Available online at https://jcst.rsu.ac.th
Formerly Rangsit Journal of Arts and Sciences (RJAS)

Journal of Current Science and Technology, January-June 2018 JCST Vol. 8 No. 1, pp. 21-31

ISSN 2630-0583 (Print)/ISSN 2630-0656 (Online)

Design of house bookkeeping software components based on separation of concerns

Meennapa Rukhiran[1*] and Paniti Netinant[2]

[1]Department of Social Technology, Ragamangala University of Technology Tawan-OK, Chanthaburi 22210, Thailand
[2]College of Digital Innovation and Information Technology (DIIT), Rangsit University, Bangkok 12000, Thailand

*Corresponding author: paniti.n@rsu.ac.th

Submitted 13 March 2018; accepted in final form .. June 2018
Available online 29 June 2018

Abstract

Separation of concerns is one of the main design principles in software development. Each section in the design addresses a concern of the system within it. Aspect-Oriented Approach (AOA) is mainly applied for supporting software design in many software applications. The most significant purpose of the separation of concerns is to handle adaptability and flexibility of the system. In this paper, we have designed a fine granularity of a House Bookkeeping Conceptual Framework based on sets of data, functional data, and aspect elements of layers. Composition of concerns in the system can achieve a better modularity of the system. A weaver model is an execution of integrating three concerns that they are relatively separated. A dynamic weaving is provided supporting adaptive and extensive changes in concerns during a run-time. The logical quantifiers of personal finance software development are introduced to express an infinite series of three-dimensional data sets. An Aspect Oriented Approach for supporting house bookkeeping software design based on a prototyping model is proposed to describe a process for analyzing, designing, implementing, and reviewing through a unique division of the fine granularity as well.

Keywords**:** *aspect elements, house bookkeeping, personal finance, separation of concerns, software development*

1. Introduction

According to a worrying rise, household debt has become a major economic and social concern in Thailand, especially with agricultures (Klinkajorn, & Runkawe, 2013; Intachom, 2011; Rungsinpinya, 2011). Household bookkeeping or bookkeeping is recommended for improving the diversity and the resilience of farms and decreasing the debt-and-poverty cycle by many researchers. Investigations into how to deal with this rise in debt should focus on many areas such as reasons for being in debt, categories of spending and receiving budget, and the ability of debt payment. Furthermore, The National Economic and Social Development Plan Issues 8 to 11 of Thailand have applied the philosophy of sufficiency economy as a main factor to drive the country's economic and social development. This philosophy is based on a fundamental principle of Thai culture which has been developed by King Bhumibol Adulyadej. Doing an account, or house bookkeeping is applied to reduce costs and manage debts more effectively in a sustainable way. In addition, a training program on doing household bookkeeping is suggested and provided to help farmers record their incomes and expenditures (Klinkajorn, & Runkawe, 2013; Rungsinpinya, 2011).

Accounting is important for business and personal purposes. The main factors of accounting consist of recording, classifying, summarizing and communicating economic information such as relevant transactions, financial statements and events for making decisions and informed judgments (Silva, Neto, Garcia, Trinta, & Assad, 2013; Leiwy, 2015). An account simply records the financial inflows and outflows in a relationship among assets, liabilities, incomes, and expenses. These can refer to the elements of financial statements on completion from one period to another (Nickels, McHugh, & Mchugh, 2012) and the facilitated solution and effective resolution of financial controls can be manipulated through a good system. In addition, Abdul-Rahamon and Adejare (2014) have mentioned that the purpose of house bookkeeping is to keep track of incomes and expenses. The recordings of the transactions can improve the probability of making a profit and

collect the financial information for filling various tax returns.

A lot of traditional accounting tasks that deal with recording and processing incomes and expenditures are recorded manually. At the same time, the role of information technology (IT) (Lanjuni, 2012) has become an important part of human life over the last few decades. Ejiaku (2014) has commented that the adoption of IT is to manage demands more efficiently and effectively in many activities, including personal finance. Software and mobile applications for house bookkeeping are being continually developed. IT and accounting systems are the major component of accounting research. While it is widely acknowledged that IT plays an important role in the field of accounting, the relationship between IT and accounting has rarely been studied. The accounting and bookkeeping software are analyzed and designed following different user requirements specification. There are many types of application software that can be developed to keep records of financial statements such as offline software, web applications, mobile applications, and enterprise applications. However, there are limitations of software maintainability which emphasize over modularity, extensibility, collaboration, and reusability (Lee & Bae, 2004; Somantri, Nugroho, Widywan, & Ashari, 2015). A separation of concerns is one of the software evolutions to bring the concept level down to the design and implementation levels. By considering many requirements at several different places in a program, the techniques of decomposition can be separated to a single module such as a component, class, method and function. Any changes can be modified to improve the system and add functionality without any effect on a running time.

