Development Of An Online Printing Press Production And Sales Management System Using Software Engineering Workbenches

Elbert M. Galas

Abstract: This project primarily aimed to develop an Online Production and Sales Management System for a printing press company. It is an online-based application designed for the management of its business production and sales that is more efficient and synchronized than the current manual system. Specifically, it focused on the development of the following modules: (1) File Maintenance Module; (2) Material Inventory Management Module; (3) Job Order Management Module; (4) Billing Module; (5) Payment and Accounting Module; (6) Report Generation Module; (7) Security Module; and (8) User Management Module. The system minimized the company's production and sales endeavor for every transaction and processes they go through, reduced impediments, and upgraded the system into a new and improved one. It enabled handling a large amount of data and made the transaction processing faster and easier. The development of the system was guided by the two main methods in developing systems, namely: Rapid Application Development (RAD) and Model-Driven Development (MDD) methods. The tools used in the development are Microsoft Visual Studio 2010 Ultimate Edition as an integrated development environment (IDE), MS SQL Server 2008 as the database management system, C# 2010 as the programming language combine with AJAX, CSS, jQuery, HTML 5, and ASP.Net 4.0 as framework or platform. The implementation of these methodologies was aided by utilizing Software Engineering Workbenches (SEWs). These workbenches are integrated into Microsoft Visual Studio 2010 Ultimate Edition. Based on the evaluation of the system, the proposed system is highly acceptable by a respondent printing press management and employees, customers, and IT experts. In terms of usability, the proposed system is rated with an average mean of 6.4 described as "Strongly Agree." Thus, it was concluded that the developed system offers functions that can help improve the performance of a printing press company services and is highly recommended for its immediate utilization.

Index Terms: Model-Driven Development, Production and Sales Management System, Rapid Application Development, Software Engineering Workbench

1. INTRODUCTION

Production means an application of processes and methods using technology to transform tangible inputs such as raw materials, semi-finished goods, subassemblies, and intangible inputs such as ideas and knowledge to arrive at the desired product such as goods or services without sacrificing the desired quality. Resources are used in this process to create an output that is suitable for use or has exchange value. [1]. A company has lots of output that ought to be made for them to reach their customer's needs. Manual monitoring of production is somewhat tedious and requires ample time, but with the advent of production management or monitoring system, this problem can be alleviated.

- *Elbert M. Galas is a graduate of Bachelor of Science in Information Technology at the Pangasinan State University – Urdaneta City Campus, Master of Information Systems at the University of the Philippines – OU Los Banos, Laguna and currently pursuing Doctor in Information Technology at the Technological Institute of the Philippines-Quezon City, elbertgalas@gmail.com. His research interest lies in the areas of Information Systems, IT Security and Protection, and Data Mining.*

Production Monitoring System or PMS [2] is an effective and efficient management tool and used widely in production lines, warehouses, and offices. It has been proven to increase productivity, save resources, boost workers' morale, and ensure production target is met on time. While Sales [1], on the other hand, is the activity or business of selling products or services. Many people are obtaining their income through selling products, and almost all of us are surviving because we buy goods and services from those sellers for our commodity needs. Since countless people are selling and purchasing different products and services for their daily lives, there are times when sellers cannot accommodate their customers immediately, especially those who have a large number of orders. Moreover, they cannot record their transactions right away, which results in their profit loss and some more plights. A business must have a sales management system for the company's attainment of sales goals in an effective and efficient manner. It can be further achieved with the utilization of appropriate planning, employment, and controlling executive resources. A Sales Management System (SMS) [3] can be thought of as the mechanism used by sales managers to make sales management more comfortable and faster. It has been defined as an Information System (IS) used by sales professionals or business entities for sales tracking, which facilitates the sales management process. The respondent printing press company traditionally touts their services in a house to house manner. They go to their apparent customers to endorse their services. This process requires more time and effort to conduct. The company does not have an inventory of materials that bring them into intricacies in scrutinizing their supplies used for production and checking their stock availability. They come across difficulties in calculating their accounts payables and receivables as they use a manual process of computing, which may lead to a lot of chaos and errors. Also, their existing system requires a lot of paper works

