System Analysis and Design using Unified Modeling (UML) and Data Flow Diagram (DFD)

Authors:
Rosziati Ibrahim, Norfaradilla Wahid, Norhanifah Murli, Hanayanti Hafit

email:
rosziati@uthm.edu.my
faradila@uthm.edu.my
hanifah@uthm.edu.my
hana@uthm.edu.my

Abstract:

This book introduces the analysis and design phase using object-oriented approach and structured-based approach. The use of examples with object-oriented approach and structured-based approach simplifies and reinforces the learning process. Each chapter presents examples to describe the concepts and elements of object-oriented approach and structured-based approach. For object-oriented approach, Unified Modeling Language (UML) specification diagrams, such as use case diagram, class diagram, sequence diagram and activity diagram, are used in case studies. For structured-based approach, Data Flow Diagram (DFD) specification diagrams, such as context diagram, data flow diagram, entity relationship diagram and flowchart, are used in case studies. The case studies were designed to support the essential topics discussed in the respective chapters.

This book consists of 6 chapters. Chapter 1 attempts to introduce the system development life cycle (SDLC) and its phases. Chapter 2 introduces rules for object-oriented approach and Chapter 3 covers the case studies using UML. Chapter 4 discusses rules for structured-based approach and Chapter 5 presents the case studies using DFD. Finally, Chapter 6 discusses the similarities and differences between UML and DFD. Each chapter includes

- Learning outcomes to alert readers with the content.
- Examples to demonstrate how the concepts of object-oriented approach and structured-based approach are implemented for the diagrams.

Keywords: Element similarities, Java, Level-0, Structured-based Approach, Unified Modeling Language

SYSTEM ANALYSIS AND DESIGN using

Unified Modeling Language (UML) and Data Flow Diagram (DFD)

Rosziati Ibrahim
Norfaradilla Wahid
Norhanifah Murli

SYSTEM ANALYSIS AND DESIGN using

Unified Modeling Language (UML) and Data Flow Diagram (DFD)

Rosziati Ibrahim
Norfaradilla Wahid
Norhanifah Murli

Author: Rosziati Ibrahim, Norfaradilla Wahid, Norhanifah Murli & Hanayanti Hafit

Published by:
Penerbit UTHM
Universiti Tun Hussein Onn Malaysia
86400 Parit Raja,
Batu Pahat, Johor
Tel: 07-453 8529/ 8698

Printed by:
Attin Press Sdn Bhd
8, Jalan Perindustrian PP4,
Tmn Perindustrian Putra Permai,
Bandar Putra Permai,
43300 Seri Kembangan, Selangor

Website: http://penerbit.uthm.edu.my
E-mail: pemasaran.uthm@gmail.com
http://e-bookstore.uthm.edu.my

Penerbit UTHM is a member of
Majlis Penerbitan Ilmiah Malaysia
(MAPIM)

Cataloguing-in-Publication Data

National Library of Malaysia

A catalogue record for this book is available from the National Library of Malaysia

ISBN 978-629-490-287-9

Contents

Preface

This book introduces the analysis and design phase using object-oriented approach and structured-based approach. The use of examples with object-oriented approach and structured-based approach simplifies and reinforces the learning process. Each chapter presents examples to describe the concepts and elements of object-oriented approach and structured-based approach. For object-oriented approach, Unified Modeling Language (UML) specification diagrams, such as use case diagram, class diagram, sequence diagram and activity diagram, are used in case studies. For structured-based approach, Data Flow Diagram (DFD) specification diagrams, such as context diagram, data flow diagram, entity relationship diagram and flowchart, are used in case studies. The case studies were designed to support the essential topics discussed in the respective chapters.

The Intended Audience

This book aims to deliver essential elements and concepts in drawing diagrams using object-oriented approach and structured-based approach to undergraduate students in computer science course and its related disciplines. The objectives of the book are to:

- introduce fundamentals of object-oriented approach and structured-based approach in problem solving.

