Different Kinds of Modern Techniques to Develop Various Information Systems

Feri Sulianta[a], Heri Heryono[b], Ulil Surtia Zulpratita[c], Sy.Yuliani[d], Ai Rosita[e], Eka Angga Laksana[f], [a,b,c,d,e,f]University of Widyatama, Bandung, Indonesia, Email: feri.sulianta@widyatama.ac.id, heri.heryono@widyatama.ac.id, ulil.zulpratita@widyatama.ac.id, sy.yuliani@widyatama.ac.id, ai.rosita@widyatama.ac.id, eka.angga@widyatama.ac.id

Basically, information system design is fundamental to the long-term sustainability of information systems. Information system architecture is a reference in developing and building new information systems if the current system is no longer able to support the activities of an organization or company. There are various information system development methods and there are various information systems architecture models. The choice of using methods and models must be carefully considered so that the information system created is right on target. A variety of techniques will be introduced to enrich knowledge in choosing development methods and architectural models that suit the needs of the organization.

Key words: *Architecture, Software Engineering, Information System, Method, Model.*

Introduction

Computer technology and information can no longer be separated from human activities, we are very dependent on computers. Humans are very dependent on the existence of information systems in many activities, including in business. If the information system is abolished, it can be ascertained that in a matter of weeks, the activities and business processes will be abandoned. This can also have an impact on the business world, without an information system, companies that are very dependent on the existence of an information system, can no longer operate and even experience bankruptcy. Information systems have been seen as a driving force for human activities and companies (Ahmed et al., 2016;

Berthold, 2003; Feri and Fajri, 2012). There are many information systems used by many companies, this depends on the needs and functionality supported by the information system.

Information System Classification

Some classifications of Computer Based Information Systems were put forward by Kroenke in 1992, namely:

Classification based on organizational structure:

• Departmental Information System: Information systems used by a department in an organization or company.

• Examples: Accounting Information System, Transaction Processing System, Billing System, etc.

• Enterprise Information System: An information system that includes departments within a company and generally is an integrated system that supports various functions of departments in a company.

• Interorganizational System: a system that is able to integrate business processes between companies, this system is generally supported by Electronic Data Interchange (EDI) in exchanging data between different company systems.

Classification based on organization in internal scope:

• Personal information system: this information system is generally used by individuals in a company. Example: various applications included in office automation systems, such as word processing applications, electronic spreadsheets, e-mail client applications and scheduling applications.

• Working group information system: a system aimed at supporting collaboration and work of groups or teams. For example: an application with a teleconference feature.

• Enterprise Information System: a system that refers to integrated information systems used by companies, where various business functions of a company are integrated in this system, for example this system accommodates sales, purchases, financial, accounts payable, and accounting transactions where all components are integrated (Feri and Fajri 2015; Feri, 2017; Feri, 2010; Feri, 2011).

Classification based on the functional area of the company (external and internal functions that exist in the company or organization, sometimes also referring to business processes in departments in the company):

• Accounting Information System

• Finance Information System

• Manufacturing Information System

• Marketing Information System

• Human Resources Management Information System

Classification based on the support provided by the information system:

• Transaction Processing System: the characteristics of this system are supporting repetitive, routine work, essential and critical activities, often used by workers at lower levels or administrative workers. Transaction Processing Systems generally become data sources for other information systems, for example sales transaction data in the Transaction Processing System, used to make accounting reports on Accounting Information Systems.

Example: Transaction system in supermarkets or minimarkets, information systems for borrowing books in libraries, systems for rental vehicles or systems used to manage bills on internet cafes (James, n.d; Jermsittiparsert et al., 2019; Kalloniatis, 2012).

• Management Information System: This system is intended to support the functions and activities of managers and generally middle-level managers. Accumulated transactional reports summarized and displayed in the form of diagrams are one example of the features in the Management Information System.

• Office Automation System: this system is intended to support office workers in their activities and this system also refers to information systems that are used individually or personally.

Example: various word processing applications, electronic spreadsheets, e-mail client applications, scheduling applications, fax machines, timetable applications, e-calendars and e-calculators, etc.