wherein even a small transaction requires many paper fill and loss of even a single document may lead to a difficult situation because all the documents are usually interrelated. Thus, the proponent decided to develop an Online Printing Press Production and Sales Management, which minimized the company's production and sales endeavor for every transaction and processes they go through, reduced impediment, and upgraded the system into a new and improved one. It enabled handling a large amount of data and made the transaction processing faster and easier. The development of the system was guided by the two main methods in developing systems, namely: Rapid Application Development (RAD) and Model-Driven Development (MDD) methods. The implementation of these methodologies was aided by utilizing different Software Engineering Workbenches (SEWs). SEW is an integrated set of CASE tools that work together to support a significant process such as software design or configuration management. Workbench is composed of two or more CASE tools and support specific software-process activities. Hence they achieve: (1) a homogeneous and consistent interface (presentation integration); and (2) seamless integration of tools and toolchains (control and data integration). [4] An example of a workbench is Microsoft's Visual C# programming environment. It incorporates several development tools such as GUI builder, a code editor, debugger, etc. Workbenches are also categorized in the same manner as tools focusing on analysis, development, verification, as well as being focused on the upper case, lower case, or processes such as configuration management that span the complete life-cycle. [4] According to Fuggetta [5], another classification of SEW is based on framework workbenches that support one or more activities, tools that support only specific tasks in systems development, and environments that support a large part of the software process. Based on these classifications, tools can be further categorized into editing, programming, verification and validation, configuration management, metrics and measurement, and project management tools. Workbenches are classified depending on the activities that they support as: business planning and modeling, user interface development, programming, verification and validation, maintenance and reverse engineering, configuration management, and project management workbenches. Finally, environments are classified as toolkits, which are integrated collections of products, language-centered, integrated, fourth-generation environments, or process-centered environments. Computer-aided Software Engineering (CASE) is a technology that uses a computer-assisted method to manage software development, especially on large, complex projects involving complicated software components and people. Using CASE allows designers, code writers, testers, planners, and managers to share a standard view of each stage of development. [6]

2 SYSTEM SCOPE

This project is an online application for managing production and sales for a printing press company. Specifically, it focused on the development of the following modules: File Maintenance Module. The system provides a tool to manage customers, employees, suppliers, and materials records. Material Inventory Management Module. The system includes controlling and monitoring of availability of raw materials stocks, needed for the production, from ordering from a supplier into its delivery. The inventory clerk will handle this module. Job Order Management Module. This module provides a mechanism for the Operation Department to manage job orders, which include creation, approval, rejection, and reactivating job orders. It also provides an interface for customers to monitor their job order's production status through their online account. Billing Module. This module will allow the billing clerk to generate a billing statement of customers. Payment and Accounting Module. This module accepts a customer's payment based on the generated bill. It also includes the management of accounts receivable and accounts payable of the company. It can automatically compute the disbursement of each transaction and can immediately produce a delivery receipt for every finished service. It also sets up a person who authorizes the receipt before releasing of finished job orders. The cashiering and accounting department will handle this module. Report Generation Module. This module will allow the cashiering and accounting department to generate reports such as a list of jobs, account payables, account receivables, daily, weekly, monthly, and annual sales reports. Security Module. This module provides administrative tools for system maintenance, facilitated management of look-up tables, system configuration, scheduled services, facilitated the management of users and groups, privileges, and security. User Management Module. It includes management and maintenance of user accounts. The administration department will manage the File Maintenance, Security, and User Management modules.

3 RELATED WORKS

A study conducted by Laar, et al. [7] entitled Design and Development of a Sales Management System for SMEs in Northern Ghana presents a tailor-made computerized sales management system which is divided into three main components namely: (1) System Entry Component, (2) Administrative Component, and (3) Point of Sale (POS) Component. The system entry component is the first part of the system every user interacts with before undertaking or carrying out any other activities. It is composed of the main program background frame, the login frame, and the system main menu. The administrative component is also called settings in this project. It is the largest component within the system; it includes several functions for monitoring and controlling or defining which functions a user can call at any point in the system. The POS system is the point where most of the data handled or processed by the sales management system are generated. Therefore, it includes a number of data input fields and data storage procedures to ensure effective tracking. The POS also consists of some administrative features such as the activation or deactivation of charges and promotions, viewing, and printing reports. This system is designed and implemented using the object-oriented development approach. In line with object-oriented development, the Unified Modeling Language (UML) is used to design class models and use case diagrams for the main components to provide a blueprint for implementation. Achuth N and Shashi Kiran G. [8], in their study entitled Analysis and Design of Computerized Production Management System for Garment Manufacturing Company, presented a production management system. The system is incorporated with three (3) modules, which include Product Development, Production, and Quality Management. Data Flow Diagrams are used to

