Title: A dynamic platform for workflow management system: a ward management perspective

Author: Nantika Prinyapol

Year: 2010

Repository DOI:

2010

# A dynamic platform for workflow management system: a ward management perspective

Nantika Prinyapol
*University of Wollongong*

# A Dynamic Platform For Workflow Management System: A Ward Management Perspective

A thesis submitted in (partial) fulfilment of the
requirements for the award of the degree

**Doctor of Philosophy**

from

**UNIVERSITY OF WOLLONGONG**

by

**Nantika Prinyapol**

B.Sc. Information Technology, Assumption University
M.Sc. Information Management, Asian Institute of Technology

**School of Information System and Technology**

**Faculty of Informatics**

**2010**

## THESIS CERTIFICATION

CERTIFICATION

I, Nantika Prinyapol, declare that this thesis, submitted in partial fulfilment of the requirement for the award of Doctor of Philosophy, in the School of Information Systems and Technology, University of Wollongong, is wholly my own work unless otherwise referenced or acknowledged. The document has not been submitted for qualifications at any other academic institution.

Nantika Prinyapol

31 March 2010

# LIST OF PUBLICATIONS

This is a list of referred conference papers that are related to this research work.

Prinyapol, N, Lau, SK & Fan, J 2010, 'A Dynamic Nursing Workflow Management System: A Thailand Hospital Scenario', in *Intelligent Automation and Computer Engineering*, Lecture Notes in Electrical Engineering, Springer, vol.52, pp489-501.

Prinyapol, N, Fan, J & Lau, SK 2009, 'A Hospital Based Dynamic Platform Workflow Management', *IAENG International Journal of Computer Science*, vol.36, no.2, pp192-198.

Prinyapol, N, Fan, J & Lau, SK 2009, 'A Dynamic Platform for Business Process Management (BPM) Using Service-Oriented Enterprise (SOE)', in *Proceedings of the 3rd International Conference on Internet Technologies and Applications (ITA09)*, Wrexham, North Wales, UK, 8-11 September 2009, pp245-252.

Prinyapol, N, Fan, J & Lau, SK 2009, 'A Dynamic Platform for Workflow Management Using Web Services: A Hospital Scenario', in *Proceedings of the International Conference on Internet Computing and Web Services (ICICWS'09)*, Hong Kong, 18-20 March 2009, pp944-949.

Prinyapol, N & Sun, Z 2006, 'Expectation and Perception of E-Customers and E-Providers for E-Service Recommendation', in *Proceedings of the 5th International Conference on E-Business (NCEB2006)*, Bangkok, Thailand, 2-3 November 2006, pp87-94.

## TABLE OF CONTENTS

## LIST OF TABLES

## LIST OF FIGURES

## LIST OF ABBREVIATIONS

| | |
|---|---|
| AJAX | Asynchronous JavaScript and XML |
| AfC | Agenda for Change |
| ANA | American Nurses Association |
| ANMC | Australian Nursing and Midwifery Council |
| ASP | Active Server Pages |
| BPD | Business Process Discovery |
| BPM | Business Process Management |
| BPMN | Business Process Modelling Notation |
| DPWFM | Dynamic Platform for Workflow Management |
| EN | Enrolled Nurse |
| FAS | Function Allocation Service |
| FS | Function Service |
| IPD | In-Patient Department |
| LAN | Local Area Network |
| LPN | Licensed Practical Nurse |
| NHS | National Health Service |
| NMC | Nursing and Midwifery Council (UK) |
| NMRA | Nursing and Midwifery Regulatory Authorities (AUS) |
| NP | Nurse Practitioner |
| NSW | New South Wales |
| OS | Operation System |
| PDA | Personal Digital Assistant |

| | |
|---|---|
| PHP | PHP: Hypertext Preprocessor |
| RN | Registered Nurse |
| RS | Recompilation Service |
| SOA | Service-Oriented Architecture |
| SOAP | Service-Oriented Architecture Protocol |
| TAFE | Technical And Further Education |
| UDDI | Universal Description, Discovery and Integration |
| UML | Unified Modelling Language |
| W3C | World Wide Web Consortium |
| WFM | Workflow Management |
| WPS | Work Profile Service |
| WSDL | Web Service Description Language |
| XML | eXtensible Markup Language |
| YAWL | Yet Another Workflow Language |