• Decision Support System: this system is intended to assist the role of middle level managers in making decisions, for example banking managers use Credit Approval Systems when analyzing the feasibility of customers to be given loans.

Note: Decision Support Systems are not intended to replace the duties of managers in making decisions, but are intended to provide input for managers in making decisions.

• Executive Information System: this system is intended to assist top managers in strategic planning. The main feature of this application is the feature in digging information (Drill Down Information). Online Analytical Processing and data mining techniques are often used as the basis for building Executive Information Systems.

• Working Group Support Systems: a system aimed at helping the work of a team or group. Example: Lotus Notes Application.

• Artificial intelligence systems or expert systems (Intelligent Support Systems): this system is intended for expert workers such as doctors, researchers or scientists. This application has expertise features or can provide output like an expert. Example: a system that can detect cancer types or systems that are able to analyze radioactive waste.

• Knowledge Information System: this system is intended to manage knowledge, for example: e-learning applications that have discussion forum features and information or knowledge digitalization processes (Kroenke, 1992; Marcus, 1992; McLeod, 1989; Pokorny et al., 2011).

Information System Architecture Development

Designing information system architecture is part of the software engineering process which is then followed up by obtaining various information resources to create information systems. Looking at several points of view, information system architecture is defined as follows as the information expert says:

• Mapping and planning information needs in an organization or company (Turban, McLean, Wetherbe, 1999).

• A certain form of using information technology to achieve goals also functions carefully in an organization or company (Laudon & Lauon, 1998).

• Design of a computer information system as a whole to meet organizational needs (Zwass, 1998).

• Defining information system requirements carefully in the form of a design involving various components of information resources which become a blueprint for building information systems (Feri Sulianta, 2014).

As stated earlier regarding the important role in modeling information system architecture, modeling information system architecture is basically intended to:

• Build an information system that is right on target, effective, efficient and quality.

• Become a reference or blueprint for building information systems.

• Be used as system documentation.

• Be used as a reference in understanding the mechanism of information systems.

• Be used as a basis for maintaining a long-term information system.

• Become the basis for developing a new information system in the future if the previous information system is out-of-date.

As the development of information technology and the creation of software such as word applications or graphic applications improve, system developers use these devices in drawing system architecture models. Drawing models with word applications or graphical applications, is no different to drawing a template flowchart, because it allows the occurrence of a high risk of error, different from using a special application in modeling information systems, this application is known as CASE (Computer Aided Software Engineering) (Pressman, 1997; Radhika, 2013; Rumbaugh et al. 2005; Sommerville, 2011).

Figure 1. Currently modelling information systems is done using modleling software (Computer Aided Software Engineering)
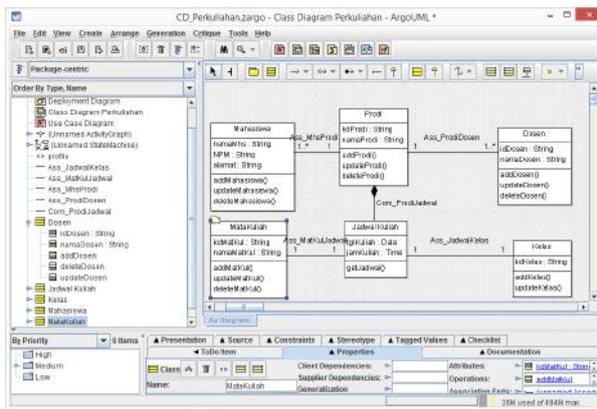
Modelling Tool

Basically modeling is only one part of the steps that must be passed in building an information system whereas currently modeling information system architecture, has been seconded to various modelling software divisions. From the various types of information systems that have been introduced previously, the development of information systems is carried out after going through a series of long processes, one of which is the creation of a system architecture or modelling of the system. There are many ways and many tools that can be used to create information system architectures and these are intended to facilitate in analyzing, documenting, building and maintaining information systems (Zhang, 2017).

