

Developing Learning Management System in Sticks

Dave E. Marcial¹, Mark B. Onte², Janice Antoniette V. Forster³, Jonathan Mark N. Te⁴,
Maria Lourdes S. Curativo⁵

^{1, 3, 4, 5}College of Computer Studies, Silliman University, Dumaguete City, Negros Oriental, Philippines

²Management of Information Systems, Silliman University, Dumaguete City, Negros Oriental, Philippines

Email address: ¹demarcial[AT]su.edu.ph

Abstract—This article describes a learning management system in a stick. Specifically, it discusses the development of a handy learning management system, called PLMS, which can run on a USB flash drive without an internet connection. PLMS is an information system that is transferable to multiple environments and manageable to a variety of infrastructure specifications using any USB flash drive. The study employed a customized development model where it combined traditional and non-traditional development models with the emphasis on a feature-driven framework. Development tools include Poodle, Sublime Text, and Fiddler. The quality of the system was tested using ISO 9126. The result shows that the quality of PLMS is very high. To successfully integrate PLMS in the classroom, teachers should choose any of the proposed implementation schemes in terms of the level of integration, mode of integration, and a terminal of integration.

Keywords—Learning Management System, e-Learning, Information Systems, Mobile Learning.

I. INTRODUCTION

In the last decade, there has been an enormous acceleration in the use of learning management system (LMS) for teaching and knowledge sharing dissemination across the Web [1]. LMS was featured in UNESCO's ICT in Education online resources. An LMS enables teachers to manage large numbers of students, instructors, and courses and provide online forums [2]. A robust LMS should be able to unify and automate system administration, utilize self-service and self-guided services, assemble and deliver learning content rapidly, consolidate training initiatives on a scalable web-based platform, support portability, and standards, and personalize content and enable knowledge reuse [3]. All of these features can be found on Moodle, short for Modular Object-Oriented Dynamic Learning Environment; it is a free learning management system that has become very popular among educators around the world as a tool for creating dynamic web sites for their students. It is web-based and can be used in large and multiple classes that are either fully online courses or blended learning infused with the social constructionist tradition and standard Sharable Content Object Reference Model, known as SCORM, packages. According to Moodle's website, 65,160 currently active sites have registered from 215 countries. This data implies that the pedagogical integration of the software has made significant acceptance among the registered users. Of the 65,160 sites, 297 are coming from the Philippines, and the Silliman Online University Learning, known as SOUL, is one in the list. SOUL (<http://cms.su.edu.ph>) was customized using Moodle and launched in 2007 as utilized in the blended learning in Silliman University. On the other hand, the website of www.capterra.com listed 258 LMSs, and most of these are licensed and expensive, need a reliable internet connection, and require high maintenance cost. Some of these LMSs are the free and open source, but only a few offer offline/portable versions that can solve the disparity of Internet accessibility.

Schools need to have at least a reliable Internet connection to maximize the benefits of having a web-based learning

management system effectively. Because of this, the use of this kind of LMS created issues regarding its accessibility and utilization [4]. One of the major restrictions on most of the LMSs is the lack of offline capabilities and autonomy (www.NetDimensions.com). Incidentally, the Philippines is considered slowest in average broadband speed among the 10 ASEAN nations; yet, it is expensive (cited in [5]). Further, the study [6] revealed that 65.6 % of the respondents said that their school has installed within 1 to 5 Mbps of Internet bandwidth. Moreover, the study [7] shows that Internet cost and access and technical problems are substantial barriers to open online learning, while social interaction is somewhat of a barrier. Thus, maximizing the benefits of an Internet-based LMS has always been a challenge in the Philippines. It is in this context that a learning management system on sticks was proposed and developed.

This article describes the development of a handy learning management system that can run on USB flash drives, called PLMS. Specifically, it discusses the analysis, design, and implementation of PLMS. "Systems analysis is a process that provides system requirements description; defines the system functional requirements, and requirement specification of the proposed system" (Bruegge, and Dutoit, 2000, cited in [8]). The result of the survey on the level of priority of features is presented in this article. On the other hand, systems design provides the data structures, architecture, interfaces, and procedural detail of the software component of the research/project (Pressman, 2001, cited in [8]). In this paper, the database schemas, integration of portable application, customization process, and screenshots are presented. Further, system implementation is the process that includes planning the integration, installation of the database, testing the system, training end-users, and other production activities (Bruegge, and Dutoit, 2000, Whitten, Bentley, and Dittman, 2004, cited in [8]). In this article, a summary of test case results, training evaluation, and quality test are briefly explained.

II. RELATED APPLICATIONS

Among the most popular portable LMS is Poodle, which stands for portable online dynamic learning environment. It is free and is the mobile version of Moodle. Poodle has been used by the Mission Aviation Fellowship (MAF), whose vision is to provide pastors and church leaders in isolated areas with access to training for life and ministry using ICT; they found another solution to this situation of the digital divide. MAF promotes portable online education.

According to the website of MAF, Poodle runs Moodle courses without an Internet connection. It mounts Moodle on portable drives such as USB sticks, memory cards, HDDs, among others. It works with a collection of included and optional portable applications.