carry out the analysis of data flow, information flow, and processing activities of the three (3) modules. Product Development database stores and keeps track of the employees, customers, suppliers, and final products. Product development describes the parts of the database represented. According to the requirements, they identify six (6) entity types corresponding to each other. The Product is developed based on customer requirements. The customer has a unique name, a unique code, and details on them as the attributes. It keeps track of the details of the customer to know whether the customer had any previous transaction with the company. The manufacturer also has a unique name, a unique code, and details about the firm. Analysis of data flow, information flow, processing activities of the three modules is carried out using Data Flow Diagrams. The analysis identified seven (7) processes in Product development, 14 processes in Production, and four processes in Quality Management modules. The design of the system entities, their relationship, and attributes is carried out using the Entity-Relationship (ER) modeling technique. The structural design of the system is carried out using the Unified Modeling Language (UML) technique. From the above-mentioned developed systems, the proponent came up with the idea to combine some features of production management and sales management system. Some capabilities of the said systems were applied to the proposed system in terms of the methodology and technology used during its development and implementation. Panagiotou, D. and Mentzas, G [9] in their study entitled A Knowledge Workbench for Software Development presented KnowBench, a knowledge workbench environment which focuses on the software development domain and strives to address these problems. KnowBench aims at providing software developers such as tools to ease their daily work and facilitate the articulation and visualization of software artifacts, concept-based source code documentation, and related problem-solving. Building a knowledge base on software artifacts by using the KnowBench system can then be exploited by semantic search engines or P2P metadata infrastructures to foster the dissemination of software development knowledge and facilitate cooperation among software developers. [10] A study entitled "Language Designer's Workbench," presented the vision of a language designer's workbench, which reduces the effort of language design by integrating implementation and verification. Their proof of concept suggests that this vision is at least feasible for a class of simple languages, and provides an outlook on a future in which software language design can be a matter of exploring design alternatives rather than grinding away on tedious implementation details. Another study [11] presented a reengineering workbench and architecture that allows for legacy systems written in procedural languages to be migrated to new object-oriented platforms. This methodology provides for specific design and quality requirements of the target migrant system to be considered during the re-engineering process through an iterative and incremental process. The migration process recovers an object model from the procedural code and incrementally refines it by taking into consideration the design requirements for the migrant system. The abovementioned studies also gave an idea to the proponent to make use of an SWE as a reinforcing tool to expedite the design and development of the proposed system.

4 METHODOLOGY

4.1 Overview

The project was guided by the two main methods in developing systems, namely: Rapid Application Development (RAD) and Model-Driven Development (MDD) methods. To be able to comply with the rapid development of the system, a SEWs are utilized.

4.2 Software Development Methodology

Developing an Online Printing Press Production and Sales Management System utilizing SEWs is challenging to do because there will be a series of tests and revisions before it will become functional and successful. Therefore, there are some useful tools in building integrated system methods such as System Development Life Cycle models. One of the most commonly used approaches is the Rapid Application Development (RAD) model, often referred to as the Prototyping. A RAD model, specifically evolutionary prototyping, is the frame of reference in developing the system. The evolutionary prototyping in Figure 1 shows the iterative process used in the development of the system. The process begins with a requirements gathering activity followed by a quick prototype designing and building. After analyzing the prototype, further refinements to the requirements are generated and the process begins again.