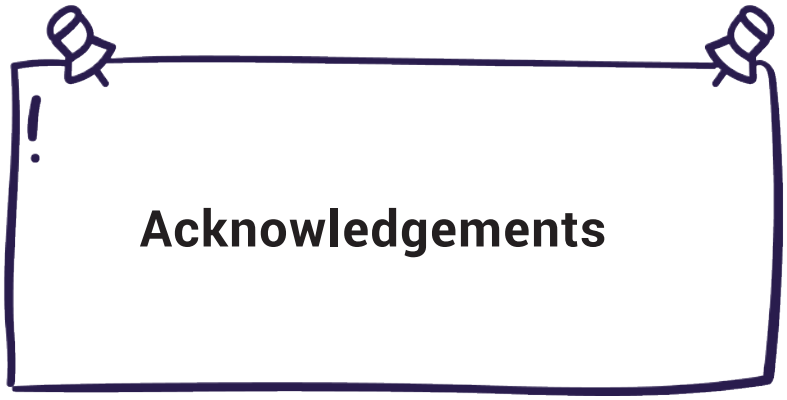
- demonstrate how diagrams can be developed based on user requirements using object-oriented approach and structured-based approach.
- implement case studies using UML and DFD.

The Content

This book consists of 6 chapters. Chapter 1 attempts to introduce the system development life cycle (SDLC) and its phases. Chapter 2 introduces rules for object-oriented approach and Chapter 3 covers the case studies using UML. Chapter 4 discusses rules for structured-based approach and Chapter 5 presents the case studies using DFD. Finally, Chapter 6 discusses the similarities and differences between UML and DFD.

Each chapter includes

- Learning outcomes to alert readers with the content.
- Examples to demonstrate how the concepts of object-oriented approach and structured-based approach are implemented for the diagrams.



Acknowledgements

We would like to convey our special thanks to individuals who have read the initial draft of this book and subsequently given invaluable comments.

We would also like to thank all our colleagues from the Faculty of Computer Science and Information Technology, UTHM who have helped and encouraged us during the course of producing this book.

Rosziati Ibrahim, Norfaradilla Wahid, Norhanifah Murli and Hanayanti Hafit

Parit Raja, Johor, Malaysia

CHAPTER 01

System Analysis and Design

Learning Outcomes:

After you have read this chapter, you should be able to

- understand the meaning of object
- know the phases inside SDLC
- know the concepts of object-oriented approach
- know the advantages of object-oriented approach

1.1 Introduction

The development of a system is usually based on the phases in the system development life cycle (SDLC). There are five main phases in the SDLC, namely the planning phase, the analysis phase, the design phase, the development phase and the maintenance phase (Hoffer et al., 2014). In the planning phase, the user requirements for a system will be collected first. These are known as user's requirements. Collecting information about user requirements for a system is very important because it will determine the functions that will be built for the system, which are known as the system's functionalities. For example, in the development of a system for the planning phase, two users have been identified for the use of a system. Therefore, from these two users, the system functions can be identified.

The system analysis and design phase is a unique process. In this phase, the requirements of a system that have been collected from the planning phase are illustrated through techniques that are easy for software developments to understand to build a system. Two approaches are often used in this system analysis and design phase. These approaches are known as object-oriented approach and structured-based approach.

1.2 Information System Development

In the development of information systems, a system will be built based on the information collected about the system. Typically, the system development life cycle (SDLC) will be used to build a system. The five main phases in this SDLC are the planning phase, the analysis phase, the design phase, the development phase and the maintenance phase which are shown in Figure 1.1.

1.6 Exercises

1. Give the definitions for the following terms:
   a. Object
   b. SDLC
   c. OOSDLC

2. Discuss the phases for SDLC.

3. Discuss the concepts of object orientation and the phases for OOSDLC.

4. Discuss the advantages of object-oriented approach.

CHAPTER 02

Unified Modeling Language (UML)

Learning Outcomes:

After you have read this chapter, you should be able to

- explain the basic concepts and purpose of UML.
- identify different UML diagrams and choose the right one for a system specification.
- apply UML rules to create clear and correct diagrams.