Another portable LMS is mEKP, which stands for Mobile Enterprise Knowledge Platform, developed by NetDimensions (www.NetDimensions.com). It is an end-user version of NetDimensions' powerful and robust LMS - running independently from a USB flash drive or from any user device that runs Java. mEKP is platform-neutral, and no installation or Internet connection is required. It is a portable Personal Learning Environment that enables users to access course catalogs, run courses, download reference materials, and take exams, as well as track and review training history information from any device with a USB port at their convenience anywhere, anytime. It supports both single- and multi-user environments. It also provides features like synchronization and content pre-loading. However, it is not free and offers a certain subscription fee.

Another licensed portable LMS is Atum-KFP developed by stratbeans.com. Atum-KFP uses proprietary Advanced Work-flow Configuration Framework that facilitates high-speed configuration of LMS as per client organization structure. It runs on mobile devices such as tablet PCs and can be used in locations where there is little or no internet connectivity.

Other LMS providers offer an application program interface (API) that is designed to accommodate offline learning on a subscription basis like Tin Can API (www.tincanapi.com). An API specifies how some software components should interact with each other. Moreover, there are software providers like upsidelearning.com that creates custom learning portals or tools on CD or USB-based mini-LMS, but again, it entails subscription cost as well as maintenance cost.

III. SOFTWARE DEVELOPMENT METHODS

A. Development Framework

The study utilized a customized software development model where it combines traditional and non-traditional development model. It consists of five major phases; these are user requirements analysis, customization and coding, system testing, training and quality testing, and delivery and dissemination (Fig. 1).

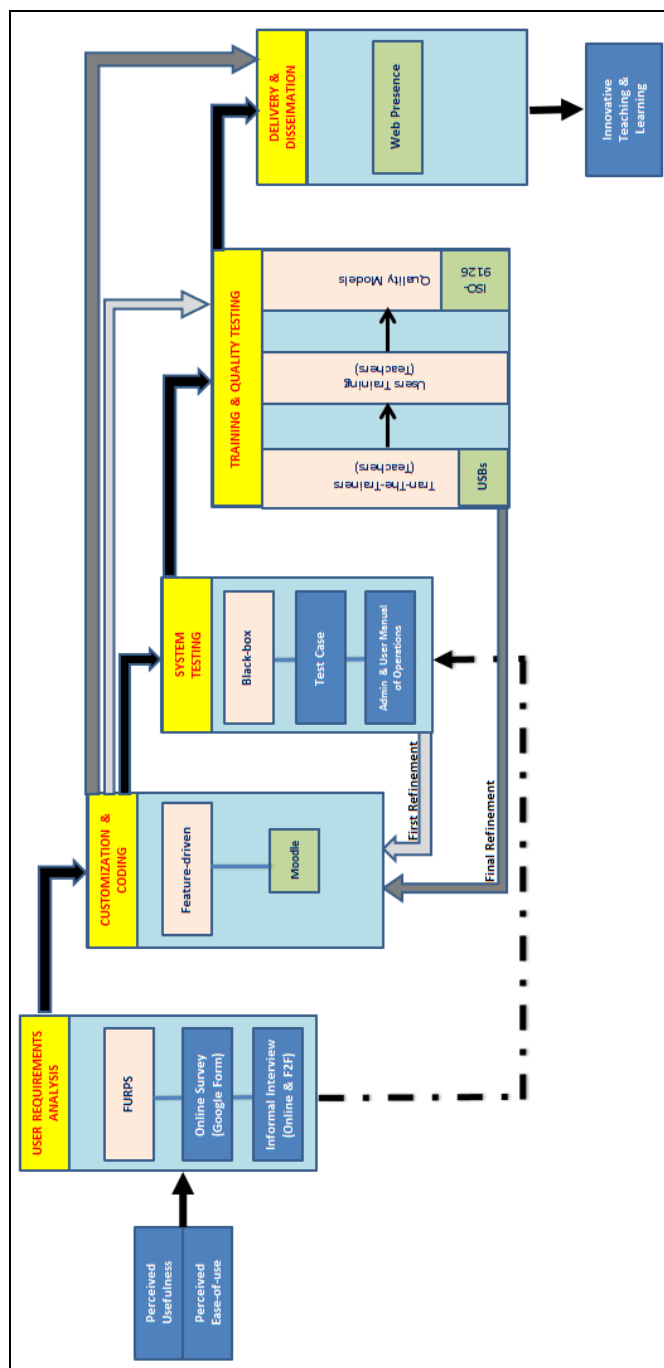


Fig. 1. Development framework of the project.

User Requirements Analysis

The quantitative data on the perceived usefulness and ease-of-use of PLMS [9] serve as input to the user needs analysis. Notably, PLMS is seen to be quite useful and perceived slightly to be easy to use. During the elicitation of functional and non-functional requirements, the FURPS (functionality, usability, reliability, performance, supportability) model by Robert Grady at Hewlett-Packard, cited in [10], was thoroughly followed. An informal interview, both online and in person, was conducted. The actual elicitation of user requirements was carried out using a survey questionnaire. Specifically, the respondents were asked about the priority

level of the features of a learning management system. There are 122 respondents who are teacher educators from four provinces in the Central Visayas region of the Philippines. The respondents were asked to rate the level of priority of the proposed features of PLMS.