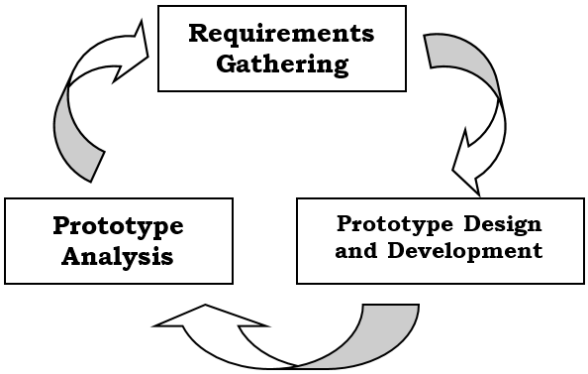

***Fig. 1**. The Evolutionary Prototype Model*

Requirements definition and analysis are the most crucial parts of any system development since the requirement is a description of what a system should need to do. Thus, to cope with the faster pace of the user's requirements, prototyping becomes a popular tool for accelerating systems development. The proponent made used of prototyping to alleviate vague requirements analysis due to rapidly changing or poorly understood requirements that arise during requirements gathering from areas of the existing system that are not well known to the proponent. In the system design stage, after all data has been gathered and analyzed, the design of the new system was done following the flow of the operations over the entire system which includes the interface and database design. Since the prototyping technique was used in the development of this system, a prototype was presented to users allowing them to give feedback. The improvement and presentation continued until such time that the users finally approved and accepted the system. In the development stage, the coding of the program by the proponent was described in

this stage. All programs needed to make the system operational were also identified and described. Testing was done after coding was accomplished. Testing the software was applied by the proponent to ensure that the defined input will produce the required results. Also, to determine whether the system can perform all of its functions in a realistic operating environment. Furthermore, feedbacks of the users were gathered and made necessary improvement out of it.

4.3 System Analysis and Design

4.3.1 System Model

Unified Modeling Language (UML) is composed of languages that primarily supports object-oriented modeling of software systems in terms of flows, objects, and messages [13]. UML is a tool for specifying software. Standardized diagram types help to describe and visually map a software system's design and structure. It can be possibly used to model any application, both specifically and independently of a target platform. Thus, the proponent analyzed and designed the proposed system through the following UML diagrams. Use Case Diagram that depicts the interactions between the system and external systems and users. In other words, it graphically describes who will use the system and in what ways the user expects to interact with the system. [12] The Class Diagram was used to describe the object and information structures used in the proposed system, both internally and in communication with its users. It describes the information without reference to any particular implementation. Its classes and relationships can be implemented in many ways, such as database tables, XML nodes, or compositions of software objects. [14] An Activity Diagram was used to document the flow of an algorithm or a business process and Component Diagrams to show the structure of a system and how it can be deployed to a logical environment. [15]

4.3.2 System Architecture

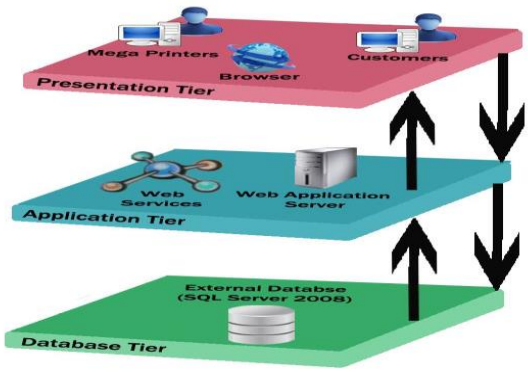

***Fig. 2**. System Architecture of the Proposed System*

Figure 2 depicts the proposed System Architecture. The presentation tier allows intended end-users to access the system through the use of a Web browser and its Graphical User Interface. This layer can access the application that is intended for the company and an application for customer use. The Application Tier is where server-side processing is being carried out. It connects the interface into the Database Tier for storing and retrieving data from the database. The Database Tier is where the database server for the system is being handled. In the proposed system design, the records will be collected centralized and consolidated from different end-users.

4.3.3 Development Tools

The following are the tools that were used in the development of the proposed system: Microsoft Visual Studio 2010 Ultimate Edition, an Integrated Development Environment (IDE) from Microsoft, was used to develop graphical user interfaces and console applications. MS SQL Server as the database management system that establishes a reliable data management product and distributes rich features, data protection, and performance for embedded applications. C# 2010 as the programming language in which the system was written. ASP.Net 4.0 as the framework or platform that was used to make the proposed system as a web applicationThe proponent also used the following software in the development of the system: AJAX, CSS, jQuery, HTML 5, recent versions of Mozilla Firefox or Google Chrome, and Windows 10 operating system.