We have recently proposed the house bookkeeping software design for supporting Aspect-Oriented Approach (AOA) by designing the execution of separating concerns (Rukhiran & Netinant, 2017a) and illustrating the software development stages using prototyping (Rukhiran, & Netinant, 2017b). In this paper, we focus on the conceptual framework of the house bookkeeping software design using three-dimensional layering based on the separation of concerns. Our design is based on the principle design of three dimensions of open layered aspect oriented operating systems (Constantinides, Elrad, Fayad, & Bader, 2000; Netinant, 2015). The aim of our framework is to illustrate the relationship among incomes, expenditures, and liabilities which are separated relatively into a set of dimensions. The time series of multi-dimensional layering is also explained as an example of the solution ideas. The execution design of the separation concerns is illustrated basically through an Aspect Oriented Software Development. The software functionalities are decomposed into smaller aspect elements. In addition, we also present the scope of house bookkeeping software in order to achieve the better understanding of analysis and design phases.

2. Literature reviews

2.1 Three-Dimensional layering

The Cambridge Dictionary (2013) defines a three dimensional as the design of having and appearing of three dimensions: length, width and height. Figure 1 shows a cube, which has six faces of the same size, eight vertices and twelve edges. A face is the flat surface on a solid figure. An edge is the side where two faces come together. A vertex is the corner of the shape where three or more faces come together. Each cube has a top layer and a bottom layer.

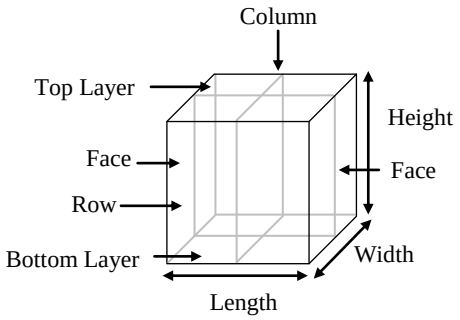

Figure 1 Three-dimensional channel

3-D means three dimensions that consist of width (X), height (Y) and depth or length (Z) as shown in Figure 2. We are able to perceive the spatial relationship between objects just by looking at them because of 3-D perception, also known as depth perception. When they look around, the retina in each eye forms a two-dimensional image of the surroundings and the brain processes these two images into a 3-D visual experience (Media College, n.d.).

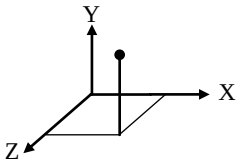

Figure 2 The coordinate system X, Y and Z axis of three dimensions

2.2 Adaptability of framework

Adapting a framework for developing software has been extensively studied in many different researches. Netinant (2015), Pukdesree and Netinant (2016), and Constantinides et al (2000) have designed Aspect-Oriented Frameworks which are based on the three-dimensional model (Netinant, 2015; Pukdesree & Netinant, 2016; Constantinides et al., 2000). The system design consists of components, aspects, and layers. The different aspects of each component are performed by crosscutting the basic functionalities of a system. The design can achieve the consistency, the stability, and the separation of concerns. The framework architecture is divided into two frameworks based on two layers: a base framework on a low layer and an application framework on an upper layer. The result mentions that the upper layer can generate more than one application.

In particular, the new scenario has required a framework to become more adaptable. In fact, adaptation is recognized as one of the most crucial factors for gaining and retaining competitive advantages (Stefanelli, Bevilacqua, & Sanctis, 2015). Carro, et al. (2016) have mentioned that the adaptability in e-Madrid project may occur when a change is requested by a new requirement. They have developed the different methodologies and processes to support the adaptation of the system. Concepts of adaptive systems are being developed on mobile devices. The application services are provided in a distributed way through new ideas and theories. The combination of rules is considered in the research on automatic, interoperable, programmable, and re-usable system (Organero, Kloos, & Merino, 2010; Munoz-Merino, Kloos, Munoz-Organero, & Pardo, 2015).