Customization and Coding

This study utilized the feature-driven approach during the customization and coding phases. Feature-Driven Development (FDD) “is a client-centric, architecture-centric, and pragmatic software process” [11]. FDD “is an agile, highly adaptive software development process that is highly and short iterative” [12]. There are five main activities in FDD, adopted from [12], that were performed iteratively. These are to (a) develop an overall model, (b) build a features list, (c) plan by feature, (d) design by feature, and (e) build by feature (see Fig. 2). Several meetings were also conducted in every activity.

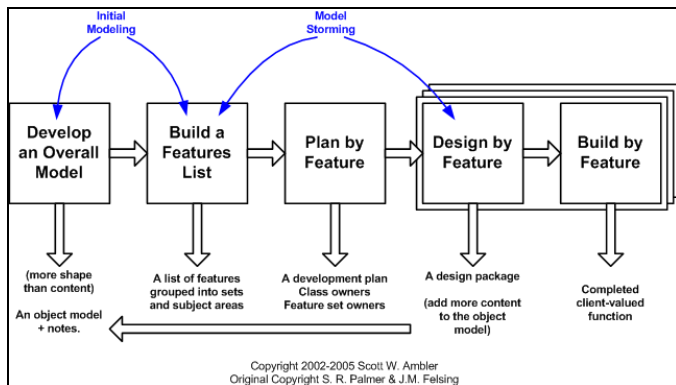


Fig. 2. Feature-driven development framework.

System Testing

A black box testing, sometimes called behavioural testing, was the technique used. Techopedia.com defines black box testing as “a software testing technique that focuses on the analysis of software functionality, versus internal system mechanisms.” A team of testers was created to test PLMS. The testers used 76 test cases as a guide using the format proposed in the study [13]. See a sample test case in Fig. 3. Findings from the testing were given to the programmer for action and revisions of the specific features. Then, a user’s manual was developed.

Training and Quality Testing

Training is one of the critical activities during the implementation phase of software development cycle [14]. It is a factor for the successful implementation of any software [15]. There were two phases of training conducted: train-the-trainers training and end-user training. The train-the-trainers training was a 3-day activity with the aim to identify a pool of trainers. Seventeen teachers were selected as trainers-ambassadors of PLMS. On the one hand, the end-user training is aimed to demonstrate, practice, and do hands-on activities with the use and integration of the newly developed digital teaching tools into the classroom. The participants of the said training must be willing to use and integrated the tools into

their teaching instruction. A total of 74 end-users were trained. At the end of the end-user training, a survey on the usability evaluation was conducted. The survey questionnaire was based on ISO 9126.

TEST CASE FORMAT					
Study Name		DEVELOPMENT OF A PORTABLE LEARNING MANAGEMENT SYSTEM			
Test Case ID:		Test Design By:			
Title:	Managing blocks.	Test Executed By:			
Test Priority:		Test Execution D:			
Pre-Conditions:	User must have successfully login as teacher. User must have selected a course to edit. Editing mode must be turned on.				
St	Test Steps	Test Data	Expected Result	Actual	Notes/Comm
1	Click the Assign roles button (face icon) to assign a role		Users will be directed to assign role page.		
2	Click the Hide button(eye icon) to ide the block		the block where the hide button is clicked will be hidden		
3	Click the configuration button (writing icon) to configure				
4	Click the delete button(x icon) to delete the block				
5	Click the movebutton to move the block.				
Post-Condition s:					
Signature Over Printed Name of the Tester					

Fig. 3. Use case sample.

Delivery and Dissemination

The PLMS package is disseminated through a link at www.davemarcial.net. It is also stored in Dropbox as well as on Google Drive. Instructions for downloading the software were made clear and straightforward considering that not all teachers are technologically inclined.

B. Development Tools

The tools used during the development of PLMS are Poodle, Sublime Text, Fiddler, and Windows. Poodle is an open-source offline learning management system tool that contains within it a portable web server, database, and PHP library. This tool is used as the primary tool to hold the PLMS. Version 2.1 was used, and it was downloaded from https://svns.maflt.org/rc/public/lt/tools/Poodle. Sublime Text is a versatile text editor that is used to update the pages in Moodle as well as the XML configuration files inside Moodle. Fiddler is a web performance and validation tool used to verify the posted messages between Moodle. Windows is the operating system used to host and test the portable application. In other words, PLMS can only run on computers with a Windows operating system with at least Windows XP SP3.

IV. ANALYSIS, DESIGN, AND IMPLEMENTATION

A. Priority of PLMS Features

Based on the survey, the highest mean of priority as perceived by the respondents is in activities where teachers

can involve students' activity in their learning. These activities pertain to assignments, forum, quizzes, and others. On the one hand, the lowest mean of priority is Editing Text where teachers can use the text editor in their writing and editing activity (see table I).

TABLE I. Priority of features.