2.1 Introduction

In system development, the System Development Life Cycle (SDLC) can be applied using an object-oriented approach, referred to as the Object-Oriented System Development Life Cycle (OOSDLC). The OOSDLC consists of five main phases: the requirement analysis phase, object-oriented analysis phase, object-oriented design phase, object-oriented implementation phase, and object-oriented evaluation phase.

Within the object-oriented analysis phase, one of the most widely used approaches is the Unified Modeling Language (UML) specification. During this phase, both user and system requirements are gathered to ensure the system meets user needs. Once the requirement gathering process is completed, UML specifications are developed based on objects.

For the system analysis and design, UML is applied through various diagrams that illustrate different perspectives of the system. These diagrams commonly include the use case diagram, class diagram, sequence diagram, and activity diagram. In the next subtopic, a more detailed explanation of these diagrams and their roles in object-oriented analysis and design is presented. These diagrams form the foundation of UML modeling, which is essential in the development of ICT applications as it provides a clear and standardized way to visualize system requirements, structure, and behavior. By using UML diagrams, developers and stakeholders can better understand complex systems, reduce ambiguity, and ensure a well-organized system design before implementation.

In applications involving artificial intelligence (AI), UML modeling is especially important due to the complexity of data flows, intelligent components, and system interactions. UML helps document and manage these elements effectively, supporting system scalability, maintainability, and long-term evolution throughout the application lifecycle.

A variety of both commercial and open-source software tools are available for the development of UML models, such as Lucidchart, Visual Paradigm, StarUML, and Creately. However, for the scope and consistency of this book, draw.io is selected as the UML modelling tool.

2.5 Exercises

1. What is Unified Modelling Language (UML)? How does it support the object approach to system development?

2. Give some of the specification rules when developing them:
   a. Class diagram
   b. Use case diagram
   c. Sequence diagram
   d. Activity diagram

3. Consider a use case named "Check Credit Eligibility". Explain how it can function as a use case in include and extend relationships within a use case diagram for a Loan Application System.

4. A system is to be developed for booking a room for Raya Hotel. Gather the user requirements for the system and develop the following diagram:
   a. Use Case diagram
   b. Class diagram
   c. Sequence diagram
   d. Activity diagram

CHAPTER 03

Case Studies using UML

Learning Outcomes:

After you have read this chapter, you should be able to

- apply UML modeling techniques to case studies using use case, class, sequence, and activity diagrams.
- demonstrate how system requirements can be represented in UML diagrams that show structure, behavior, and interactions.
- evaluate multiple UML diagrams to ensure consistency, accuracy, and completeness in modeling case study systems.

3.1 Introduction

This chapter will discuss and solve two case studies using UML specifications in the system analysis and design phase. Four important diagrams for UML specifications need to be produced in each case study to develop a system in the analysis and design phase.

3.2 UML - Case Study 1

This section will examine a case study in general. For example, a food ordering system can be implemented online to make it easier for users to order food and have it delivered to them. The summary of the case study for this online food ordering system is further elaborated in Case Study 1 as follows:

Case Study 1

> Develop a system for food ordering and delivery. This system allows users to order food online. The food ordered by the users will be delivered by a delivery person, and a delivery fee will be charged based on the food ordered.

From the issues mentioned in Case Study 1, referring to Figure 1.2, the requirements analysis phase will be conducted first. In this phase, the system requirements are gathered. From Case Study 1, it can be explained that this system has three types of users: regular users, administrators, and delivery personnel. Regular users can log in to the system after completing the registration process. Once successfully registered, users can place food orders by selecting items from the menu through this food ordering system. Meanwhile, administrators can log in to the system and manage order and menu information. The administrator will also send information to the delivery personnel so that the food can be delivered to the system's users. The users and their requirements for this food ordering system are shown in Table 3.1.

3.7 Exercises

A clinic management system is designed to help clinics manage its daily operations. The system allows staff and patients to perform general clinic-related tasks. Staff can register new patients, manage appointments, send reminders, and handle payments. Patients can register, book appointments, receive reminders, and attend consultations. Doctors can view patient records, conduct consultations, and update medical information. The system also sends automatic appointment reminders and generates reports for clinic management.