Moreover, the theoretical framework is proposed to interpret Service-Oriented Cross-layer Infrastructure for Distributed smart Embedded devices by Cannata, Georasa, and Taisch (2008). There are two purposes for developing software: to explicate the effective tools and methods, and to achieve the flexible, reconfigurable, scalable, and distributed embedded system. The new application and the mobile equipment assistance are important keys for this implementation that can support the flexible and dynamic way in a business process.

2.3 Separation of concerns

The principle key of the separation of concerns is to decompose the whole consideration of software design, including components, functionalities, and processes into the distinct feature, named aspect. The separation of concerns is a design for improving the reuse of existing functionalities, components, and maintainability (Benedi, 2006; Microsoft, 2008). In addition, multidimensional separation of concerns is described using UML (Unified Modeling Language) through the use of different diagram types that can crosscut the logical concerns (i.e. high-level design and technology-independent) in a system model (France, Ray, Georg, & Ghosh, 2004). The AOA has proved the separation of concerns through crosscutting concerns. AOA has clearly defined two kinds of software units: a component and an aspect. The functional decomposition is used to define the component, and the performances and semantics of components are encapsulated to define the aspect in a systemic way (Kiczales et al., 2001). Netinant and Elrad (2016) have proposed the separation of concerns for supporting mobile system developments. The Communication Closed Layers (CCL) is a formal system that explains characterizations of the semantic constraints among the components of a mobile system. The description of separation uses philological notation to express the processes by decomposing the components into two different layers.

3. Our house bookkeeping conceptual framework (HBKF)

House bookkeeping or bookkeeping is applied in many fields, according to different personal finance or business purposes. Basically, house bookkeeping is to record financial transactions during a particular period of time. House bookkeeping involves the financial management for an individual or a family in saving, and spending money. Both house bookkeeping and accounting are essential for

running business functions. It is claimed that the differences are that bookkeeping is responsible for recording of financial transactions while accounting is responsible for interpreting, classifying, analyzing, reporting and summarizing financial data. Moreover, accounting involves interpreting and analyzing data but house bookkeeping does not (Integrated CPA Group, n.d.).

Many researchers have been applying the separation of concerns to solve software problems like modularity, cost of maintenance, duplication, scattering, and reusability. However, we have not found any evolution of the Aspect-oriented paradigm using the separation of concerns in order to improve software qualities in house bookkeeping software. We have conducted many common keywords such as house bookkeeping, budget, account, expense manager, and daily money considering on the rating system of Google Play, and have found that the highest view rate is Microsoft Excel (ME), AndroMoney (AM), and Money Lover (ML), respectively. By comparing executions of these mobile applications and our framework (HBK), Table 1 shows the comparison of the functionalities as follows:

Table 1 The comparison of the functionalities

Existing Mobile Applications Functionalities	Microsoft Excel	AndroMoney	Money Lover	Our HBK
Income records	✓	✓	✓	✓
Expenditure records	✓	✓	✓	✓
Liability records	×	×	×	✓
Balance calculation	✓	✓	✓	✓
Income category	✓	✓	✓	✓
Expenditure category	×	×	×	✓
Liability category	×	×	×	✓
Proof of payment	✓	✓	✓	✓
Calendar display	✓	✓	✓	✓
Comparison statement	✓	✓	✓	✓
Notification	×	✓	✓	✓
Functional Extension	×	×	×	✓

We have recently divided the principle of house bookkeeping into three concerns (income, expenditure, and liability) as shown in Figure 3. Firstly, income concerns considering from sources of money consist of Earned Incomes (EI), Portfolio Incomes (PoI) and Passive Incomes (PaI). The categories of EI are working, owning a business, consulting and gambling. PoI is trading paper assets, and selling real estate. PaI is rental income, bonus, insurance, retirement, interest and stock. Secondly, expenditure concerns from types of spending money are Daily Expenses (DE), Personal Expenses (PE), House Expenses (HE) and Family Expenses (FE). DE consists of food and transportation. PE is clothing, travel, sport, book and social & entertainment. HE is mobile & internet, repairing equipment and parking. FE is made up of tuition fees, alimony, medical fee and donations. Finally, the categories of liability concern are current liabilities (CL) and long-term liabilities (LL). CL is separated into car loans, credit card debt, home equity loans, interest, taxes and rental mortgage. LL is bonds payable, notes payable, bank loans, deferred revenue and mortgage.