Features	Functions and Processes	Mean	Description
ACTIVITIES	I can involve students actively in their learning (assignments, forum, quiz, etc...)	8.72	High
QUESTIONS	I can create questions for use in quizzes and other tests.	8.65	High
GRADES	I can use the gradebook, scales, and advanced grading methods.	8.62	High
RESOURCES	I can add static materials to my course (pdf, documents, video, links, etc...)	8.58	High
TRACKING PROGRESS	I can control and display progress through a course.	8.36	High
REUSING ACTIVITIES	I can copy or recycle elements of my course.	8.29	High
BLOCKS	I can add extra items and information to the sides of my course page (calendar, menu, feedback, etc.)	7.88	Medium
CLASS ENROLLMENT	I can give students access to my class.	7.81	Medium
GROUPING USERS	I can put students into groups and why this is useful.	7.80	Medium
CLASSES	I can set up classes.	7.75	Medium
EDITING TEXT	I can use the text editor and what the icons mean.	7.59	Medium

B. Database Schemas

Fig. 4 is Moodle's role database schema. Moodle core entities have its respective tables in the form of the "user," "course," "course_categories" and "course_modules" tables. In managing these entities, Moodle uses roles to control the features and functionalities available to the users. The roles are given rights and capabilities to manage other entities, including user, course, and modules. Moodle uses enrollments to associate users with existing courses. The "enroll" and "user enrollments" tables hold details of this association (see Fig. 5).

C. Portable Application Integration

A portable app is a computer program that a user can save and run on a portable device and use on any Windows computer without installing it (PortableApps.com). According to the website "when your USB flash drive, portable hard drive, iPod or other portable device is plugged in, or your cloud drive is synced, you have access to your software and personal data just as you would on your PC." Moreover, it was proven in this study. Moreover, this study also demonstrates that when the device is unplugged, none of the personal data is left behind. By packaging portable versions of the Apache Web Server, MySQL database, and Firefox browser, one can bring an offline learning management system.

Normally, applications like MySQL database require being installed. However, using a portable version, all files are already packaged inside the drive. "Portable Apps usually use advanced compression techniques resulting in a 20-30%

savings in packaged files" (PortableApps.com). All data, including registry and configuration or settings required by the applications, are stored on the drive. It is self-contained which means it does not need any remote files or utilities for it to run (PortableApps.com).

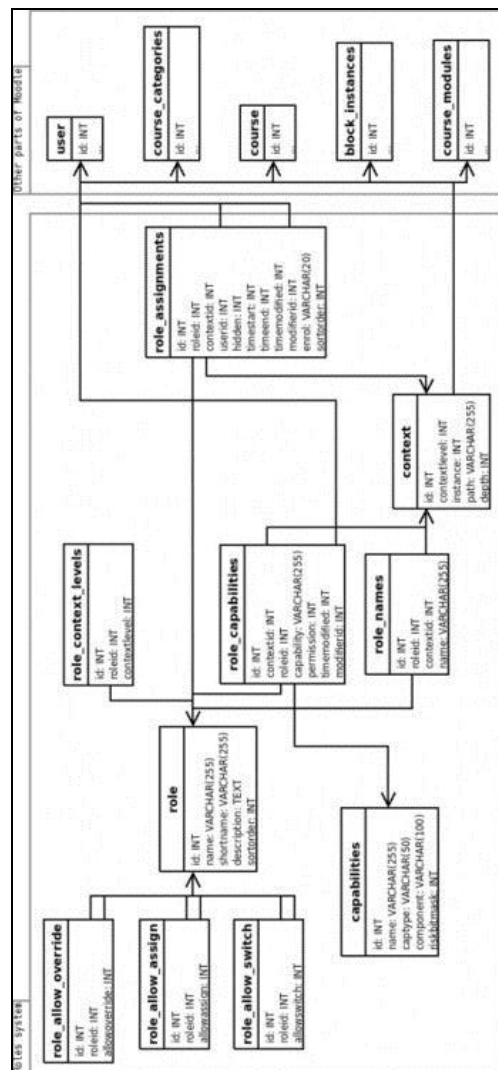


Fig. 4. Moodle's role database schema.

One can create portable apps using special applications called virtualization or portable application creators. VMware ThinApp is one such application. These applications work by identifying the files required for an application and stores them. The application identifies files by taking a snapshot before and after installing a candidate application for portability. Differences between the snapshots are then identified as required and packaged along with the application.

D. Customization

In setting up the program, the built-in features were initially used to configure and update the PLMS to what is required. These features come in the form of "Administration" modules, which range from administering the entire site to specific entities such as "users" and "courses." In controlling

the layout and display of the page, themes were applied to the “Appearance” module.