1. For the clinic management system, identify all requirements for each of the users.
2. From the four main UML diagrams discussed in this chapter, identify which diagrams represent structural and which represent behavioral aspects of the system. Then, develop all the corresponding diagrams.

CHAPTER 04

Data Flow Diagram (DFD)

Learning Outcomes:

After you have read this chapter, you should be able to

- explain the purpose and role of DFDs
- differentiate between levels of DFDs (context diagram, Level-0, Level-1, etc.) and explain the relationships between them.
- identify external entities, processes, data flows, and data stores
- apply DFD rules and conventions
- construct context diagrams and Level-0 and Level-1 DFDs

4.1 Introduction

Structured system analysis and design is based on the phases of the system development life cycle (SDLC) phases. This SDLC was discussed in Chapter 1. The same phases will be used for this structure-based analysis and design. Figure 1.1 from Chapter 1 discussed five important phases in SDLC, which include the planning phase, analysis phase, design phase, development phase and maintenance phase.

In structured-based approach, user requirements are gathered and referred as user requirements, as discussed in Chapter 1. Based on these user requirements, the structured-based analysis and design phases will be described through specific diagrams. These diagrams, namely the context diagram, data flow diagram, entity relationship diagram, and flowchart, are constructed using draw.io This chapter will further explain the structured-based approach in more detail.

4.2 Structured-Based Approach

Structured-based approach can also be applied in the development of a system. Figure 4.1 illustrates the SDLC that can be used for the structure-based method, as discussed in Chapter 1. For the discussion on the structure-based method, the phases of this SDLC are revisited in Chapter 4.

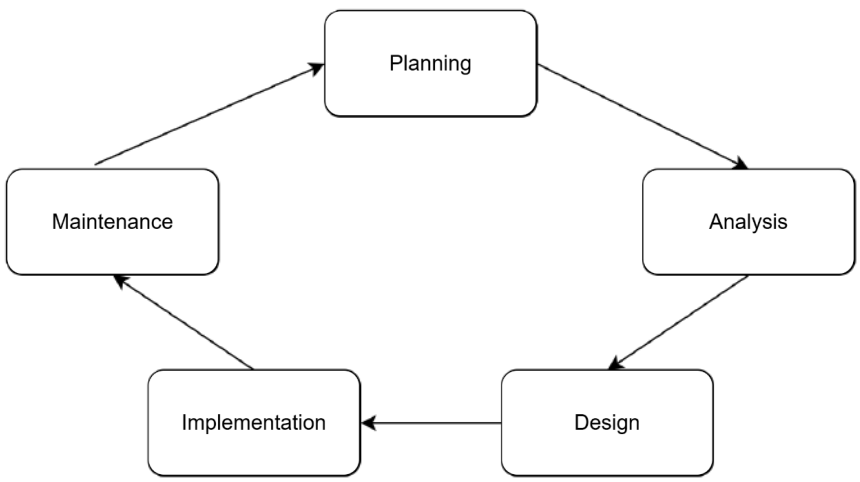


Figure 4.1 System development life cycle [Hoffer *et al.*, 2014]

4.8 Exercises

1. Describe the advantages of structured-based approach.

2. Give some of the specification rules when developing the:

   a. Context diagram
   b. Data flow diagram
   c. Entity relationship diagram
   d. Flowchart

3. A system is to be developed for booking a room for Raya Hotel. Gather the user requirements for the system and develop the following diagram:

   a. Context diagram
   b. Data flow diagram
   c. Entity relationship diagram
   d. Flowchart

CHAPTER 05

Case Studies using (DFD)

Learning Outcomes:

After you have read this chapter, you should be able to

- identify four essential diagrams required in analysis and design phase of system development: context diagram, DFD, ERD and flowchart
- produce context diagram that graphically represent a system
- produce level 0 DFD from context diagram
- decompose processes from level 0 DFD into level 1 DFD
- define external entities and data stores
- determine the relationship between external entities and data stores
- construct ERD
- generate detailed flowchart for each entity interacting in the system

5.1 Introduction

This chapter explains how two case studies can be solved using Data Flow Diagrams (DFDs). Four essential diagrams in DFD modeling must be produced for a given case study in order to construct a system during the analysis and design phase. These four diagrams include the context diagram, the data flow diagram, the entity-relationship diagram, and the flowchart.

