that would give the students an easy and speedy access through internet technology and use of mobile phones to a range of handy and important information about their school activities, program curriculum, enlistment and grades.

II. Objectives and Scope of the System

A. Objectives

The main objective of this paper is to develop a Mobile Web-Based Student Integrated Information System for the College of Computer Studies in Manuel S. Enverga University Foundation, Lucena City, Philippines that will be useful to the students to check academic result, manage profile, view grades and manage self-enlistment. Moreover, the system will can be viewed in mobile phones. The specific objectives of the study are the following:

1. To analyze the existing student information portal to determine the functional requirements and overall architecture that will serve as a basis in designing the specifications of the mobile web-based system through fact-finding techniques such as observation, interview and research;
2. To design and develop a web-based student integrated information system and can be viewed through mobile phones; and
3. To test the system if it can be viewed to a mobile phones and whether it could be run in any platform.

B. Significance of Developing the System

Improvement is a word that entails responsibility and achievement. Responsibility does not stop in introducing, but

Maria Cecilia G. Cantos is an Associate Professor IV in Manuel S. Enverga University Foundation, Lucena City, Quezon Province, Philippines. She is also a DIT student in Technological Institute of the Philippines, Quezon City, Manila, Philippines (phone: +63423734027, +639294564307; e-mail: mbgranada@yahoo.com).

Lorena W. Rabago is the department chair of Information Technology in Technological Institute of the Philippines, Quezon City, Manila, Philippines.

Bartolome T. Tanguilig III is the dean of the Information Technology Education department of Technological Institute of the Philippines, Quezon City, Manila, Philippines.

requires continuous enhancement so that the environment where they belong becomes challenging. This paper serves a development and improvement in the lives of the people concerned and a step-forward for the continuous growth of the university. Basically, it is because learning for everyone is not confined to a manual process, this innovation may even give a way to the global competitiveness of College of Computer Studies in Manuel S. Enverga University, Lucena City, Philippines. Competitiveness is not only the major concern, but this also introduce cutting of expenses without sacrificing the quality of services offered to the students. Finally, matters concerning students can be easily compared and problems that may arise can be given urgent answer.

C. System's Architecture

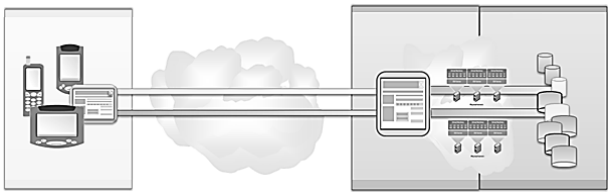

Fig. 1 System architecture of the system

Fig. 1 presents how the hardware and the software components are interrelated and dependent between each component.

The user will browse the site on any browser. After which the browser will be able to connect through http into the web server. After the web browser applied to connect to its server, all data will be delivered from database server into the server to be viewed on the browser. In a web application, there are nodes for the client/user, web server, application server and database server. The components running in the client/user are the web browsers. The web server runs all the PHP pages, codes and scripts. The application server contains all the transactional components and owns the bulk of the logic. The database node might also execute some key stored procedures, but mostly runs the database code. The web server serves as the center of the system by sending in and out data, both in database server and in the mobile phone.

D. System's Features

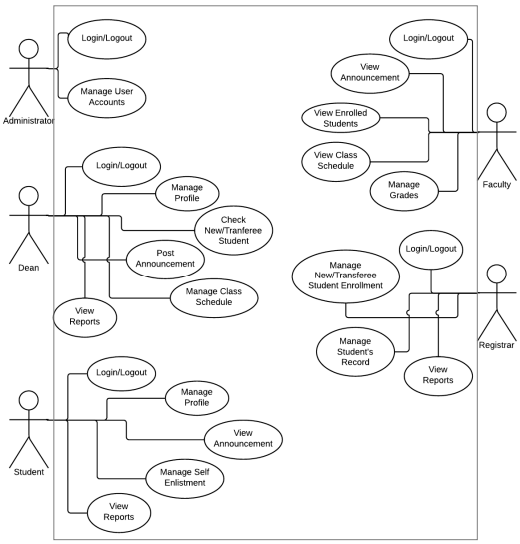

Fig. 2 Use case diagram of the system

Fig. 2 shows the UML diagram of the system. This presents the functionalities of the system and the system's requirements from the user's perspective. The system has five actors: the administrator, the dean, registrar, faculty member, and student.