The ease of building a system architecture effectively and efficiently is now enabled by the CASE Tool software tools. The CASE Tool includes a Modeling Tool that has various functions such as:
• Model information systems and processes that occur in the system.
• Document information systems
• Provide flexibility and variety of choices in modeling information systems.
• Designing a programming interface (mockup).
• Make work scheduling in designing information systems carefully.
• Automatically generate programming codes.
• Creating a prototype, the information system as a prototype that will be used as the basis for making actual applications.
• Media in communicating and determining client needs regarding the procurement of software.
• Reference in studying the current system and further system development.

Many modeling applications can be used for this purpose. Some applications are categorized as commercial and paid, such as: Rational Rose, IBM RSA, Microsoft Project, Power Designer, Mockup Builder. Some provide a shareware or trial version to be used for a certain period of time, some applications may be used forever with limited features, for example the Balsamiq Mockup application.

Many of these tools are obtained free of charge (freeware) or freely developed and distributed (open source). For example, such as: Gantt Chart Project, Star UML, Argo UML, etc. Each of these applications generally has its own features and advantages that are not found in other applications.

Some of the below are examples of application interfaces used in modeling and developing information systems:

Figure 2. Power Designer 6.0 is used to make data flow diagrams (DFD Level 1)

Figure 3. Sybase Power Designer ver 12.5 which is used to create 'Conceptual Data Model' which maps in detail the ERD (Entity Relationship Diagram) [Source: Sybase.com]

Power designer applications are intended to model information systems before they are encoded using a programming language. The type of diagram used is known as the term namely: Context Diagram and Data Flow Diagram or Data Flow Diagram. This diagram can be generalized and detailed to reveal business processes in detail.

Another data modeling application is CA ERwin Data Modeler for modeling databases, this application also has documentation features, it supports object-based modeling, and the modeling results can be transformed into various other modeler tools. This application supports database design such as: DB2, IDS (Informix), MySQL, ODBC, Oracle, Progress. SAS, SQL Server, Sybase, Sybase IQ, and Teradata.

Figure 4. Interface of CA ERwin Data Modeler [Source: Erwin.com]
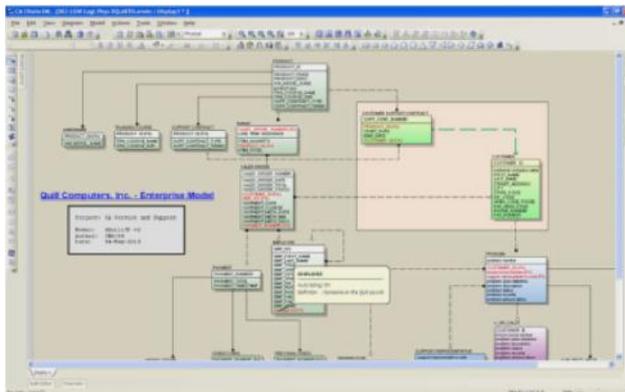
Figure 5. UML Argo used to create a Class Diagram
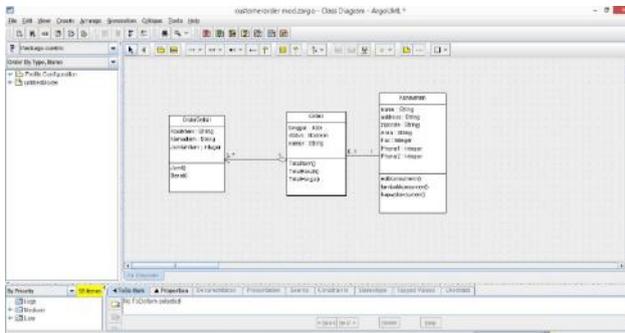
Figure 6. *UML Argo used to create Sequence Diagrams*
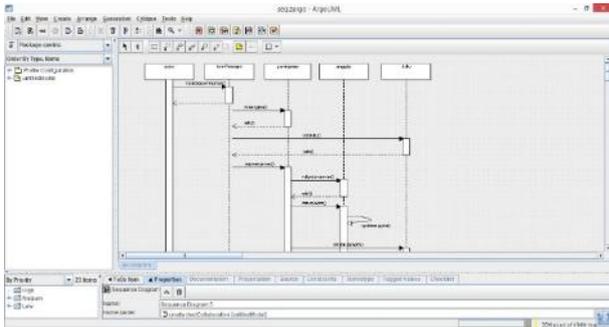
Some of the above images present the creation of models such as Use Case, Class Diagram, and Sequences Diagram. These models are models that are made if the user wants to build an application based on object-based modelling. Modelling tools can generally be further empowered to generate programming code, as exemplified in Figure 7 below about transforming Class Diagrams based on objects into programming codes.
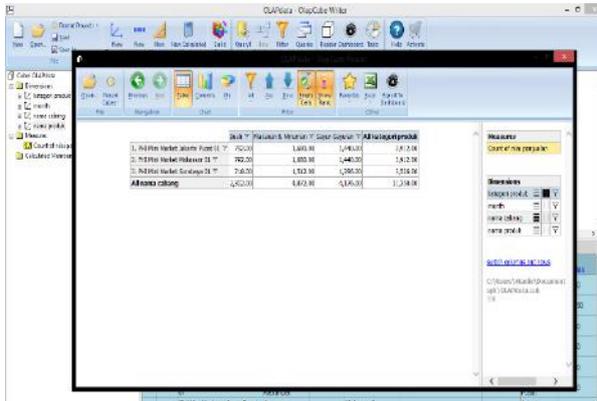
The development of various CASE Tool applications greatly facilitates developers in building information systems. Furthermore, CASE Tool was also intended for a broader need, managing the creation of information systems, which involves scheduling, managerial teams, system maintenance and even making information system prototypes that could be operated.