5.2 DFD - Case Study 1

This section describes a commonly used case study. For consistency, the case study discussed in Chapter 3 will be reused for the case study involving DFDs. To facilitate the discussion, the case study from Chapter 3 is rewritten in this chapter as follows.

Case Study 1

> Develop a system for food ordering and delivery. This system allows users to place food orders online. The food ordered by the user will be delivered by a delivery personnel, with a delivery charge applied in accordance with the food ordered.

From the problem statement of Case Study 1, as presented in Chapter 3, the system requirements are first gathered. Table 5.1 presents the system requirements, which are the same as those in Table 3.1.

Table 5.1 System requirements for the *Food Ordering* and *Delivery System*

User	User Requirement
User	Register
	Login
	Order Food
Delivery Personnel	Food Courier
Admin	Login
	Update Information
	Order Delivery

5.6 Exercises

1. Design a system for online library book borrowing and reservation management. This system allows users (such as students or library members) to search, reserve, and view the status of book loans online. The Librarian (administrator), meanwhile, is responsible for updating the book catalogue information and processing loan and reservation requests made by users. Your design should include context diagram, level 0 and level 1 DFD, entity relationship diagram and flowchart.

2. A developer wants you to develop a Staff Management System (SMS). The system has details for each staff, such as his/her name, ID number, salary, marital status, and job title. These details are recorded in a staff database. The SMS can allow a staff to login into the system and view his/her details. The SMS is also able to allow a staff (which is an administrator) to login into the system and view, update and delete the record of the staff within the database. Your design should include context diagram, level 0 and level 1 DFD, entity relationship diagram and flowchart.

CHAPTER 06

UML and DFD

Learning Outcomes:

After you have read this chapter, you should be able to

- know the diagrams to be used in UML
- know the diagrams to be used in DFD
- identify the differences between UML and DFD
- identify the similarities between UML and DFD

6.1 Introduction

In the system analysis and design phase, two approaches can be used to produce the specification of a system. For UML specifications, the object-oriented approach is used, while for DFD specifications, the structured-based approach is used. For the object-oriented approach, four important diagrams are used for UML specifications, namely use case diagram, sequence diagram, class diagram and activity diagram. Meanwhile, for the structured-based approach, four diagrams are also used for DFD specifications, namely level zero data flow diagram, level one data flow diagram, entity relationship diagram and flowchart.

6.2 Differences and Similarities between UML and DFD

In the analysis and design phase of object-oriented approach, UML specifications are used. Whereas, in the analysis and design phase of structured-based approach, DFD specifications are used. The diagrammatic similarities used between UML specifications and DFD specifications are summarized as in Table 6.1.

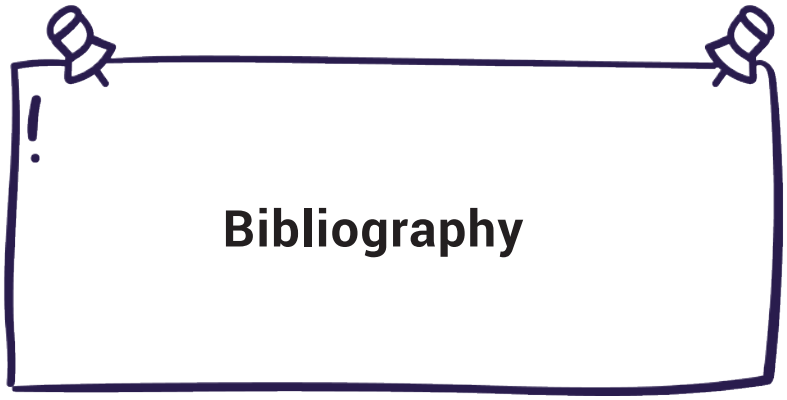
Table 6.1 The diagrammatic similarities used between UML and DFD.