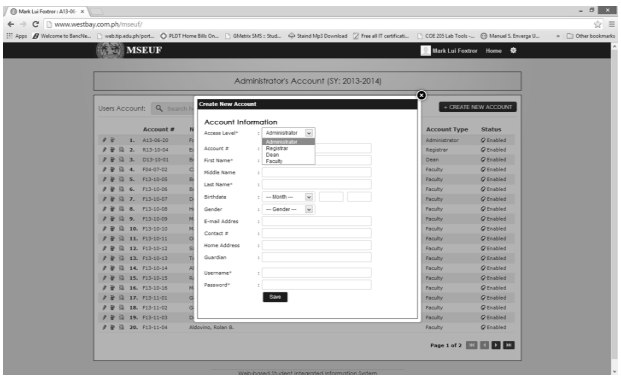

Fig. 3 Administrator's portal

The administrator, as shown in Fig. 3, oversees the whole operation of the system. The administrator manages the user account of the would-be-users, where the administrator can add, edit, delete, and view user accounts as administrator, registrar, dean and faculty.

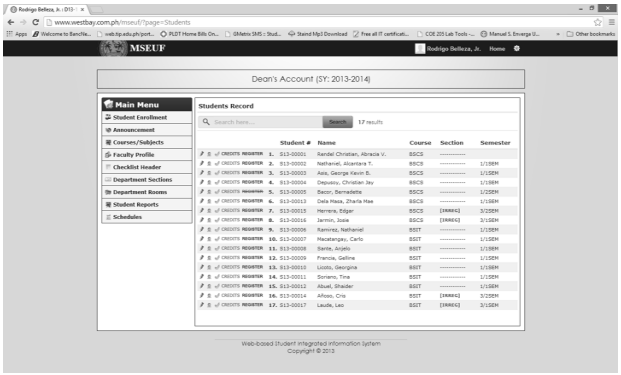

Fig. 4 Dean's portal

Fig. 4 illustrates the dean's portal in which the dean can manage profile (i.e. changing password, change pictures, etc.). He can also check for the registered/enrolled student coming from the registrar if the student is a transferee/new student; post announcements that can be viewed by the students and faculty members; manage program curriculum; create faculty schedule according to the specialization and courses currently offering to the current academic year and semester. And, he can also view student's report, linked to the registrar's office, such as grades in a particular semester and evaluation/checklist sheet of the student.

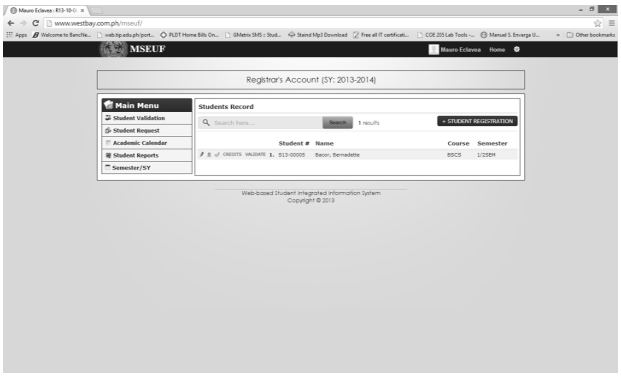

Fig. 5 Registrar's account portal

Fig. 5 shown above is the registrar's page. The registrar is responsible for new/transferee student enrollment/registration. He can also use the system to validate student registration sent by the dean. He can also manage student's request such as: changing/adding/dropping courses of the student. He can also post the activities that are related to academic; activate current semester and current academic school year; manage student's records like viewing the grades of the students and the curriculum of the different courses offered by the college and view student information; and view the evaluation checklist of the students where he can monitor the status of the students' grades from the student's previously taken and currently enrolled courses.

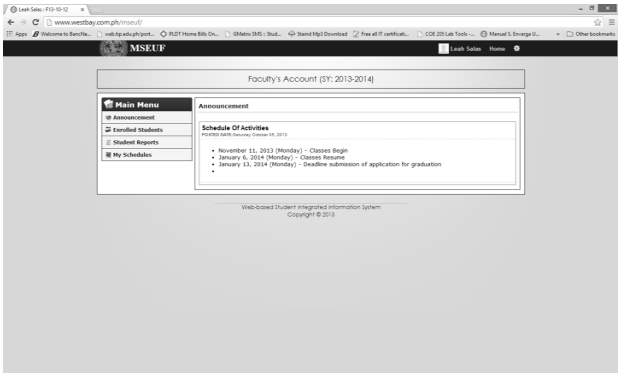

Fig. 6 Faculty account portal

The faculty account, as shown in Fig. 6, can also access the system where he/she can view announcements from the registrar or dean; view enrolled students with a corresponding class schedule; view class schedules plotted by the dean; and manage grades of the students.