# ABSTRACT

This research proposes a dynamic recompilation platform for a workflow management system to manage a hospital ward. Literature review has shown that ward management in hospitals and nursing care processes are complicated and it is not easy to design and develop a ward management system that is easy to use and one that suits requirements of any ward due to the complex nature of the hospital environment. A workflow management system that can be customised and recompiled is desired due to the dynamic nature of the nursing care process. This research investigates the feasibility of using web service technology to develop a workflow management system that enables a nursing supervisor to customise their work requirements using a dynamic recompilation technique. The two main features of the proposed system are customisation and dynamic recompilation. Customisation allows users to modify functions within the web service repository to suit individual tasks based on their work profile and situations, whereas dynamic recompilation allows multiple web service repositories to be recompiled and arranged into a new set of dynamic functional services when task assignment changes. This research proposes a framework of ward workflow management system using web services technology. We called the proposed system the dynamic platform for workflow management system (DPWFM) consisting of four web service repositories that include work profile service (WPS), function service (FS), function allocation service (FAS) and recompilation service (RS). There are three perspectives to the DPWFM: organisational, functional and procedural. The organisational aspect of the WPS defines the organisational roles of individual nurses in the hospital, the functional aspect of the FS describes tasks, activities and services to be performed and the procedural aspect of the FAS describes the allocation and assignment of tasks. The recompilation aspect of the

DPWFM is the RS that dynamically recompiles the function services using inputs from the WPS, FS and FAS to create an agenda workflow in the form of scheduled tasks to help nurses in organising and performing the assigned tasks. We will present a scenario to show how the dynamic recompilation of the DPWFM can be applied in a ward. The architecture of the proposed system consisting of three architectural layers of presentation, business logic and data layers will also be presented. The contribution of this research is the development of an innovative approach of using web services technology to manage workflow in the hospital ward.

## ACKNOWLEDGEMENTS

Without the help of many people, I might never have finished this thesis. First of all, I would like to express my gratitude to my supervisor, Dr. Sim Kim Lau, who has supported and guided me with her degree of patience and professionalism that greatly exceeded the requirements of the supervisory role and thus ensured that I maintained the focus and commitment necessary to complete this task.

I am especially indebted to Dr. Joshua Fan for his tremendous helps. He constantly encouraged and motivated me in every stage of my research. In the past several years, I have learned so much from him. He allowed me to learn through the reflection on my experiences rather than telling me what to think. I am thankful for those couching as I have grown as a person as well as reaching my professional goals. I am very privileged to have the opportunity to work under his tutelage.

Special thanks go to Prof. Peter Eklund, my co-supervisor for his tremendous supports for my proposal presentation. I also would like to thank my former supervisor, Dr. Zhaohao Sun, for his guidance during the early stage of my study. Appreciation and my gratitude are to all members of the Faculty of Commerce, Sydney Business School and the Faculty of Informatics for the great supports.

The extremely difficult process of Ethics approval from University of Wollongong and Port Kembla Hospital (NSW, Australia) would not have been successful without the help of Mr. Rob Gordon, deputy director of the Centre for Health Service Development, UOW. Thanks are also due to Mrs. Katharine who helps in note taking of the interviews with Port Kembla

Hospital. I also wish to extend my thankfulness to all nurse unit managers in the Port Kembla Hospital and many head nurses of the hospitals in Thailand for participating in my interview and offering the significant information to support my research study.

In addition, I am greatly indebted to my university, Dhurakij Pundit University, Thailand, which has granted me the scholarship for postgraduate studies that has the supported my throughout my PhD Candidature.

Special appreciation is given to my parents and my brother for their love, patience, understanding and support.

Last, but not least, I also would like to show my appreciation to all of my friends (Yui-Ken, P’Mam, Noi, P’Om, Riam, P’Jeab, Tok, P’Keaw, P’Lek, P’Aree, Ying, Joey, Soi, Peng, P’Kim, Chon, P’Tee, P’Tung, P’Tu, Winnie, P’Boong and Marco) for their friendship and encouragement.