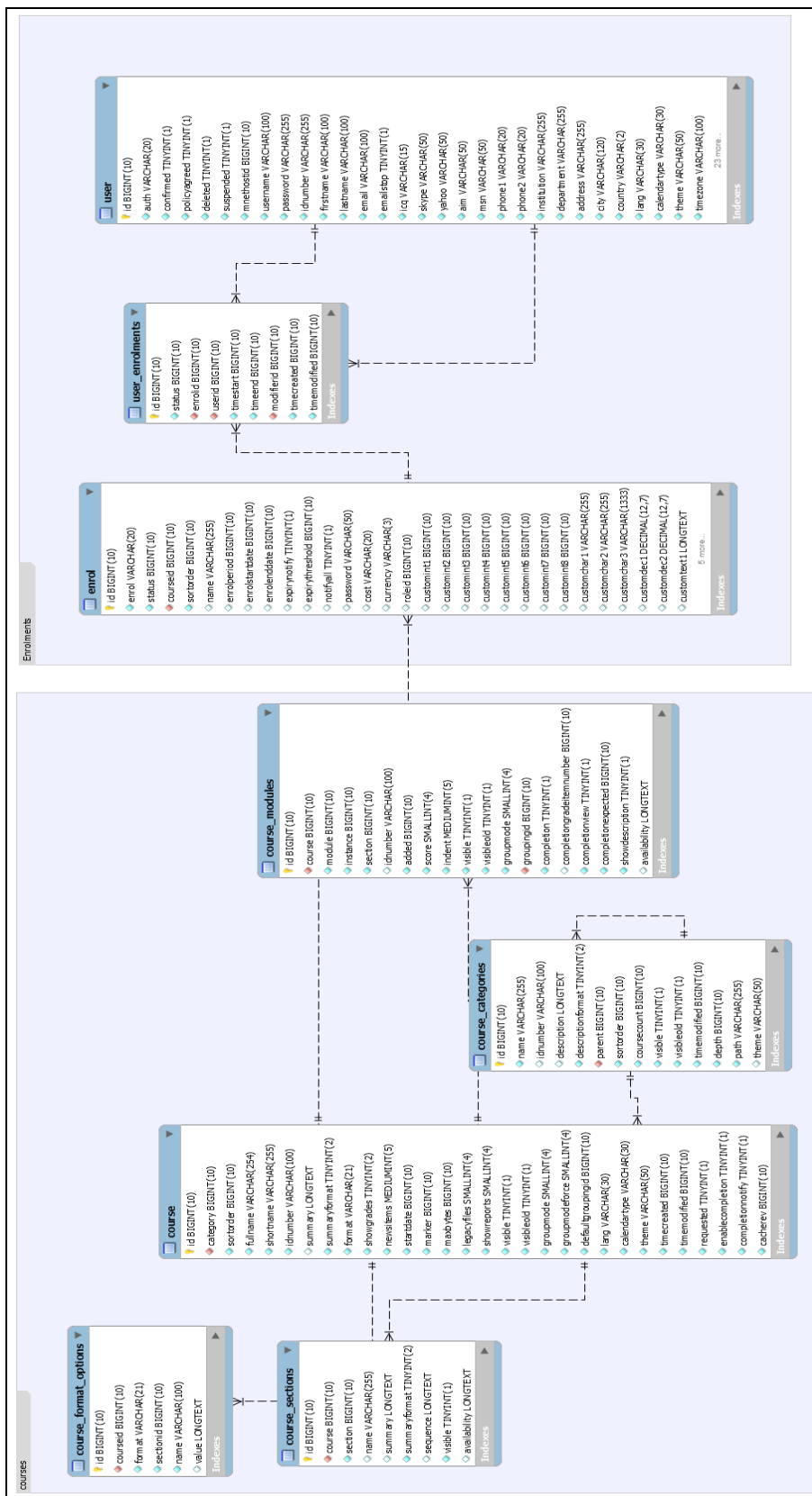


Fig. 5. Moodle's enrollment database schema.

In setting up the program, the built-in features were initially used to configure and update the PLMS to what is required. These features come in the form of “Administration” modules, which range from administering the entire site to specific entities such as “users” and “courses.” In controlling the layout and display of the page, themes were applied to the “Appearance” module.

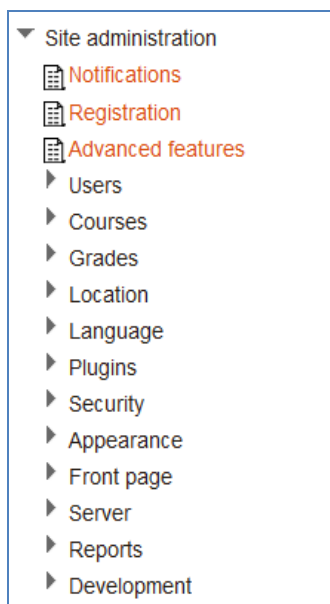


Fig. 6. Edit tool box.

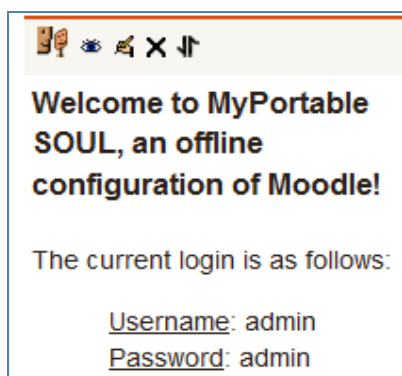


Fig. 7. Permission Module

Another built-in feature used is the quick edit toolbox that appears in most elements in the app. This quick edit box can be used to hide/show and edit the content inside it (Fig. 6). An important aspect of configuring involved the definition of roles for the project. This configuration was done by using the “Permissions” module to define roles (Fig. 7). Roles are then assigned capabilities which allow or deny them a feature in the program (Fig. 8). Proper setup of these capabilities is essential in allowing the entire PLMS process to work (Fig. 9). In the case of PLMS, it was customized in such a way that teachers are allowed to create courses and enrol users.

For features, like Hiding Course Creator, which cannot be toggled using built-in capabilities, corresponding files must be edited using a text editor. This customization also applies to page elements, such as links and menus, which cannot be

removed or added to be changed by updating the source code itself. Some are there by default that needs the codes to be edited. See figure 10 for the sample editing activity.