Specific Developer Tool
CASE Prototyping Tool in Data Drilling

One type of system known as the Executive Information System generally has the ability to drill down (dig) information. With the abundance of software products today, without having to model or write programming code, system analysts or programmers can first experiment with applications such as data mining and OLAP tools to study data and the need for 'strategic applications for the needs of top managers' which will later made. The OLAP tool, used to experiment with data before creating an Executive Information System, can be seen as modelling and is included in the design phase of system development. The following is an example of the OLAP Cube Application which is able to draw data and represent it in graphical form:

Figure 7. *Analyzes the sales value of product categories in each branch*

Others Software Engineering Tool

Computer-based information systems do not have to always be built by typing programming codes. If the application is intended only to handle work whose scope is limited and is only intended to 'support' individual work in offices (known as the Office Automation System), it is not recommended to use large resources only to meet the needs of information systems with a small and simple scope. Some end user application products have been able to build applications to meet needs with limited scope.

Such applications actually have good performance, especially in terms of user friendliness, for example the Microsoft Excel application which has the following features:
• Macro Programming and Visual Basic Application.
• Arithmetic functions ready to use.
• Generator diagram (chart generator).
• Worksheets (spreadsheets) that can hold lots of data.
• Other automatic features in managing data in a spreadsheet.
Note: Macro is a small program that consists of a collection of commands that are run in an office package application program environment (including: Excel, Access, Word). Macro is basically designed to make it easier to work with Microsoft Office package applications. Macro commands are actually VBA commands, so you can say VBA is a superset of Macro. While Visual Basic Application is a Visual Basic programming language that is integrated in the application. Inserting Macros and Visual Basic Application on Excel objects is much easier than having to build information systems using Visual Basic.

The following is an example of a transaction information system for recruitment selection made using Excel and VBA:

Figure 8. *Input data in a program created using Excel - the data entered is stored in a separate spreadsheet*