# CHAPTER 1: INTRODUCTION

This chapter gives a synopsis of the thesis that investigates application of web services to support workflow management (WFM). Background and statement of problem will be discussed too.

The chapter is organised as follows. Section 1 presents an introduction. Section 2 describes the statement of problems. Section 3 discusses an overview concept of the research framework. The three main components of the proposed framework have been briefly described. Research aim and research objectives are presented in section 4 and 5. Section 6 describes the research method to understand the research methodologies. The contribution and limitation of the research are raised in section 7 and 8. Section 9 gives the organisation of the thesis.

## 1. 1 Introduction

According to Australian Bureau of Statistics (2008), the number of business organisations in Australia using the Internet in business was 86.8% during the year ended 30 June 2008; 99% of big companies that employed two hundred or more workers use the Internet. Nearly 90% of healthcare and social services, medium and large hospitals, have started to pay attention to the use of the Internet to manage workflows. Among businesses that use the Internet in healthcare services, more than 90% use broadband as the main connection type (Australian Bureau of Statistics 2008). In addition, many organisations have replaced paper-based systems with web-based systems. In a study by Prinyapol et al. (2010), it is observed that medical staffs such as nurses concur that they can collaborate their work and transfer information between wards and other departments in the hospital easier and more accurately if it is done through computerised systems. However, workflows in the hospital are complex and complicated as well as inflexible, in particular workflows between wards. Our results show that currently, many older nurses are not skilful in using the computerised systems in their works. The process is complicated by some departments using different platforms for their computer systems. Therefore, developing a flexible workflow management system, that can be customised to suit individual wards, and for nursing care processes is the focus of this research.

In the last decade, workflow management technology and tools have been introduced in business organisations to improve efficiency of business processes. Hospitals have also used workflow management system to support their organisations (Shefter 2006). In fact, nursing care is considered one of the most difficult workflow management systems to develop due to the complexity of patient care process in the ward. The responsibilities of nurses include medical care, nursing work, patient documentation and treatment delivery (Hannah et al. 2006). In addition, nurses also need to assist patients in cleaning and bathing. The nurses also need to communicate with patients' families on patient's condition and treatment plan.

Medical record-keeping activity is an essential element in the patient care process (Just 2008). Generally, nurses have to record all information about patient care as part of the patient documentation procedure. For example, the in-charge nurse has to write down doctor's orders into nursing tools such as Kardex cards or nurse notes and then distribute these tasks to other nursing staff. The nurses then have to follow their assignments appropriate to their levels of job responsibilities. They also have to crosscheck and record the tasks once the tasks are completed. Although the processes look simple, the nurses constantly need to think and plan ahead for the next task ahead. Sometimes nurses focus more on medical prescriptions than on nursing care (Mbabazi and Cassimjee 2006). In addition, the tasks can get more complicated when there are many patients in the ward. Potential for making errors can be high and often results in nurses working in stressful environment (King et al. 2009).

Health informatics is crucial for hospital. Hospitals that have more accurate information systems than their competitors are also considered to be more efficient in ward performance (Bauer 2009). However, the existence of vast amounts of information related to a hospital may not necessarily help nurses to carry out their tasks efficiently. Instead, nurses frequently use irrelevant information for their routine tasks. For example, it may take a long time to search for the reports or information they need. Sometimes, it results in information overload and redundant information. As a result, this can affect nurses' performance and increase the possibility of making errors. Although computerised systems have been introduced to improve efficiency in work practice, paper-based patient documentation is still operating in a lot of hospitals (Middleton et al. 2009). A workflow management system that enables the workflow of nursing care is not easy to achieve and is also difficult to design to suit individual ward.