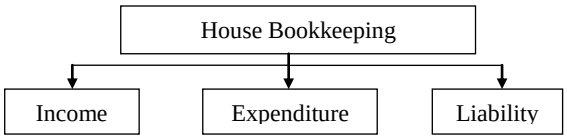

Figure 3 The contents of house bookkeeping

The design of House Bookkeeping Conceptual Framework (HBKF) is based on those concerns in Figure 3. The coordinate system X, Y and Z axis based on the three-

dimensional model in Figure 2 are focused on expenditure, income, and liability concerns, respectively. The records of transactions are summarized separately using three different concerns. Our HBKF is integrated separating concerns in order to realize the relationships of the personal financial statement for saving expenses, avoidance of debts, and increasing earnings.

In Figure 4, the design of HBKF is separated using three-dimensional layering to present the relationships among sets of data. There are three sets of data which are relatively separated. The scope of the granularity is to break the larger partitioning into subdivisions. Each set of data is divided into smaller components. All three of our concerns are categories to the subdimensions for defining smaller elements. The principles of the separation concerns have been described in the previous paragraph. The functional data is a crosscutting concern through sets of data. Each dimension can express a set of dimensions. We let a set of functional data = {{expenditure}, {income}, {liability}}. Functional data (n) = {n ∈ F; n > 1}. n is the number of transaction recordings.

However, it depends on a requesting purpose of the data selection. The data can take an action in one-dimensional layering or more. For example, a one-dimensional approach can be selected all recordings of expenditures. A two-dimensional approach may be used to compare the summarization of incomes and expenditures. A three-dimensional approach is selected to compare financial statements through three records. Therefore, different situations of its financial statement report can be adjusted dynamically such as by selecting on a screen once an amount of income changes a report from each week to each month.

Based on the traditional accounting tasks, Yang (2011) has mentioned that it is necessary to concern variations of the time period that can be related to record transactions. We express the time function using the principle design of multi-dimensional layering to represent the time series of records. For an example of the time function, a component is divided into a day layer. The day layer has the related series to a week layer. In Figure 5, the time series are applied using multi-dimensional layering. Each layer is divided for indicating the day function. By starting layers from Day1 to Day7, seven

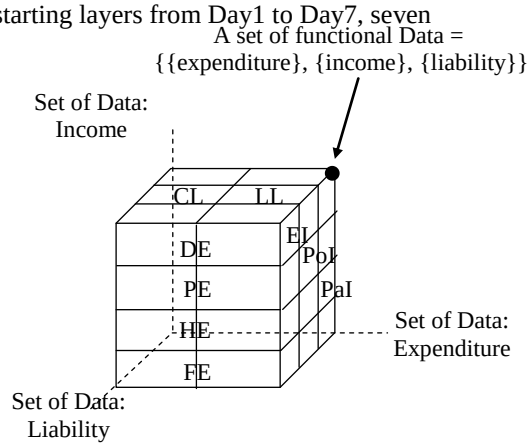

Figure 4 The design of HBKF is based on three-functional data

layers are combined as a week layering known as a time series. In addition, a month layer is provided when the combination of four weeks is required by the system. For instance, the depreciation and salary allocation can be changed from daily to weekly as well as monthly.

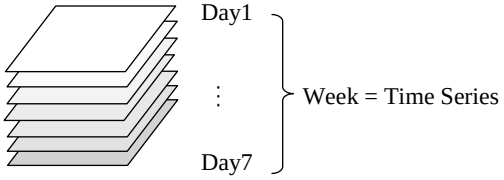

Figure 5 The time series in multi-dimensional layering

4. The separation of concerns

We have designed the House Bookkeeping Conceptual Framework (HBKF) shown in Figure 4. The framework has created the individual recordings by dividing concerns (income expenditure, and liability records) in three-dimensional layering. Data is assigned for each plane in a dimension, and the crosscutting point which is related to different category concerns, named functional data. The design can reduce a number of the call requests in a transaction because the functional components of the dimensions are decomposed. A loosely coupled relationship is an important key in making the aspect elements less interdependent. Moreover, high-cohesion focuses on assigning a single functionality to each aspect. Each element is assigned different functionalities by discerning

from the elements which cut across many basic data. For example, an object-oriented design allows classes to inherit interfaces from another class and a complex hierarchy of classes is possible. Layering design of interfaces must be declared as the operations of any interfaces of a class are independent from each other. On the other hand, in an aspect-oriented design, an aspect is a behavior that cuts across through multiple objects and methods. A single behavior can apply to multiple classes. These concerns avoid the tangled methodology and allow the reuse of the same aspect with duplication (scatter). Therefore, our HBKF can be extended to provide support for the separation of concerns differently. We have proposed the samples of aspectual properties as shown in Figure 6.