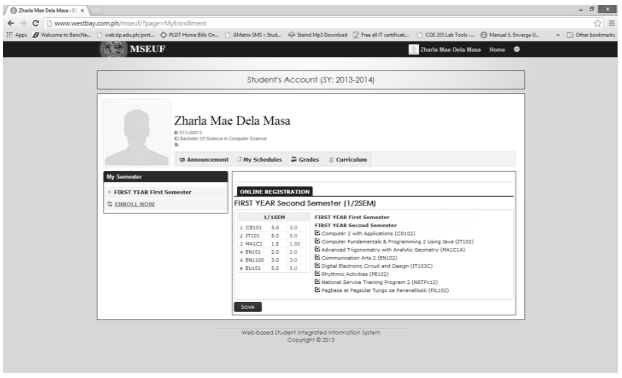

Fig. 7 Student's account portal

Fig. 7 illustrates the student's account. The student is the main user of this system. By using the system, the student can also manage profile (i.e. changing password, change pictures, etc.).He/she can view announcements, class schedule, grades and program curriculum. Student can also monitor their evaluation/checklist for them to know which course will be taken for the next semester or what courses are taken already. The system will automatically determine the year level of the student according to their credited courses or courses taken. If the student is a regular student, the class schedule of a particular block section will be displayed automatically. If the student is an irregular student, the system should check first the following:

1. Pre-requisite or co-requisite check: if the course has any pre-requisite and co-requisite from the curriculum; and whether the student passed the requisite. If the student doesn't, he/she will not be able to take such course.
2. Course is not offered anymore check: whether the course is open in the student's wanted semester. If not, the student will not be able to add such course.
3. Credit course check: if the course will be taken is credited or not.

The student can also manage their own enlistment in which the student will see the courses offered that are aligned to their program curriculum. The student can also send request to the registrar for approval with the following cases: when student wants to add/drop a particular course/s.

III. Related Works

An integration of inter-connectivity of information system (i3) in University of Sindh (UoS) was developed for sharing and exchanging all kind of information mostly associated with student in different departments, and thus each department can easily and quickly find the status of student for required results. They have solved many problems frequently occurred in university information system [5].

Furthermore, the research paper of an MSCS student developed and evaluated a mobile system providing user-centric information services for Norwegian University of Science and Technology (NTNU).According to the results, most of the participants strongly agreed to a mobile version of the time tables that are extremely useful [6].

Also, a WAP-Based Students Information System was developed to help the students and lecturers on campus to find and access information based on ad-hoc basic, which is of interest and relevant to students or lecturers; they only need a PDA or a mobile phone [7].

All of these related works have a common goal and that is ensuring that the students can access information at any time, at any locations. A fast, accurate, fair and objective student integrated information system will definitely help students to check their academic result and related courses information, view the own personal information, check for announcement, and even courses enlistment in a handy manner. With these existing applications in the market and with the works reviewed, the proponent drew the idea of proposing a mobile-based student integrated information system.

The problem of optimal system configuration for Web application servers wherein they formulated the problem of finding an optimal configuration for a given set of applications as a black-box optimization problem was studied. Then a Smart Hill-Climbing algorithm was proposed by using ideas of importance sampling and Latin Hypercube Sampling. Hence, the algorithm was efficient in both searching and random sampling. It was consisted of estimating a local function, and then, Hill-Climbing in the steepest descent direction [8]. Moreover, hill climbing as a mapping technique was only useful for relatively small graphs. The results of the paper illustrated that hill climbing comes within a factor of two of optimum for the moderate size graphs considered. They also stated that performance of hill climbing decreases with increasing graph size. They added that any state space search technique will probably suffer from large running times due to the size of the state space [9]. From the simulation results they had made DDFD join query that hill climbing algorithms was effective for delegating a query to a vertex where lower overall network traffic was incurred when the evaluation queries were retrieved in the distribution of data. Furthermore, they also stated that cost of performing the hill climbing algorithm was low compared to determining the global optimum location and delivered a large proportion of the delegation benefit [10].

Developing schedules of a particular student in traditional way is a long process that involves making many numbers of choices. It takes a long time and effort to accomplish. With Hill Climbing Algorithm, however, the process of class scheduling of a student can perform these processes and end up with an optimum solution.

IV. Hill Climbing Algorithm

Hill climbing algorithm is simply a loop that continuously moves in the direction of increasing value, which is uphill. It stops when it reaches a "peak" where no neighbor has a higher value. The algorithm does not maintain a search tree, so the current node data structure need only record the state and its objective function value [11]. In this algorithm, only a local state is considered when making a decision of which node is to expand next; when a node is entered all of its successors nodes have a heuristic function applied to them; the successor node with the most desirable result is chosen for traversal.