## 1. 2 Statement of Problems

Currently, ward management is based on manual systems to support health treatment and nursing care process and computerised systems are used mainly for transaction based processing such as billing, inventory control, patient database and patient profile. Nurses usually maintain medical records manually and record results of patient observations manually. In many situations, it is not unusual for nurses not able to provide adequate medical record-keeping in a timely manner particularly during busy periods (Kim and Lewandrowski 2009). One of the reasons that computerised systems of nurse notes have not been successfully deployed is nurses' unfamiliarity to use the computer and difficulty in typing nursing note when they are in the ward taking care of patients (Specht 2008). In addition, very often the nurses have to adjust or change their work practices to fit into the system that is deployed in the ward (Grimes and Jackson 2006). For example, results from Prinyapol et al. (2010) show that nurses in Thailand would prefer to use computerised system if the system can truly help them to improve efficiency of the nursing care processes. Currently, many wards still use paper-based process as a primary form of patient documentation. At the moment, ward management still has not fully been computerised in terms of ward and nursing care management.

Shefter (2006) shows that many hospitals use network systems to communicate between ward departments. Swanson and Dans (2000) have shown that several hospitals have used management tools to organise workflows, however these applications have not been able to cater and meet the actual requirements of workflow in the ward. In a lot of cases, it is very difficult to modify precise procedures or rules of an existing system and it is also expensive to maintain. As discussed previously, when hospitals deploy a new computerised system, very often they also need to change the ways in which nurses carry out their routine tasks (Bilignaut 1999). Intensive training courses are required to teach nurses to use the system correctly and to reduce anxiety of nurses (Specht 2008). The complexity of learning a new system is one of the factors that results in the unsuccessful deployment of the system (Murphy 2004). As a result, instead of reducing workloads and improving work performances using computerised systems, nurses find that they are loaded with additional work that is not part of their major responsibility. Consequently, the nurses reported an increase in stress levels which leads to unproductive work (Ghahramani et al. 2009, Prinyapol et al. 2010). Nurses reported they require simple and friendly user interface design to help them to handle constantly changing workloads and to improve the patient documentation process (Parker and Baldwin 2008, Matnieseir 2010, Tynan 2010). Individualised and customisable information needs and a simple to use system are essential features that the nurses are looking for in any computerised system.

Compilation services, dynamic access and customisation of information are some of the key issues that this research wishes to address. Several researchers have conducted research in web services and workflow management (Alonso et al. 2004, Chatterjee 2004, van der Aalst and van Hee 2004, W3C 2006), however practical application of web services recompilation has not been thoroughly investigated, in particular from a hospital ward organisation perspective. Hence, this research aims to find a suitable solution that enables nurses to improve their work performances using web services technology.

## 1. 3 Overview of Research Framework

A dynamic platform for a workflow management (DPWFM) that employs web service technologies to organise workflow in ward environment is developed in this research. The proposed concept has three perspectives: organisational, functional and procedural as shown in Figure 1.1. Organisational perspective describes the characteristics of the workers. We called this perspective the work profile service repository (WPS) in our framework. Functional perspective describes task-related requirements and we called it the function service repository (FS). Procedural perspective describes operational setting, business logics and business rules, which we called the function allocation service repository (FAS).

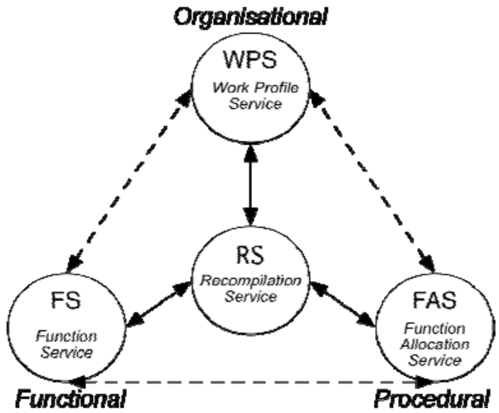



Figure 1.1 Dynamic platform for workflow management components

The web services are mostly stored as objects in the web service repositories. The WPS repository consists of worker profiles that include name, qualification, position and access authorises. With the WPS, the system can describe user roles and access levels in using the FS services. The FS repository consists of details of task functions and the relationships of functions and sub-functions such as predecessor ordering of function task. The FS also describes business logic within the organisation. The FAS allows supervisor to assign tasks based on nursing work requirements. Different nurses have different responsibilities, therefore the schedule arrangement of the FAS of each nurse requires the customisation by supervisor.