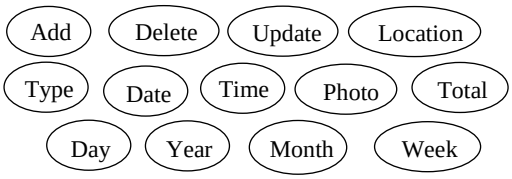

Figure 6 The thirteen aspect elements are assigned using the principle key of separation concerns

In Figure 6, the thirteen small functionalities of the aspect elements are designed separately. The aspectual properties are excluded from the functional data. The aspect element is defined as a set of computational properties. The aspects (e.g., add, update, delete, week, and sum) are generated separately from the system functionality and methodology. Both functional data and aspect elements can define as an infinite series of executions.

A composition rule of the three-dimensional layering is defined as the combination of semantic data (derivation dataset) from one dimensional layering or more. For instance, a dataset of incomes is assigned to a vertical dimension. The requirement of income records can be called to report only an amount of incomes. The comparison of a financial statement can represent an amount of incomes and expenditure by using the functional data that is a crosscutting point between two-dimensional layering. We assign the semantics of the functional data to a set of house bookkeeping concerns in the database table names by expressing an infinite series from 1 to n. In fact, the table name is a database table name that relates to the functional data. The functional data set F U {F1, F2, F3, …, Fn}. The aspect element is a sequence of methodologies which starts corporately more than one aspect to m aspects. An aspect set A U {A1, A2, A3, …, Am}. An object is an execution of calling the aspect elements and the functional data using crosscutting concern. We assume a weaver to call the object for the final execution by using the functional formula n x m for crosscutting concerns as shown in Figure 7 (Rukhiran & Netinant, 2017a). Weaving is a process of transforming in order to solve scattered solution and avoid tangled methodology.

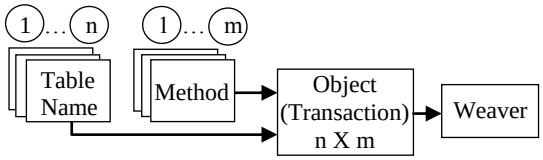

Figure 7 The execution design of the separation concerns through Aspect Oriented Software Development

We have mentioned that the aspects have been excluded from the functional data. We assume the object as an execution of calling the aspectual properties and the functional data through crosscutting concern. The weaver can call the object to execute corporately. The aspect encapsulates the crosscutting concern known as join point. The join point defines the method of execution. The execution is handled through the different semantics that depend on the object that has invoked the method (Benedi, 2006). Therefore, there are many join points that across cutting together in an object. This across cutting is known as a point cut. The point cut is defined as a pattern of matching join points. The weaving is performed using the point cut by taking join points between the functional data and the aspect. The execution flow uses an adviser to decide the method calls that should be used based on particular join points. The advice provides types of behavior to execute when a join point is captured at a running time.

Figure 8 shows an example execution of crosscutting concerns. The aspects (e.g. add, type, photo, and total) take an action through a transaction named shortly t1. A set of incomes in one-dimensional is called by the object. The crosscutting concern is exactly required in order

to execute the aspects and the functional data. The functional data presents a set of dimensions in a hyperspace approach. Hyperspace approach is defined as a space of the multi-dimensional matrix to contain specific concerns of a dimension (Benedi, 2006). The hyperspace of this design is a concern space in the three-dimensional layering that can adapt using the multi-aspect and multidimensional data.

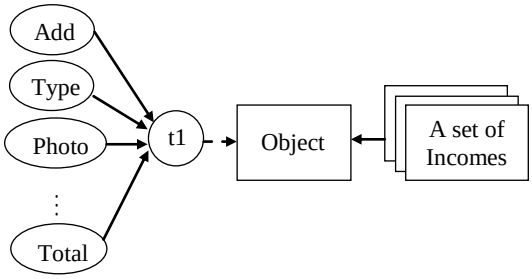

Figure 8 An execution design of the aspects in one-dimensional layering of income records

5. The scope for software development using HBKF

The analysis scope is composed of four practices: physical level relates to the specifics of HBKF for supporting software design; system level provides the comprehensive model for transforming the concept design into the implement phase; information level explains the purposes of adapting HBKF as a common software for improving user's financial statement, and business process level presents the final gold of this research. The practices relate to information system development of house bookkeeping. The design is increasing a personal efficiency economy in different levels as follows:

1. Physical level: The design is an open source framework for supporting the development of software and mobile application for all devices such as offline program, online system, and mobile platform. It provides the suitable design through the conceptual framework from the analysis phase to implement phase.