Role	Description
Manager	Managers can access course and modify them, they usually do not participate in courses.
Course creator	Course creators can create new courses and teach in them.
Teacher	Teachers can do anything within a course, including changing the activities and grading students.

Fig. 8. Role and its description.

Course overview	
View course overview report report/courseoverview:view	Allow
Course categories	
Create courses moodle/course:create	Not set
Course completion	
View course completion report coursereport/completion:view	Allow
Logs	
View course logs coursereport/log:view	Allow

Fig. 9. Allow features.

```
// Get the directory to this file
$moodleDir = realpath(dirname($_SERVER['config.php']));
// Parse the path and get the elements
$dirArray = explode("\\", $moodleDir);

// Build the dirroot and dataroot path
// dirroot: <drive>:\www\moodle
// dataroot: <drive>:\moodledata
$dirRoot = "$dirArray[0]\\$dirArray[1]\\$dirArray[2]";
$dataRoot = "$dirArray[0]\\moodledata";
// End build dynamic paths

// Get dynamic port number and host information
$httpHost = $_SERVER["HTTP_HOST"];
$wwwRoot = "http://$httpHost/moodle";
// End get dynamic port

unset($CFG); // Ignore this line
$CFG = new stdClass();
```

Fig. 10. Editing moodle config (Default directories).

E. Screen Shot of PLMS

The succeeding figures are screenshots of the major features of PLMS. It is presented according to priority results of the survey. Fig. 11 is the screenshot of Activities module. As a teacher, the students may be required notes, texts, files, and offline activities from which they can submit on the date and time specified. Quizzes for students may also be formulated. Based on the pace of the lessons, questions may be created for use in quizzes and other tests (Fig. 12).

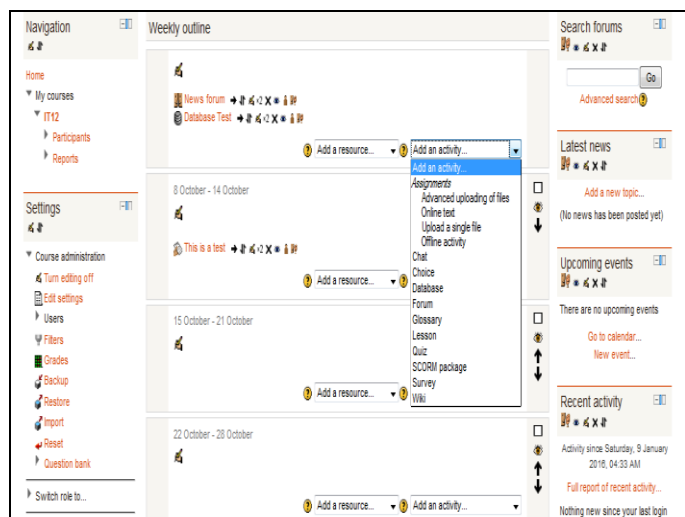


Fig. 11. Activities.

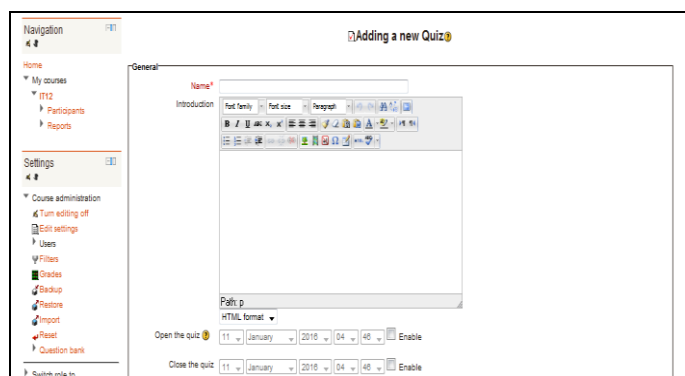


Fig. 12. Questions.

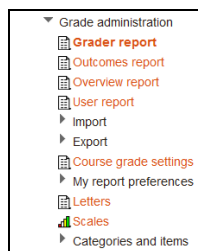


Fig. 13. Grades.

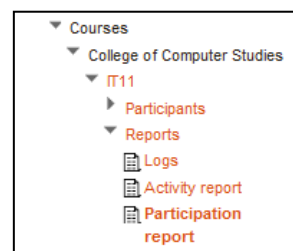


Fig. 14. Tracking progress.

Submission by students may be graded individually, and teachers can control and track the progress of the course through the application (Fig. 13). Each post is also dated so teachers can see what the class was doing on particular dates (Fig. 14). Files that can be uploaded and shared on the application may be in different formats, such as .pdf files, .doc/.docx files, videos, website links and the like. Teachers may customize their learning environment to include a description or purpose of the course, announcements, calendars, feedback, and others.

PLMS can handle class enrollment as well as grouping of users. Getting more students to be part of the learning system is very easy and quick. It may be with or without a password, and teachers can create groups. Groups are useful if teachers have two or more sections with the same group of activities. This way, teachers do not have to post notes, exams, announcements, and others, to every section, but make all these available to all when teachers want to. At the same time, posting of exams and submissions of assignments may be customized so that date and time may be different for a group of students or sections. Teachers also have full control of creating more classes in the system. To make it easier to understand for the students, the text editor is available within the application itself. Moreover, this allows the teacher to specify the font, size, appearance, and colors for the text much like working with other more common applications like MS Word. Here, teachers may change their texts to bold, italics, underline, subscript, and superscript. Teachers may also add numbered/bullet lists or change justification settings.