4.3.4 Utilization of a Software Engineering Workbench

The proponent took advantage of the useful features of UML (Unified Modeling Language) support integrated into the Visual Studio 2010 Ultimate Edition IDE during the analysis, design, and development of the system. This technology served as the workbench to understand, design, and develop the system. It helped the proponent to visualize how the system will work, clarify users' needs, define the architecture of the system, analyze the code, and ensure that code meets the requirements. Figure 3 depicts the list of UML diagrams integrated in the Visual Studio 2010 that the proponent used in modeling the project.

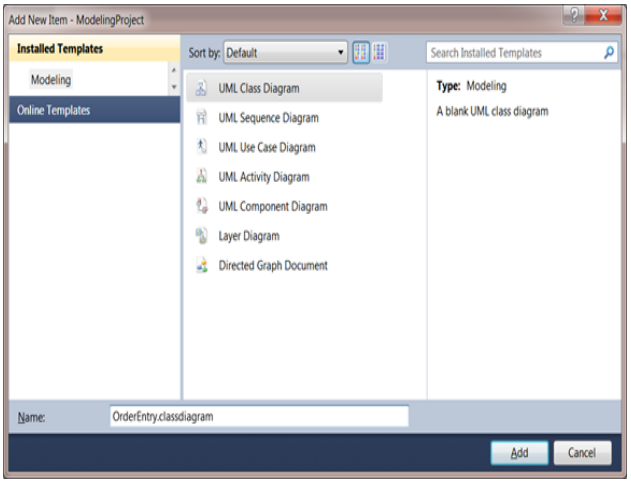

***Fig. 3**. UML Interface of Visual Studio 2010*

The proponent made use of this workbench in creating the Use Case, Class, and Activity Diagrams of the proposed system. In addition, a T4 text template was also used to generate code from the UML model. Text Template Transformation Toolkit is a template-based code generation engine. In this workbench, definitions of model elements are stored in a centralized repository as part of the Modeling Project; they can be displayed on multiple diagrams. Contents of the repository can be browsed using UML Model Explorer and add existing elements to new diagrams by dragging them from this window onto the design surface. Another interesting and beneficial workbench tool integrated in

Visual Studio 2010 that the proponent used was the Entity Framework. Entity Framework is a Microsoft ActiveX Data Object .NET (ADO.NET) technology extension that allows the creation of the model of the database and make it possible for the Integrated Development Environment (IDE) to automatically create some of the code required to make the connection between an application and the database real. [16] At first, the proponent created the normalized database and its schema to set the relationship among entities using the Microsoft SQL Server Management Studio, another SEW for databases. Figure 4 depicts the database schema of the proposed system.

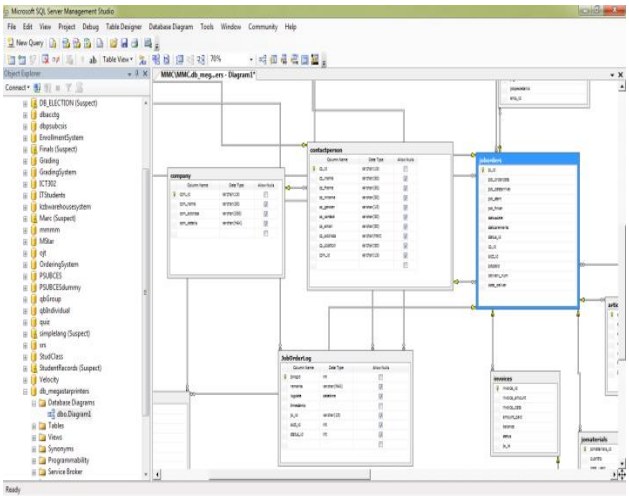

***Fig. 4**. Database Schema*

To be able to manipulate the database through the ASP.Net framework, it is accessed through the Entity Framework tool of the ASP.Net framework. Figure 5 shows how to access databases created from MS SQL Server 2008 DBMS or create a new functional entity diagram or database schema through Entity Framework. Figure 6 shows the entity data model of the proposed system. Entity Framework in ASP.NET made this task significantly simpler. Using Visual Studio 2010, the proponent was able to visually designed entity data models and easily access these models from code allowing the ADO.NET Entity Framework to handle the connections and transactions to the underlying database.