2. System level: The contents of house bookkeeping is divided into three concerns (income, expenditure, liability records). Each one is separated relatively into smaller dimensions. An adaptable software is able to reconfigure when the set of requirements changes. The functionality of software can be reusable and adjustable because the fine granularity is defined by a set of data, functional data, and aspect elements of layers. The new features are welcome additions at a running time and a compiling time. We also design the basic functionalities and abilities of house bookkeeping for an effective usage as follows:

1). Record transactions of income, expenditure and liability daily by categorizing of their aspects systematically.

2). Track personal finance statements adaptively for day, week and, month by illustrating via calendars, charts and graphs.

3). Manage info graphics of each category to categorize easily.

4). Remind recordings of transactions and payments automatically.

5). Transfer recordings of liabilities to expenditures and reduce the total of incomes separately.

6). Notify the chronological payment date.

7). Backup data and synchronize files.

3. Information level: The purpose of this design is to achieve knowledge-based information of a personal finance statement by designing the conceptual framework of household bookkeeping. The HBKF can lead to develop an information system for supporting any platforms. Therefore, the financial transactions are able to record systematically. Each transaction is classified into the correct category. Both short-term and long-term liability can be managed effectively. The most important settlement is computed and planned orderly. Assets such as deposit, balance and interest are reported, and the personal finance statement reflects the economic substance.

4. Business process level: There is rising awareness that an adaption of IT can make more benefits available in individual and family financial statements. The result of using the house bookkeeping software is helpful to reduce expenditures and avoid informal debts from a loan shark. The awareness of spending on unnecessary items is recognized increasingly. Significantly, behavioral economics is changed by saving money and changing habits. The final outcome is for a practical approach toward the sustainable development of Philosophy of Sufficiency Economy. Thus, to do house bookkeeping may become more valuable in the

current situation of economic recession and future roles.

6. Discussion

Throughout this article, we have designed the House Bookkeeping Conceptual Framework (HBKF) using three-dimensional layering. The artifact is based on the separation of concerns, including the functional data and the aspect elements. By dividing the contents of house bookkeeping into three concerns, we have purposed the sets of dimensional data (income, expenditure, and liability records). The design of the hyperspace is a concern space of the three-dimensional layering. The layers can be semantically adapted (Netinant & Elrad, 2016) by multi-aspect and multidimensional data. The functional data is defined as an execution of the crosscutting concern between the sets of data in order to take a data derivation through the multi-dimensional. The conceptual framework of the separation concerns illustrated through Aspect Oriented Software Development enables adaption using Software Development Life Cycle (SDLC) (Massey & Satao, 2012; Isaias & Issa, 2015; Rukhiran, Bunpalwong, Boonsong, & Promput, 2017). For example, the implementation phase, the different aspect of interactions can be adjusted easily without affecting the structural and behavioral diagrams. In the deployment phase, the interface design is friendly, used for the dynamic user response. The dynamic change of the output is caused by a change in the input. This approach can also support a better extension and can easily achieve the adaptability in the architectures of software development.

Due to the popularity of mobile devices, smartphones and tablet computers are increasingly used at present. We recommend using this HBKF to develop on a mobile application. In this case, it can be handled similarly to Carro et al. (2016) and Cannata et al. (2008) as they have analyzed the user problems and found the solution by design software on mobile devices.

7. Conclusions

The digital trends of Information and Communication Technology (ICT) present a new main factor of software design and development. Application software is creating useful and convenient artifacts on laptop and smart phone platforms. While most people traditionally used to record financial transactions, a current change is occurring in human behaviors. Personal computers and portable devices such as smart phones and tablets provide more efficient benefits of recording financial transactions. Technology trends have enhanced the better capabilities of controlling real-time, tracking balance, notifying payment, and prevented the forgetting of records.

The new finding of House Bookkeeping Conceptual Framework is distributed through this research. The architecture analyzes using the principle of the dimensional layering by separating the disciplines from incomes, expenditures and liabilities. At this level, an individual category can indicate differently. Multi-dimensional layering represents relativity in the time-series. The day layer is a small element that can fill in the subdimension. The execution can integrate multi layering to present the week layer. Thus, the advantage of three-dimensional layering can truly promote software adaptability, manageability, extensibility, and maintenance.