UML	DFD
Use case diagram	Level zero data flow diagram
Sequence diagram	Level one data flow diagram
Class diagram	Entity relationship diagram
Activity diagram	Flowchart

From Table 6.1, for both approaches, the system requirements are illustrated through use case diagram (object-oriented approach) and level-0 data flow diagram (structured-based approach). The similarities between these two diagrams are in the system users and user requirements. Figure 3.1 in Chapter 3 shows an example of a use case diagram and Figure 5.2 in Chapter 5 shows an example of level-0 data flow diagram. Both diagrams have similarities in the system users (User, Administrator and Order Sender) and system requirements (Register, Login, Order Food, Update Information and Order Delivery).

6.4 Exercises

1. State diagrams that are always used in UML.
2. State diagrams that are always used in DFD.
3. Discuss the differences between UML and DFD.
4. Discuss the similarities between UML and DFD.



Bibliography

Bahrami A. (1999) *Object-Oriented Systems Development*. Singapore: McGraw-Hill.

Cornford T. and Smithson S. (2006*). Project Research in Information Systems – A Student's Guide.* 2nd Edition, Palgrave Macmillan.

Creately. (2026). https://creately.com (Search Date: 14/01/2026).

Dawson C.W. (2009), *Projects in Computing and Information Systems – A Student's Guide,* 2nd Edition, Addison Wesley.

Deitel, P and Deitel, H. (2017). Java: How to Program. 11th ed. Pearson.

Dennis, A., Wixom, B.H. and Roth, R.M. (2006). *Systems Analysis and Design*. 3rd ed. Hoboken: John iley & Sons, Inc.

Draw.io. (2025). https://www.draw.io (Search Date: 14/01/2025)

Hoffer J., George J. and Valacich J. (2014). *Modern Systems Analysis and Design*. Pearson International Edition.

Ibrahim R. (2000) Formal Methods applied to Component-Based Systems. (Ph.D. Thesis, QueenslandUniversity of Technology (QUT), Brisbane, Australia, March 2000.)

Ibrahim R. and Jamel S. (2006) *Object-Oriented Programming Using C++ - AN INTRODUCTION*. Malayisa: McGraw-Hill.

Lewis, J. (2015). Java Software Solutions: Foundations of Program Design. Pearson. Call number: QA76.73.J38 .L48 2015.

Liang, Y. D., (2019). Introduction to Java Programming. Pearson.

Lucidchart. (2026). https://www.lucidchart.com (Search Date: 14/01/2026).

Micallef J. (1988) Encapsulation, Reusability and Extensibility in Object-Oriented Programming Languages. *Journal of Object-Oriented Programming*. Vol. 1, No. 1, April/May 1988.

Microsoft. (1996) *The COM Specification*. Draft, Microsoft Corporation.

Object Management Group. (2017). Unified Modeling Language (UML), Version 2.5.1.

Object Management Group. (2025) https://www.omg.org/spec/UML/2.5.1/ (Search Date: 09/09/2025)

Rosziati Ibrahim, Noor Azah Samsudin, Norhalina Senan, Muhammad Amir (2021) *Object-Oriented Programming and Data Analysis using Java*, Penerbit, UTHM, ISBN 978-967-2975-16-8.

Rosziati Ibrahim. (2020) *Analisis dan Reka Bentuk Sistem Menggunakan Bahasa Pemodelan Bersepadu (UML) dan Gambar Rajah Aliran Data (DFD),* Dewan Bahasa dan Pustaka, ISBN 978-983-49-2557-4.

Rosziati Ibrahim. (2014) *Project Development Methodology for Computer Science Projects using an Object-Oriented Approach.* UTHM Publication, ISBN 978-967-0468-55-6.

Rosziati Ibrahim. (2008) *An Introduction to Object-Oriented Programming with UML Using Borland C++*. Malaysia: UTHM Publication, Batu Pahat.

Rosziati Ibrahim and Noraini Ibrahim. (2009). A Tool for Checking Conformance of UML Specification. *Proceedings of the 2009 World Academic of Science and Technology (WASET),* Volume 51, pp. 262-266.