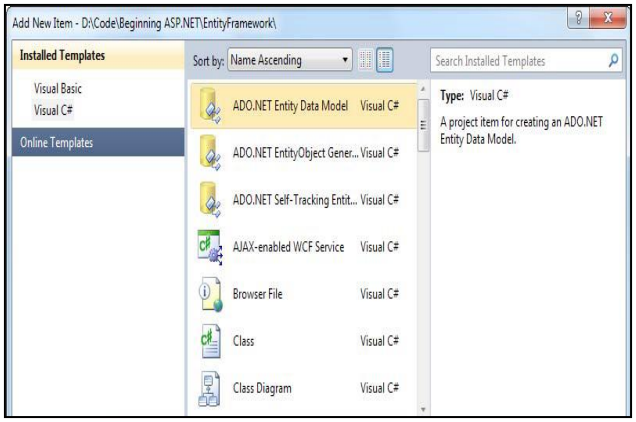

***Fig. 5**. The Entity Framework*

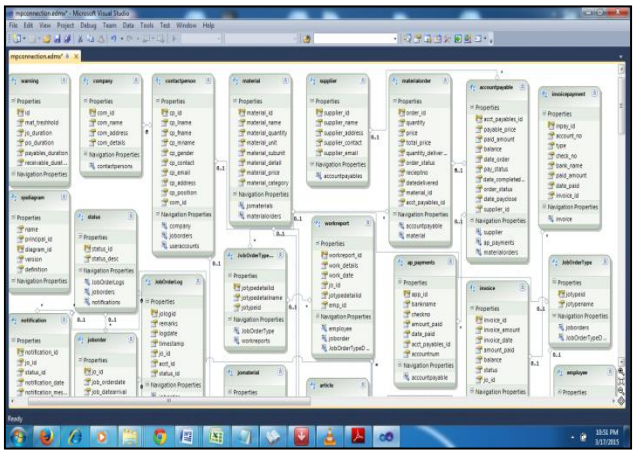

***Fig. 6**. Entity Framework of the Printing Press Production and Sales Management System*

```
public static List<joborder> getall()
{
    MP.db_megastarprintersEntities cont = new MP.db_megastarprintersEntities();
    List<joborder> li = (from s in cont.joborders orderby s.job_orderdate descending select s).ToList();
    if (li != null)
    { return li; }
    else { return li; }
}
```

***Fig. 7**. A C# Code*

Figure 7 shows a C# code using LINQ to query a list of Job Orders with its corresponding company. The proponent did not use complicated joining of tables through Inner Join commands since Entity Framework automatically generates it and abstracted into a class that may serve as a dataset. Figure 8 shows the output of the LINQ query. Furthermore, the proponent made use of MS Visual Studio 2010 to provide auto code generation based on the created UML diagrams.

***Fig. 8**. Output*

5 SYSTEM EVALUATION

The proponent conducted a system usability test that was participated by the management, employees, customers of the

respondent printing press company, and IT experts. The respondents were asked to answer the system usability survey questionnaire. According to the rating of the respondent printing press management and employees, the average weighted mean of the system usability is 6.4 with the interpretation of "Strongly Agree." According to the rating of customers, the average weighted mean of the system usability is 6.5 with the description of "Strongly Agree." According to the evaluation of IT experts, the average weighted mean of the system usability is 6.4 with the interpretation of "Strongly Agree." In terms of usability, the proposed system was rated with an aggregate mean of 6.4 described as “Strongly Agree.”. It implies that the system is strongly high in usability. It also means that the majority of the respondents strongly agree with the capabilities, functions, and the ease of use of the proposed system. Thus, based on the system evaluation, the proposed system is highly acceptable by the respondent management and its customers and some IT experts.