These three perspectives of WPS, FS and FAS web services generate results in a form of workflow using a recompilation process resulting in the agenda workflow as the outcomes of a recompilation services (RS). The agenda workflow becomes the scheduled function services to individual nurse that allows individual nurse to perform their tasks based the scheduled tasks and function allocation. User-friendliness interface, customisable function and dynamic recompilation are main features of the proposed system.

The DPWFM allows the supervisors to customise function tasks for their subordinates. Every time tasks are assigned or changed, dynamic recompilation services will be functionally re-compiled. The nurse can easily follow the new systematic workflow arrangement. This proposed system embodies the dynamic recompilation and customisation is illustrated as proof of concept to demonstrate the feasibility of the proposed framework.

## 1. 4 Research Aim

This research aims to investigate development of a dynamic platform for nursing workflow management using technologies of web services to enable customisation and recompilation capabilities in nursing workflow information system.

## 1. 5 Research Objectives

The objectives are as follows:

- To investigate current state of nursing workflow system.
- To investigate the applicability of web services technology to ward management system.
- To propose a web service framework for nursing workflow system.

## 1. 6 Research Methods

This research is conducted in the following stages:

A literature review has been conducted to explore current state of ward management. The nursing care workflow and functions have been investigated. In particular, the suitability of web service technologies and workflow management as the basic framework to enhance nursing workflow system is investigated.

There are two phrases of surveys that have been conducted in Thailand and Australia. The first phrase is an observation and interview. It has been conducted to identify the existing routine problems in the nursing healthcare sector and illustrate the current workflows during the handover time between working shift of head nurses and medical nurses in the Port Kembla Hospital, NSW, Australia and five hospitals, Bangkok, Thailand. The survey, in form of short interviewing and observation, collects the data from one head nurse and observing the workflows of nurse staffs in a ward. The second phrase is a demonstration of the DPWFM system to the nurses for evaluating the usability of the DPWFM platform in the ward hospitals. It has been conducted after implementing the prototype of the DPWFM application. The participants are six head nurses and three medical nurses in five Bangkok hospitals, Thailand.

A framework has been proposed to improve nursing workflow system in the ward using web services and web-based technologies. The proposed conceptual framework is called Dynamic Platform Workflow Management (DPWFM). An illustration scenario has been applied and a physical design framework has been proposed.

Finally, the prototype of the DPWFM system has been developed to demonstrate its effectiveness for improving the work performances. The prototype applies open source technologies to facilitate nurses' communication through the web applications.

## 1. 7 Research Contribution

In this research, we propose to develop an easy-to-use and flexible platform to improve nursing workflows and enhance efficiency in patient documentations in a ward management. The web services technologies have been used to support nursing workflow practice. The advantages of using web service development are to allow dynamic interoperation and be able to support multiple integrated systems. The main contribution of this research is the development of an innovative method of ward workflow management system that has the advantage of customisation and recompilation.

## 1. 8 Organisation of Thesis

The rest of thesis is organised as follows.

Chapter 2 of this research is literature review that helps to understand the cultures of nursing care processes in a ward hospital. The discussions of nursing roles, responsibilities and functionalities are described. The ward nursing information flow and nursing tools are investigated. A children ward in Thailand has been illustrated to understand the nursing workflows in the real situation. Literatures provide theoretical backgrounds in relation to workflow management system, components and methodologies and to understand the existing theoretical techniques of enhancing workflow process. Web service architecture and related technologies are discussed.

Chapter 3 describes the proposed conceptual model of Dynamic Platform for

Workflow Management (DPWFM) for a hospital ward. Four main web service repositories will be described to understand the components of the DPWFM. These repositories are web profile service, function service, function allocation service and recompilation service. The main features of the DPWFM such as customisation and dynamic recompilation will be identified. Data, presentation and business logic are three layers in the DPWFM model to allow interacting to different user roles are introduced.

Chapter 4 discusses a prototype development framework based on the DPWFM model. Designing issues and technologies tools used will be described. We will also describe physical design for each component of the DPWFM model.

Chapter 5 illustrates the DPWFM walkthrough to present the ward management workflow. A scenario case of nursing care process is used to represent the culture of nursing workflow in a hospital ward. Firstly, nursing care with paper-based workflow will be discussed. Then, the nursing care using the DPWFM framework will be illustrated and compare to manual workflow. Several interface screens of the DPWFM are used to demonstrate how the nurses can view the web application when they use the DPWFM.