This HBKF seems to be able to support all stages of Software Development Life Cycle. The methodology has only proposed the conceptual design in the three dimensions layering. For the future research, to achieve the aim of philosophy of sufficiency economy, we intend to develop a mobile application by using HBKF as a design guideline. Due to the adaptability of HBKF, the contents should focus on the execution of a component model for effective software maintenance and evolution. The components should be specified clearly in the areas of the characteristics, functions, and structures.

8. References

Abdul-Rahamon, O., & Adejare, A. (2014). The analysis of the impact of Accounting Records keeping on the performance of the small scale enterprises. *International Journal of Academic Research in Business and Social Sciences*, *4*(1), 1-17.

Benedi, J. P. (2006). *PRISMA: Aspect-Oriented Software Architectures*. (A Thesis for the degree of Doctor of Philosophy in

Computer Science. University of Valencia, Spain).

Cambridge Dictionary. (2013). *Three-dimensional*. Cambridge advanced learner's dictionary. Cambridge, UK: Cambridge University Press.

Cannata, A., Gerosa, M., Taisch, M. (2008). SOCRADES: A framework for developing intelligent systems in manufacturing. *Proceedings of the IEEE International Conference on Industrial Engineering and Engineering Management*. December 8-11, 2008. Singapore, Singapore. pp. 1904-1908. DOI: 10.1109/IEEM.2008.4738203.

Carro, R. M., Molins-Ruano, P., Rodríguez, P., Sacha, G. M., Delgado-Kloos, C., Muñoz Merino, P. J., . . . Martín, S. (2016*).* eMadrid project: Ubiquitous learning, adaptation, adaptability and accessibility. *Proceedings of the IEEE International symposium on computers in education*. September 13-15, 2016. Salamance, Spain. pp. 1-4. DOI: 10.1109/SIIE.2016.7751871.

Silva, F.A.P., Neto, P. A. M. S., Garcia, V. C., Trinta, F. A. M., & Assad, R. E. (2013). Monext: An Accounting Framework for Infrastructure Clouds. *Proceedings of the IEEE 12th International Symposium on Parallel and Distributed Computing*. June 27 – 30, 2013. Bucharest, Romania. pp. 26-33. DOI: 10.1109/ISPDC.2013.13

Ejiaku, S. A. (2014). Technology Adoption: Issues and Challenges in Information Technology Adoption in Emerging Economies. *Journal of International Technology and Information Management*, *23*(2), 59-68. http://scholarworks.lib.csusb.edu/jitim/vol23/iss2/5

France, R., Ray, I., Georg, G**.**, & Ghosh**,** S. (2004). Aspect-oriented approach to early design modeling. *IEEE Software, 151*(4), 173-185. DOI: 10.1049/ip-sen:20040920.

Intachom, N. (2011). *Study the Impact Liability of The Military Mind Department to support the 9th Infantry Brigade*. (A Thesis for the degree of Master of Public Administration in Public Administration. Graduate School Silpakorn University, Thailand).

Integrated CPA Group. (n.d.). In house accounting & bookkeeping. Retrieved November 9, 2017, from http://integratedcpa.integratedfg.com/in-house-accounting-bookkeeping.

Isaias, P., & Issa, T. (2015) *Introduction to information system models and methodologies*. High Level Models and Methodologies for Information Systems (pp.1-19). London: Springer-Verlag. DOI: 10.1007/978-1-4614-9254-2_1.

Kiczales, G., Hilsdale, E., Huguin, J., Kersten, M., Palm J., & Griswold, W. G. (2001). An Overview of AspectJ. *Proceedings of the 15th European Conference on Object-Oriented Programming*, Lecture Notes in Computer Science (LNCS), Springer-Verlag Press. June 18-22, 2001. Budapest, Hungary. pp. 327-354.

Klinkajorn, S., & Runkawe, N. (2013). Factors that Influence to Debt of Agriculturist in Nakon Ratchasima Case Study of Dankhunthot and Nonsong District. Retrieved November 9, 2017, from http://narinet.sut.ac.th:8080/xmlui/bitstream/handle/123456789/210/factors.pdf?sequence=1&isAllowed=y.

Lanjuni, L. (2012). Study on the Reconstruction of Accounting Information System based on the Financial Accounting Conceptual Framework. *Proceedings of the IEEE World Automation Congress*. June 24-28, 2012. Puerto Vallarta, Mexico. pp. 1-4.

Lee, J. S. & Bae, D. H. (2004). An aspect-oriented framework for developing component-based software with the collaboration-based architectural style. *Information Software Technology*, *46*(2), 81-97. DOI:10.1016/S0950-5849(03) 00111-3.