Satzinger J.W. and Orvik T.U. (2001) *The Object-Oriented Approach: Concepts, System Development, and Modeling with UML.* 2nd Edition. Thomson Learning.

Schildt, H. (2017). Java: A Beginner's Guide. 7th Edition. New York: McGraw-Hill Education.

Schildt, H. (2018). Java: The Complete References. 11th Edition. New York: McGraw-Hill Education.

Sommerville I. (2007). *Software Engineering*. 8th Edition, Addison Wesley, England.

Szyperski C. (1998) *Component Software: Beyond Object-Oriented Programming.* ACM Press, Addison Wesley.

UML 2.0 Diagrams (2020). UML 2.0 Diagrams, http://www.omg.org/spec/UML/2.0/ (Search Date 1/4/2020)

Biography of Authors

Rosziati Ibrahim is with the Software Engineering Department, Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia (UTHM). She obtained her PhD in Software Specification from the Queensland University of Technology (QUT), Brisbane and her MSc and BSc (Hons) in Computer Science and Mathematics from the University of Adelaide, Australia. Her research area is in Software Engineering that covers Software Specification, Software Testing, Operational Semantics, Formal Methods, Data Mining, Image Processing and Object-Oriented Technology.

Norfaradilla Wahid is a Senior Lecturer at Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia (UTHM). She received her Ph.D. in Computer Science from La Trobe University Australia, and Masters of Computer Science and Bachelor of Computer Science from Universiti Teknologi Malaysia. She has more than 10 years teaching experience in various computing courses and receives a number of research grants from UTHM and Ministry of Education. She is also certified with the International Requirements Engineering Board (IREB) Certified Professional in Requirements Engineering (CPRE) Foundation Level by of the Malaysian Software Testing Board. Her research interest is Database Modelling, Web Ontology, Internet of Things and Blockchain.

Norhanifah Murli is a Senior Lecturer at Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia (UTHM). She received her Ph.D. in Computer Science from Auckland University of Technology, New Zealand, Master of Information Technology from Universiti Teknologi MARA and Bachelor of Information Technology from Multimedia University. Her research area includes Multimedia Content Development, Human-Computer Interaction and Data Mining.

Hanayanti Hafit is a Senior Lecturer at Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia (UTHM). She received her Master Degree in Information Technology from Universiti Putra Malaysia and Bachelor of Information Technology from Universiti Utara Malaysia. Her research area includes Human Computer Interaction and Web Technology.

Index

S

T

U

SYSTEM ANALYSIS AND DESIGN using

Unified Modeling Language (UML) and Data Flow Diagram (DFD)

This book introduces the analysis and design phase using object-oriented approach and structured-based approach. The use of examples with object-oriented approach and structured-based approach simplifies and reinforces the learning process. Each chapter presents examples to describe the concepts and elements of object-oriented approach and structured-based approach. For object-oriented approach, Unified Modeling Language (UML) specification diagrams, such as use case diagram, class diagram, sequence diagram and activity diagram, are used in case studies. For structured-based approach, Data Flow Diagram (DFD) specification diagrams, such as context diagram, data flow diagram, entity relationship diagram and flowchart, are used in case studies. The case studies were designed to support the essential topics discussed in the respective chapters.

This book consists of 6 chapters. Chapter 1 attempts to introduce the system development life cycle (SDLC) and its phases. Chapter 2 introduces rules for object-oriented approach and Chapter 3 covers the case studies using UML. Chapter 4 discusses rules for structured-based approach and Chapter 5 presents the case studies using DFD. Finally, Chapter 6 discusses the similarities and differences between UML and DFD. Each chapter includes

- ✔ Learning outcomes to alert readers with the content.
- ✔ Examples to demonstrate how the concepts of object-oriented approach and structured-based approach are implemented for the diagrams.

UTHM
Universiti Tun Hussein Onn Malaysia
Penerbit

ISBN 978-629-490-287-9
03300
9 786294 902879

PENERBIT UTHM