F. System Test Results

The testing was done on different desktop computers. USB flash drives with 4GB and 8GB capacities were used. The Moodle folder containing the portable application was saved in both flash drives. The 8GB flash drive was tested on a laptop computer running Windows 7 Starter 32-bit with Intel Atom N570 processor and 2GB memory, while the 4GB flash drive was tested on a desktop computer running Windows 7 Starter 32-bit Intel Core2 and 2GB memory.

The result suggests that PLMS passed the test, and it is ready to use. There were no deviations encountered during testing, and most of the expected results were achieved. Errors encountered were deemed very minimal, and it was determined that these would not affect the performance of the application. The errors were more on buttons and links not found on pages and unclear file types that were required. The application was observed to be running at a moderate pace and noticeably slowed down during the loading of certain pages

and saving of specific resources or activities. The browser usually loads a page for an average of two to five minutes. There were some instances when the application prompts an error during the first time that it is loaded from the USB flash drive. The USB flash drive would have to be removed correctly, and the application would be loaded again to make the application run smoothly. The loading speed was further tested by comparing Internet-enabled computers with computers without an Internet connection. The application loads faster on computers without an Internet connection. After the testing, the team met with the programmer to summarize and discuss the minor glitches of PLMS. Each bug or error was solved before the train-the-trainers training. The train-the-trainers' participants also gave their comments and suggestions for the improvement of PLMS.

G. Train-The-Trainers Training Outputs

There are three major deliverables during the train-the-trainers training. These are the action plan, terms of engagement, and a list of comments and suggestions. The action plan consists of four interrelated objectives. The first goal is to assist the launching activity. Second, to integrate PLMS in the classroom, and the third is to cooperate with the effectiveness evaluation of the two digital applications. To promote PLMS is the final objective of the action plan. Similarly, Terms of Engagement were signed. This contract was made between the project leader and the identified trainers, testifying their willingness to be part of the experiment. Most importantly, throughout the lecture and demonstration, the trainees contributed their inputs and suggestions for the improvement of PLMS.

H. Launching and End-user Training Outcomes

The end-user training was evaluated excellent regarding the participants' levels of reaction, learning, behavior, results, and management. The quantitative result implies that the trainees are extremely satisfied and engaged during the training, and they see the relevance of the training to their teaching profession. The result of the level of learning connotes that trainees highly perceive that they extremely acquire the intended knowledge, skills, attitudes, confidence, and commitment based on their participation in the end-user training. The result of the level of behavior signifies that the trainees highly believe that they can apply what they learned from the training when they are back on their teaching job. The level of results indicates that the trainees highly believe that targeted outcomes occur as results of the training event and subsequent reinforcement. Moreover, the degree of training management indicates that the trainees highly believe that the training was very organized and managed well.

I. Quality Test Results

A total of 116 respondents tested the quality of PLMS. Using a survey questionnaire which was adapted from ISO 9126; the respondents strongly agree that PLMS is of acceptable quality. Specifically, the results show an overall mean of 3.55 of the quality level of PLMS (Table II). The result implies that PLMS is highly functional, reliable, usable,

efficient, portable, and embedded with the pedagogical characteristic.

TABLE II. Quality level.

Quality Components	Total	
	Mean	Description
Functionality	3.62	Strongly Agree
Reliability	3.46	Strongly Agree
Usability	3.39	Strongly Agree
Efficiency	3.49	Strongly Agree
Portability	3.71	Strongly Agree
Pedagogical Characteristics	3.61	Strongly Agree
Overall mean	3.55	Strongly Agree

V. CONCLUSION

The development of the learning management system on a stick is successful. PLMS organizes classroom information and learning activities, even without Internet connectivity. It is a kind of information system that is transferable to multiple environments and manageable to a variety of infrastructure specifications using a USB flash drive. It has features like those mentioned in the study [3] that can centralize and automate administration, use self-service and self-guided services, assemble and deliver learning content offline, support portability and standards, and personalize content and enable knowledge reuse for teachers. PLMS was designed using pedagogical principles embedded in Moodle, to help teachers develop innovative teaching and learning practices. Because it is portable, PLMS is susceptible to virus attacks that need proper handling among the users.

VI. RECOMMENDATIONS

Based on the testing that has been done to check the speed of the application, a PC with no internet connection makes the application load faster. Using a faster USB flash drive like the 3.0 USB is recommended. Users must take full responsibility in ensuring the security of the files most especially to virus attacks. Users should install portable antivirus app. Teachers and students must regularly create a backup file of their PLMS resources. For future development, a MAC version of PLMS is recommended.

There must be regular user training activities that will emphasize on the pedagogical use of PLMS. Specifically, teachers must be trained in managing the PLMS administrative account. Notably, the training conducted was purely on the teacher's account to avoid complexity in operation considering that the expected users are not digitally skilled. Moreover, teachers should choose any of the proposed implementation schemes in terms of the level of integration, mode of integration, and a terminal of integration. See table III for the full description of all the schemes.