Leiwy, D. (2015). *Principle of accounting*. London, UK: University of London International Programmes Publications Office.

Massey, V., & Satao, K. (2012). Comparing various SDLC models and the new proposed model on the basis of available methodology. *International Journal of Advanced Research in*

Computer Science and Software Engineering, *2*(4), 170-177.

Media College. (n.d.) What is 3D? Retrieved November 9, 2017, from http://www.mediacollege.com/3d/intro.html.

Microsoft. (2008). *Mobile application architecture guide*. Retrieved November 9, 2017, from http://robtiffany.com/wp-content/uploads/2012/08/Mobile_Architecture_Guide_v1.1.pdf.

Munoz-Merino, P. J., Kloos, C. D., Munoz-Organero, M., & Pardo, A. (2015). A software engineering model for the development of adaptation rules and its application in a hinting adaptive e-learning system. *Computer Science and Information Systems*, *12*(1), 203-231. DOI: 10.2298/CSIS140103084M.

Netinant, P., Constantinides, C. A., Elrad, T., Fayad, M. E., & Bader, A. (2000). Supporting the design of adaptable operating systems using aspect-oriented frameworks. *Proceedings of The International Conference of Parallel and Distributed Processing Techniques and Applications*. June 24-29, 2000. Nevada, USA. pp. 271-278.

Netinant, P. (2015). Design adaptability for multilingual mobile application software. *Proceeding of the 24th International Conference on Software Engineering and Data Engineering*. October 12-14, 2015. California, USA. pp. 189-195.

Netinant, P., & Elrad, T. (2016). Separation of concerns in designing mobile system. *Rangsit Journal of Arts and Sciences*, *6*(1), 89-96. DOI: 10.14456/rjas.2016.8.

Nickels, W. G., McHugh, J. M., & Mchugh, S. M. (2012). *Understanding Business.* 10th Edition. New York, USA: McGraw-Hill Education.

Organero, M. M., Kloos, C. D., & Merino, P. M. (2010). Personalized service-oriented e-learning environments. *IEEE Internet Computing*, *14*(2), 62-67. DOI: 10.1109/MIC.2009.121.

Pukdesree, S., & Netinant, P. (2016). Conceptual framework: The adaptive biometrics authentication for accessing cloud computing services using iPhone. *Advances in Intelligent Systems and Computing*, *463*, 209-216, Springer-Verlag Press. DOI: 10.1007/978-3-319-40415-8_20

Rukhiran, M., & Netinant, P. (2017a). The concept design of house bookkeeping software using aspect oriented approach. *Proceedings of the 2017 International Conference on Information Technology*. December, 27-29, 2017. Singapore, Singapore. pp. 232-236.

Rukhiran, M., & Netinant, P. (2017b). Aspect oriented approach for supporting house bookkeeping software design. *Proceedings of the 2017 International Conference on Software and e-Business*. December, 28-30, 2017. Hong Kong, Hong Kong. pp. 49-54.

Rukhiran, M., Bunpalwong, M., Boonsong, S., & Promput, S. (2017). Supporting the design of knowledge management system (KMS) for RMUTTO. *Proceedings of the 8th Rajamangala University of Technology International Conference*. August 8-9, 2017. Bangkok, Thailand. pp. 213-226.

Rungsinpinya, P. (2011). Home Accounting. *Journal of Business, Economics and Communications. 7*(1), 20-28. Retrieved from https://www.tci-thaijo.org/index.php/BECJournal/article/view/54656

Somantri, M., Nugroho, L. E., Widywan, W., & Ashari, A. (2015). Design of Agent Framework Using Aspect-Oriented Approach. *Proceeding of IEEE International Seminar on Intelligent Technology and Its Applications*. May 20-21, 2015. Surabaya, Indonesia. pp. 297-302. DOI:10.1109/ISITIA.2015.7219995.

Stefanelli, F., Bevilacqua, M., & Sanctis, I. D. (2015) Adaptability into supply chain strategy: the adaptable PCSA framework. *Proceeding of the IEEE International* Conference *on Industrial Engineering and Systems Management*. October 21-23, 2015. Seville, Spain. pp. 378-387.

Yang, W. (2011). Influence of computerized accounting systems on bookkeeping

conceptual framework. *Proceeding of the 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce*. August 8-10, 2011. Dengleng, China. pp.7100-7102.
DOI: 10.1109/AIMSEC.2011.6010509