6 CONCLUSION

Based on the findings, it can be concluded that the proposed Online Production and Sales Management System will make the whole production monitoring and sales management process of the respondent printing press company efficient and effective. Both the management and their customers from different locations can simultaneously monitor up-to-the-minute production information (production status and performance) through online browsing without requiring them to visit the office physically. Thus, the management can monitor a real-time key production performance and sales index and plan operation strategically to meet customer needs, while saving time, effort, and money. It lessens the processing time of the assessment of materials used and needed for production. It provides a fast and accurate inventory of raw materials, calculation of bills, tracking of accounts payables and accounts receivables, and generating sales reports. All production and sales reports will support the quality decision-making of the management. Furthermore, the proponent also presented SWEs, which are useful aid in the development of the proposed system. The workbench is integrated into Microsoft Visual Studio 2010 Ultimate Edition. It has powerful UML modeling tools that can be used for conceptual and logical analysis during software development. Its UML Profiles can be used to extend standard models and allow modeling information systems at the physical level. Custom profiles can be developed to tailor UML models to the needs of individual projects. These gave the proponent a powerful workbench for rapid application and model-driven development of the proposed system, where large portions of the system can be modeled in UML and generated with a code generator.

7 RECOMMENDATIONS

Based on the conclusions, it is highly recommended that the respondent printing press company should utilize the proposed system. For a successful implementation of the proposed system, the following recommendations should be considered: The system should be uploaded to the internet to be able to apply its full functionality; There should be higher specifications of the hardware for better performance of the system; There must be an intensive hands-on training of all the users of the system for them to be familiarized with the system’s capabilities and features. The company should have human and material resources to maintain or upgrade the system periodically; and, Other developers can take advantage of using SEWs to achieve competitiveness in the form of faster development of complex Information Systems with improved quality.

ACKNOWLEDGMENT

The authors wish to thank A, B, C. This work was supported in part by a grant from XYZ.

REFERENCES

[1] Business Dictionary.com, “Production,” http://www.businessdictionary.com/definition/production.html, 2018.
[2] AUSPAC, “AUSPAC Corporation Production Monitoring System,” http://www.auspac.com.my/auspac/index_main.html, 2018.
[3] R.D. Hisrich and R.W. Jackson, “Sales and Sales Management, “Barons Education Series, inc. New York 117782, 1993.
[4] Wikipedia, “Computer-Aided Software Engineering,” http://en.wikipedia.org/wiki/Computeraided_software_engineering#Workbenches, 2018.
[5] A. Fugetta, “A Classification of CASE Technology,” IEEE Computer, Vol. 26, No. 12, pp. 25-38, 1993.
[6] M. Rouse, “CASE (Computer-Aided Software Engineering),” http://searchmanufacturingerp.techtarget.com/definition/CASE-computer-aided -software-engineering, 2018.
[7] D.S. Laar, “Design and Development of a Sales Management System for SMEs in Northern Ghana,” International Journal of Innovative Research in Advanced Engineering (IJRAE), Issue 5, Vol 3, pp. 66-77, ISSN: 2349-2163, Oct 2011.
[8] N. Achuth and G. Shashi Kiran, “Analysis and Design of Computerized Production Management System for Garment Manufacturing Company,” International Journal of Engineering Research & Technology, Vol.3 Issue 3, pp. 66-71, ISSN: 2278-0181, March 2014.
[9] D. Panagiotou and Mentras,” A Knowledge Workbench for Software Development.”
[10] Eelco Visser, et.al,” A Language Designer’s Workbench.”
[11] L.Ladan Tahvildari and K. Kontogiannis, ”A Workbench for Quality-Based Software Re-engineering to Object-Oriented Platforms.”
[12] L. Bentley and J Whitten, System Analysis and Design for the Global Enterprise,7th Edition, McGraw-Hill New York, USA, 2007.
[13] T. Runice and D. Mark, Making Effective Business Decisions Using Microsoft Project, John Wiley & Sons, 2013.
[14] MSDN, “UML Class Diagrams: Reference,” https://msdn.microsoft.com/en-us/library/dd409437.aspx, 2018.
[15] Oleg Sych, “UML Modeling in Visual Studio,” http://www.olegsych.com/2010/01/uml-modeling-and-code-generation-in -visual-studio-2010, 2015.
[16] M. MacDonald, “Beginning ASP.Net 4 in C# 2010,” Apress, United States of America, 2010.