Belbow is a VBA program that is added to the Excel worksheet so that the application can save to a separate spreadsheet that functions like a database:

```
'Kode saat tombol Input ditekan
Private Sub cmdInput_Click()

'wsDTBS adalah worksheet Database
Set wsDtbs = Worksheets("Database")
'BarisLanjut = record database setelah baris terakhir
BarisLanjut = wsDtbs.Cells(Rows.Count, "A"). _
End(xlUp).Offset(1, 0).Row

'Jika nama pelamar belum diisi
If Range("D8").Value = "" Then
    'Kotak pesan informasi nama pelamar belum diisi
    MsgBox "Nama pelamar belum diisi", _
    vbOKOnly + vbInformation, "Nama Pelamar Kosong"
    'Seleksi sel D8
    Range("D8").Select
    'Keluar dari Sub Procedure
    Exit Sub
'Jika umur pelamar belum diisi
ElseIf Range("D14").Value = "" Then
    MsgBox "Umur pelamar belum diisi", _
    vbOKOnly + vbInformation, "Umur Pelamar Kosong"
    'Seleksi sel D14
    Range("D14").Select
    'Keluar dari Sub Procedure
    Exit Sub
'Jika IP pelamar belum diisi
ElseIf Range("D16").Value = "" Then
```

```
        MsgBox "IP pelamar belum diisi", _
        vbOKOnly + vbInformation, "IP Pelamar Kosong"
        'Seleksi sel D16
        Range("D16").Select
        'Keluar dari Sub Procedure
        Exit Sub
    End If

    'Input nama pelamar ke dalam worksheet database kolom A
    wsDtbs.Cells(BarisLanjut, 1).Value = Range("D8").Value
    'Input jabatan yang dilamar ke dalam kolom B
    wsDtbs.Cells(BarisLanjut, 2).Value = Range("D10").Value
    'Input pendidikan pelamar ke dalam kolom C
    wsDtbs.Cells(BarisLanjut, 3).Value = Range("D12").Value
    'Input umur pelamar ke dalam kolom D
    wsDtbs.Cells(BarisLanjut, 4).Value = Range("D14").Value
    'Input IP pelamar ke dalam kolom E
    wsDtbs.Cells(BarisLanjut, 5).Value = Range("D16").Value
    'Input jenis kelamin ke dalam kolom F
    wsDtbs.Cells(BarisLanjut, 6).Value = Range("D18").Value
    'Input hasil seleksi ke dalam kolom G
    wsDtbs.Cells(BarisLanjut, 7).FormulaR1C1 = _
    "=SELEKSI(RC[-5],RC[-4],RC[-3],RC[-2],RC[-1])"

    'Panggil Sub Procedure cmdLast_Click (tombol Last)
    Call cmdLast_Click

End Sub
```

Building applications doesn't have to always be about writing programming codes. Many present-day applications have a ready-made model (prototype) which is one of the choices in modeling information systems. Information specialists can directly model the system needed by the user with prototyping tools which are then directly applied by the user. Template features, object-based programming, prototyping and various features in programming languages such as wizards, builders or generators minimize programmer work in writing programming codes.

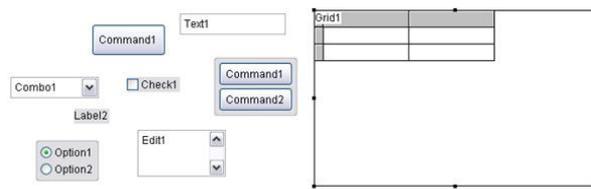**Figure 9.** *Visual objects in Graphical User Interface (GUI) based programming*

Figure 10. *Listing program automatically built*

```
lcLastWord = GETWORDNUM(THIS.cLine,THIS.nWords)
IF !ALLTRIM(THIS.oFoxCode.UserTyped) == ALLTRIM(lcLastWord)
    lcLastWord = THIS.oFoxCode.UserTyped
ENDIF

* Locate Command
SELECT tip, data, cmd, case FROM (_FoxCode) ;
  WHERE UPPER(ALLTRIM(expanded)) == UPPER(ALLTRIM(m.cCmd)) AND TYPE=UPPER(lcType) ;
  INTO ARRAY aTmpItems

* This section is for lines with multiple words such as SET TEXTMERGE or BROWSE NOMODIFY.
* We need to parse word by word to try and locate actual command.
IF _TALLY=0
    lnLastRec = 0
    lcWord = ""
    FOR i = 1 TO THIS.nWords
        lcWord = lcWord + UPPER(ALLTRIM(GETWORDNUM(cCmd,m.i)))
        LOCATE FOR UPPER(ALLTRIM(expanded))==lcWord AND TYPE=UPPER(lcType)
        IF FOUND()
            lnLastRec = RECNO()
            lcWord = lcWord + " "
        ELSE

            * Check for multi words such as ZOOM WINDOW -- 2nd pass only
            IF m.i=2 AND THIS.nWords=2
                LOCATE FOR ATC(lcword, expanded)#0 AND TYPE=UPPER(lcType)
                IF FOUND()
                    lnLastRec = RECNO()
                    lcWord = lcWord + " "
                ENDIF
            ENDIF
            IF lnLastRec=0 OR (m.i=2 AND THIS.nWords>2)
                lcWord = lcWord + " "
                LOOP
```