TABLE III. PLMS implementation scheme.

Levels of Integration	
Resource Management	As a resource repository of files, such as file pdf, doc, xls, and others
Activity Management	As a resource repository and as a portal for classroom activities such as assignments and quizzes
Modes of Integration	
Specific	The teacher prepares and distributes PLMS in a particular purpose. For example, conducting exams, practice testing for board review, reading purposes, or any other precise purpose.

	The teacher collects PLMS at a specified date and review student's PLMS for grading or monitoring purposes.
Periodic	The teacher prepares and distributes PLMS in a periodic mode of integration. For example, during Prelim, Midterm, Pre-final, and Final period. The teacher collects PLMS at the end of each period and review student's PLMS for grading purposes.
Full	The teacher prepares and distributes PLMS in a full mode of integration. For example, prepares PLMS from day 1 to day X of his/her classes. The teacher collects PLMS at the end of each period (prelim, midterm, pre-final, final) or the end of the final period and review student's PLMS for grading purposes.
Terminal of Integration	
Freestyle	Students can work and manage his/her PLMS to any computer like on campus, at home, or in computer shops. In this integration, a USB flash drive or another portable drive is necessary for all students.
Laboratory-based	PLMS is installed in the un-networked laboratory. In this case, students can only manage his/her PLMS inside the laboratory. In this integration, a USB flash drive is not necessary.

ACKNOWLEDGMENT

The funding support provided by the Commission on Higher Education through the Philippine Higher Education Research Network, facilitated by Silliman University through Research and Development Center is gratefully acknowledged. The services of Joross Barredo in formulating the test cases and the assistance of Alfie Arcelo are also gratefully acknowledged.

We would like to thank Department of Science and Technology – Philippine Council for Industry, Energy and Emerging Technology Research and Development (DOST-PCIEERD) for the funding support in presenting this article at the First International Conference on Interdisciplinary Research Methodology and Innovation held at The Federal Kuala Lumpur, Kuala Lumpur, Malaysia on July 12-13, 2017. Our appreciation also goes to the participants of this conference for their comments and suggestions on the second version of this paper. Our appreciation also goes to the reviewers of the 7th International Conference on Cloud Computing, Data Science and Engineering held at Amity

University, Uttar Pradesh, India for their comments and suggestions on the original version of this paper.

REFERENCES

- [1] M. Cuéllar, M. Delgado, M. Pegalajar, "A common framework for information sharing in e-learning management systems" *Expert Systems with Applications*, vol. 38, issue 3, pp. 2260-2270, 2011.
- [2] UNESCO Bangkok, "What is a learning management system?", Retrieved from UNESCO: <http://www.unescobkk.org/education/ict/online-resources/features/lms/>, 2006.
- [3] R. K. Ellis, "Field guide to learning management systems," Learning Circuits, American Society for Training & Development (ASTD), 2009.
- [4] F. Paragina, S. Paragina, A. Jipa, T. Savu, and A. Dumitrescu, "The benefits of using MOODLE in teacher training in Romania," *Procedia - Social and Behavioral Sciences*, vol. 15, pp. 1135-1139, 2011.
- [5] L. S. Marasigan, "Phl's slow but expensive internet service," *Business Mirror*, <http://www.businessmirror.com.ph/phls-slow-but-expensive-internet-service/>, August 23, 2015
- [6] D. E. Marcial, "Information technology resources in higher education institutions in the Philippines," *Philippine Information Technology Journal*, vol. 5, issue 1, 2012.
- [7] D. E. Marcial, R. Caballero, J. Rendal, and G. Patrimonio, "I am Offline": Measuring Barriers to Open Online Learning in the Philippines", *Information Technologies and Learning Tools Journal*, vol. 45, issue 1, pp. 28-41, 2015
- [8] C. Sta. Romana, R. Gamboa, D. E. Marcial, G. Gabison, and A. Sioson, "Undergraduate research and capstone project manual," Philippine Society for Information Technology Educators, 2012.
- [9] D. E. Marcial, "Teacher education perceptions of a proposed mobile classroom manager," *The IAFOR Journal of Education*, vol. 3, pp. 13-29, 2015.
- [10] P. Eeles, "Capturing architectural requirements," IBM Developers Work, 2015. <http://www.ibm.com/developerworks/rational/library/4706.html#footnotes>
- [11] S. W. Ambler, "Feature driven development (FDD) and agile modeling," *Agile Modelling*, 2005. <http://www.agilemodeling.com/essays/fdd.htm>
- [12] S. Goyal and J. Schiller, "Feature driven development", Technical University Munich, WS 2007/08.
- [13] U. Eriksson, "How to develop a template for test cases," ReQtest, <http://reqtest.com/tutorials/how-to-develop-a-template-for-test-cases/>, September 28, 2012.
- [14] J. A. Hoffer, J. F. George, and J. S. Valacich, *Modern Systems Analysis and Design*, Upper Saddle River, New Jersey: Pearson Education, Inc., 2011.
- [15] G. Wickes, "5 Keys to End-User Training". Retrieved from <http://www.sunviewsoftware.com/learn/blog/5-keys-to-end-user-training>