Chapter 6 shows the evaluation of the DPWFM system. There are two phrases of survey have been conducted. The first phrase of survey is to investigate the current state of nursing workflow system in the ward hospitals. The second phrase of survey is to evaluate the applicability of the DPWFM platform for ward management system.

Finally, Chapter 7 concludes the presentation of thesis and summarises research outcomes. Future research direction will be proposed.

results show nurses require simple and friendly user interface design to help them to handle the constantly changing workloads and to improve patient documentation process. The survey results also show that computerised systems of nurse notes have not been successfully deployed is nurses’ unfamiliarity to use the computer and difficulty in typing nursing note when they are in the ward taking care of patient.

# CHAPTER 2: LITERATURE REVIEWS

This chapter provides the background of nursing information system to understand the nursing workflow process in a ward hospital. The classifications of nursing roles, responsibilities and functionalities are described. A nursing scenario of the children ward in Thailand has been used to understand the nature of existing nursing workflow. This chapter also provides theoretical background related to workflow management system and business process management to understand the components and methodologies to enhance workflow process in the organisations. The web service architecture and web-based technologies are discussed.

This chapter is organised as follows. Section 1 introduces nursing roles and functions to better understand the nurse characteristics. Section 2 describes the nursing tools and documentations that are commonly used in a ward. Section 3 describes a scenario in a children ward to illustrate issues related to nursing care process in Thailand. Section 4 and 5 discuss the information flow of the nursing care process during a shift and between shifts to clearly illustrate the nursing workflow in the ward. Section 6 discusses the overviews workflow management system including their components and methodologies. Section 7 describes the related web technologies. Section 8 describes the technologies used for web services development. Chapter summary is presented in Section 9.

## 2. 1 Nursing Roles and Functions Classification

Nurses are the largest staff members in healthcare environment (Audit Scotland 2007, p3). Before continuing the discussion of nursing functions, we will clarify the term ‘nurse’ in this research. There are many levels, divisions or bands of nurses depending on their countries, qualifications and certifications. Nursing classifications can also be determined based on job descriptions and levels of payment (often called bands) in some countries.

In Australia, the Australian Nursing and Midwifery Council (ANMC) is an organisation that works with the states and territory Nursing and Midwifery Regulatory Authorities (NMRA) for setting standards for statutory nursing and midwifery regulation (ANMC 2010). In the Australian States and Territories, except Victoria, nurse-level-1 is a registered nurse (RN) who possesses a Bachelor of Nursing or Bachelor of Science (Nursing) from a university. Nurse-level-1 also includes enrolled nurse (EN) who possesses diploma from TAFE or Health Training College (Health WA 2009). In Victoria, RN is division 1 registered nurse and EN is a division 2 registered nurse (DEST 2001). Both RN and EN are two broad levels of healthcare staffs in the Australian States and Territories (DEST 2001). Other classifications of nurses are technical nurse and nurse aide. Technical nurse helps RN such as putting injection or measuring blood pressure. Nurse aide help nurses by doing routine tasks to keep patient comfortable and tend to their basic needs such as bathing and cleaning.

In the United Kingdom, the National Health Service (NHS) is the largest public funded health service (NHS 2009). NHS consists of four public funded healthcare systems: NHS England, NHS Scotland, NHS Wales and Health and Social Care in Northern Ireland. The majority of nurses who work for NHS are registered with the Nursing and Midwifery Council (NMC). NHS classifies nursing and midwifery jobs by the Agenda for Change (AfC) pay scales. For example, clinical support worker nurse is classified in band 2, advanced nurse and midwife team manager are classified in band 7 and nurse consultants are in band 8(a)-(c) (NHS Careers 2009).

In the United States, the American Nurses Association (ANA) define nurses through job functions (ANA 2010). In addition, the website of America, StudentDoc (2009) explains that nurses can be classified based on job descriptions such as nurse supervisor or head-nurse, RN, licensed practical nurses (LPN), nursing-home nurse, home-health nurse and nursing aides. In this thesis, the term 'nurse' refers to medical nurse at levels 1 and 2 with primary responsibility on patient care in the hospital ward.