The database can also be awakened by its prototype (made instantly), without having to define one by one the attributes and table structure. Generally the database is used as a pilot and will be redesigned or database prototypes converted into a large scale database environment if it is to be used as a database in information systems. For example, the database blueprint is done using Microsoft Access and then converted (upsize) to the Oracle database or MYSQL database.

The existence of the wizard features, templates and generators are the basis of the prototyping tool before the actual application is created. The report builder feature, form builder, project generator or wizard facilities for programming tools show that the programming tool has the capacity to make prototyping. Even in many cases, the results of prototyping can be directly used as ready-to-use applications after going through several reviews of their functions and feasibility.

**Figure 1.4** *Building a database with Database Builder*
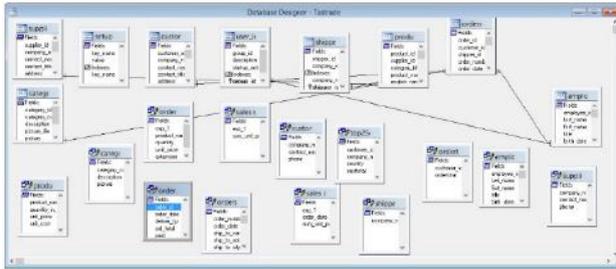
Note: The various facilities of the wizard are elements of the prototyping tool that enables programmers to make prototypes quickly. In some cases, prototypes can be used by users, and sometimes prototypes are used as models to build more complex information systems.

Some prototyping features commonly found in programming tools include: Application Wizard, Cross-Tab Wizard, Database Wizard, Documenting Wizard, Form Wizard, Graph Wizard, Import Wizard, Label Wizard, Local View Wizard, Mail Merge Wizard, One-To - Many Form Wizard, One-To-Many Report Wizard.

In the context of prototyping tools in designing information system architecture interfaces, the Balsamiq Mockup application is intended to create application interface models such as the homepage interface or information system. This makes it easier for the system analyst or programmer to communicate to users about the form of the system that will be built.

**Figure 11.** Balsamiq Mockup used in designing the user interface
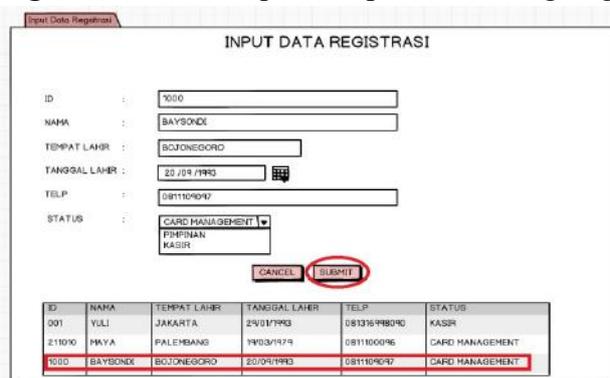
Application-based Project Management

Usually good application development uses project management knowledge in 'locking' the information system creation time is complete at the scheduled time. To meet this need an application with a Gantt Chart scheduling approach that maps time-based actions and actions can be used. Many tools can be used in software project management, even Excel can be used to create Gantt Charts which will be aimed at scheduling information system development.

Figure 1.5 Manage scheduling with the Gantt Chart method in Excel
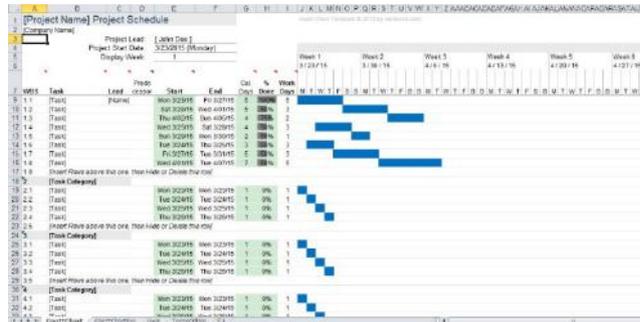
To use a special Gantt Chart application, a free application such as GanttProject 2.7 Ostrava can be sourced. The above are examples of the many tools used in designing and modelling information systems. This makes it easy for developers so that the software engineering process can be planned to meet the company's needs for information systems (Turban, 1995; Saudi et al., 2019).