The Hill Climbing algorithm is an optimization problem that can usually also be modeled as a search problem; searching for the optimum solution from among the solution space. The hill climbing technique of search is as follow [12]:

1. Start with an initial solution, also called the starting point. Set current point as the starting point.
2. Make a move to a next solution, called the move operation.
3. If the move is a good move, then set the new point as the current point and repeat (2). If the move is a bad move, terminate. The last current solution is the possible optimum solution.

The proponent adopted this search technique for the system particularly in creating self-enlistment and finding the best set of courses to the class schedule, since this is the major feature of the system, for the student to take course in a convenient way especially for the student in an irregular year status.

Fig. 8 depicts a complete graph of self-enlistment process of a student who finishes the program:

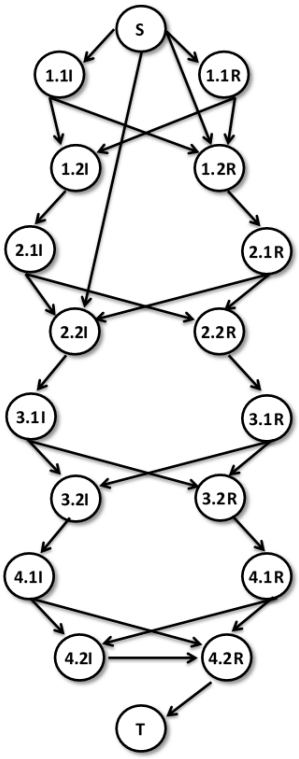

Fig. 8 Self-enlistment process

The starting point of the process is when the student will register/enlist courses that correspond to his/her program's curriculum. Each year and semester is referring to set of courses that need to be completed before going to the next node or neighbor until it reaches the last node, the peak of the hill, which is to complete all courses according to the curriculum. Assume that *S* node is a student; *I* and *R* nodes refer to a set of courses completed per semester and per year level; and *T* node means that the student completed all the courses according to his/her program.

Let say, a student (*S node*) is a freshman student, first semester, and he/she will register courses. Definitely, the freshman student is a regular student, which is the *1.1Rnode*, therefore, he/she will register a block of section. Then, if the freshman student finishes the first semester and passed his/her registered set of courses, then he/she will automatically register the second semester's block section that is the *1.2Rnode*. Otherwise, the student will take a broken schedule only if he/she fails to complete the courses in the particular semester regardless of year level, the *1.1Inode*, which means, the student will become an irregular student and then it will continue to *1.2Inode*. Hence, the movement of the graph will vary through checking of the courses to the following instances: (a) pre-requisite/co-requisite; (b) course offering; and (c) passed/failed automatically be an irregular student. The status in terms of year level of a transferee student may vary according to the credited courses aligned to the curriculum. From the *S node*, the path can be in *1.1Inode*, or in *1.2Rnode,* or in *2.1Inode*, or in *2.2Rnode* depending on the credited courses; therefore, if the student will be any of these nodes, then, a broken schedule will be applied for the student. The student must reach the *T node* that indicates all courses are completely done.

V. System Methodology

Rapid Application Development (RAD) was used as the system development method. The proponent utilized Rapid Application Development approach in order to ensure the proponent that the system is really the users' need. Rapid Application Development (RAD) is a development lifecycle designed to give much faster development and higher-quality results than those achieved with the traditional lifecycle. It is designed to take the maximum advantage of powerful development software.

RAD compresses the step-by-step development of conventional methods into an iterative process. The RAD approach thus includes developing and refining the data models, process models, and prototype in parallel using an iterative process. User requirements are refined, a solution is designed, solution is prototyped, the prototype is reviewed, user input is provided, and the process begins again (http://www.casemaker.com/download/products/totem/rad_wp.pdf, n.d.)

A. Models, Technologies, Tools and Technique Used

The following are the models, technologies, tools and techniques that the proponent used in the construction of the mobile web-based system.

The proponent used Unified Modeling Language (UML) in the illustration of the system to help them visualize the functions, flow and the components that will compose the system. These diagrams were used to illustrate the overview and functional requirements of the system.

PHP or Hypertext Preprocessor, a widely-used general-purpose scripting language especially suited for Web development that could be embedded into HTML [13], was utilized to develop a dynamic and functional web-based system to support and serve the mobile web-based system. Thus, the proponent utilized the PHP as the programming language and MySQL as the database.