Conclusion

There are various information systems today and various tools in creating information system architectures that will generate applications, data (databases), information technology-based interfaces and services. Even tools with analytical capabilities that exist today enable developers to experiment before creating an Executive Information System or Decision Support System. For reasons of efficiency and effectiveness in making information systems, the latest techniques are needed to act as catalysts for the various methods of developing information systems. In this way, information system projects take place without obstacles and synergize for each process, enabling coordination between professionals involved in information system design projects. Many extra benefits are potential when designing modern architecture, one of which is the automatic transformation of the information system model into a semi-finished or complete application.

Acknowledgment

REFERENCES

Ahmed, U., Zin, M. L. M. & Majid, A. H. A. (2016). Impact of Intention and Technology Awareness on Transport Industry's E-service: Evidence from an Emerging Economy. 산경연구논집 *(IJIDB)*, *7*(3): 13-18.

Berthold, H. (2003). Intelligent Data Analysis: An Introduction. Second Edition.

Feri, S. & Fajri, R.U. (2012). Excel Intelligent – Cara Pintar Membangun Strategi Bisnis. Elexmedia Komputindo Publisher.

Feri, S. (2010). Data Mining. Elexmedia Komputindo Publisher.

Feri, S. (2011). OLAP Excel – Cara Hebat Excel Mengelola Data. Elexmedia Komputindo Publisher.

Feri, S. (2017). Arsitektur Sistem Informasi. Penerbit Andi.

Feri, S.a & Fajri, R.U. (2015). Teknik Hebat Merancang Aplikasi Instan Berkualitas. Elexmedia Komputindo Publisher.

James, E. P. (n.d). Comparison of Software Development Lifecycle Methodologies. SANS Institute.

Jermsittiparsert, K., Siam, M., Issa, M., Ahmed, U. & Pahi, M. (2019). Do consumers expect companies to be socially responsible? The impact of corporate social responsibility on buying behavior. Uncertain Supply Chain Management, 7(4): 741-752.

Kalloniatis, C. (2012). Innovative Information Systems Modelling Techniques. Intech Open.

Kroenke, D.M. (1992). Management Information System. Watsonville: Mitchell McGraw Hill.

Marcus, A. (1992). Graphic Design for Electronic Documents and User Interface. Addison Wesley.

McLeod, R (1989). Management Information System:, Prentice Hall.

Pokorny, J., Repa, V., Richta, K., Wojtkowski, W., Linger, H., Barry, C. & Lang, M. (2011) . Information Systems Development - Business Systems and Services: Modeling and Development. 2011. Springer.

Pressman, R. S. (1997). Software Engineering : A practitioner's approach 4th Edition, Mc Graw Hill.

Radhika, D. A. (2013). Comparision of different SDLC models. International Journal of Computer Applications & Information Technology, II(I), January 2013 (ISSN: 2278-7720)

Rumbaugh, J., Ivar, J. & Grady, B. (2005). The Unified Modelling Language Reference Manual Second Edition, Addison-Wisley.

Saudi, M.H.M., Sinaga, O., Roespinoedji, D. & Jabarullah, N.H. (2019). Industrial, Commercial and Agricultural Energy Consumption and Economic Growth Leading to Environmental Degradation, Ekoloji, 28 (107), 299 – 310.

Sommerville, I. (2011). Software Engineering. Addison-Wesley.

Turban, E. (1995). DecisionSupport and Expert Systems: management SupportSystem, Fourth Edition, Prentice-Hall, Inc., United States of America.

Zhang, H. (2017). The Effect of Exchange Rate on Sino-EU and Sino-US Trade. Asian Journal of Economic Modelling, 5(4): 380-395.