In developing the system, the proponent used Adobe Dreamweaver as the Integrated Environment Development (IDE) which is very effective when developing using PHP and MySQL. The proponent utilized the Adobe Photoshop CS4 for designing the interface of the web-based part of the proposed system and for editing the images and MySQL for the database management system.

Prototyping means undergoing a series of trials and errors while creating a model of the system. Prototyping is used in almost every development project [14]. Rapid prototyping involve multiple iterations of a three-step process: prototype, review and refine. Prototype process converts the description of the solution into mocks-ups, factoring I user experience standards and best practices. Review involves sharing the prototype with users and evaluate whether it meets their needs and expectations. Refining was based on feedbacks on areas

that need to be clarified [15]. Furthermore, prototyping was used as a software development technique that emphasizes short development time. Also, the fact-finding techniques were used in the data gathering.

VI. Testing

The system's testing process was done by the proponent before uploading the system to the internet. The process was done in different processor, operating system and different mobile platforms.

The system was tested using a Dell Inspiron 640mlaptop in Windows 8 pro operating system with Intel Core 2 CPU and an Acer Aspire V3-471G laptop with Intel Core i7-3610QMprocessor with Linux operating system. Moreover, the system was tested using Samsung Galaxy Tab P1000 with 2.3x version of android operating system and Apple Ipad4 with IOS7 operating system. The testing was started in checking of the compatibility to the platform and hardware that will be utilized in using the system.

The process of testing to different processor, operating system and different mobile platforms was successful in terms of the compatibility of the system. But, however, the resolution of the mobile phone was the encountered error in the system.

VII. Conclusion

With the development of the system, the proponent was able to accomplish the main objective that is to develop a system that will perform student information management through internet technology and in a handy manner using mobile phones.

Moreover, hill climbing algorithm was very operational in creating class schedule and finding the best path for a course corresponds to program's curriculum. With this, the system provides a very useful tool for the students in terms of self-enlistment procedure.

Future developments to enhance the functionality of the system will be forecasting of number of courses to take and that the system will be a mobile application.

References

[1] Steele, G., Leonard, M., Haberle, C., &Lipschultz, W. (2000). Technology and Academic Advising. Academic Advising News, 21.

[2] Murray, S., & Le Blanc L. (1995). A decision support system for academic advising. Paper presented at the 1995 ACM symposium on Applied computing, February 26-28, Nashville, TN, USA.

[3] Zimmerman, J. B. (1999). Mobile computing: Characteristics, business benefits, and the mobile framework. Retrieved from http://ac-support.europe.umuc.edu/~meinkej/inss690/zimmerman/INSS%20690%20CC%20%Mobile%20Computing.htm.

[4] Romando, R. (2007, March 28). What is mobile computing?.Ezine @rticles.com. Retrieved from http://ezinearticles.com/?What-is-Mobile-Computing?&id=408148.

[5] Chandio, A. A., Zhu, D. &Sodhro, A. H. (2012). Integration of inter-connectivity of information system (i3) using web services. Retrieved from http://www.iaeng.org/publication/IMECS2012/IMECS2012_pp651-655.pdf.

[6] Moe, S. P., "Design and evaluation of s user-centric information system (master's thesis)," unpublished.

[7] Ismail, M. N. (2009). Development of WAP based students information system in campus environment. Retrieved from http://www.ijcte.org/papers/043.pdf.

[8] Xi, B. et al. (2004).A smart hill-climbing algorithm for application server configuration. Paper presented at the WWW2004, May 17–22, 2004, New York, New York, USA.

[9] Stone, P. D. et al. (n.d.) Query delegation and hill climbing algorithms for DDFD join queries. Sponsored by US Army Research Laboratory and the UK Ministry of Defense.

[10] Freeman, J.A. &Skapura, D. M., *Neural networks* (Algorithms applications and programming techniques). Addison Wesley Publishing Company, 2006, pp. 223-226.

[11] Russell, S. & Norvig, P. (2006). Artificial intelligence a modern approach.

[12] Amit (2010, March 14). Hill Climbing: A simple optimization method [Web blog message]. Retrieved from http://echorand.me/2010/03/14/hill-climbing-a-simple-optimization-method/.

[13] PHP.NET (2008). PHP hypertext processor. Retrieved from http://www.php.net.

[14] Satzinger, J. W., Jackson, R. & Burd, S. (2004). *Systems Analysis and Design*. Course Technology.

[15] Cerejo, L. (2010, June 16). Design better and faster with rapid prototyping. Retrieved from http://www.smashingmagazine.com/2010/06/16/design-better-faster-with-rapid-prototyping/.