### *2.1.1 Job Functionalities of Nurses*

The job functionalities of nurses can be diverse and they are involved in the patient care delivery. We identify four broad categories of nursing care functions (Liaskos and Mantas 2002, Mol 2008, St Joseph's Healthcare System 2009). These are patient care (Street 2002), ward management (Buchan 1998), communication (Antai-Otong 2007), and education and research (Cormack 2000, Craven and Hirnle 2008). Figure 2.1 depicts these four major functions of nursing.

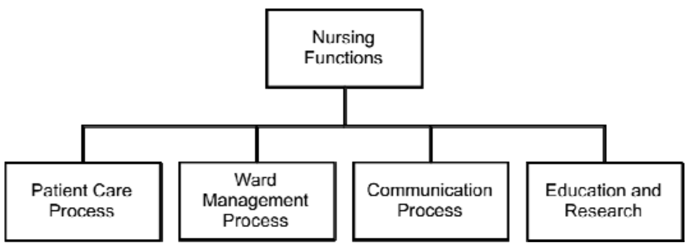



Figure 2.1 Nursing functions

In the patient care process, nurses aim to deliver healthcare professionalism (Hannah et al. 2006). Examples of patient care process include providing medical treatments, patient safety, diagnostic observations, maintaining health condition of patient, treatment of life-threatening emergencies, recovery of chronically ill, and rehabilitation of people who were injured. It also includes taking care of patient's daily hygiene such as cleaning. The nursing work functions include a combination of patient cares, medical care, treatments and nursing care tasks and liaise with allied health and alternative therapists such as dietetics, psychologist, physiotherapist, occupational therapist and music therapist. Figure 2.2 shows the combined care of various patient care responsibilities.

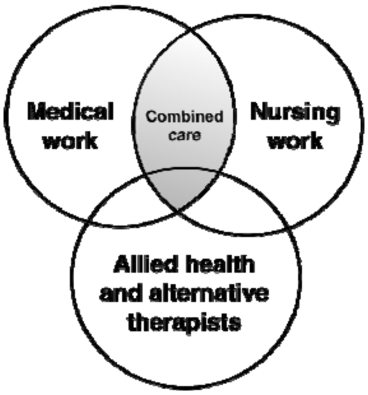



Figure 2.2 Patient care responsibilities (NIA 2004, p3)

In the hospital, nurses work in shift. There are twelve-hour shift and eight-hour shift depending on the policy of the hospital. In general, the nursing shift is organised into three common shifts per day: day, afternoon and night shifts. Each shift is usually eight hours long. However, nurses regularly have to arrive fifteen minutes to one hour earlier for transferring information session normally known as handover time. Patient information such as examination results, observations, treatments and medication schedule are recorded in nursing notes and handover sheets. The recorded information is transferred from one nursing staff to another during handover time to the next to ensure patient care process is continuous and accurate.

The transferring information process is organised in the ward meeting at the end of each shift or in the ward round at the beginning of the shift, depending on the ward manager who is usually the head-nurse, supervisor nurse or nurse unit manager. The nursing tasks and responsibilities are distributed to the nurses on duty by the in-charge nurse. The in-charge nurse is also responsible for allocating patient care tasks and schedule treatment cares plan for each patient based on the doctor's diagnosis orders. Figure 2.3 illustrates an example of the Australian nursing shift scenario of low-care hospital ward in New South Wales (NSW). In this example, there are three shifts per day. The main shift is eight hours long and the supported shifts (or sometimes referred to the short shifts) are six hours for day shift and four hours for night shift. As shown in Figure 2.3, D8 is an eight-hour day shift; D6 is a six-hour day shift; A8 and A4 are respectively eight-hour and four-hour afternoon shifts, while N8 is an eight-hour night shift. Due to the hectic nature in the day shift, the handover time from the day shift to the afternoon shift can take up to one hour long (from 2.15 pm – 3.15 pm) and it involves discussion about patients' profiles, patient's demographics and location, their observations and treatment cares plan. Handover times for the afternoon shift to the night shift and the night shift to the day shift usually are of shorter duration.

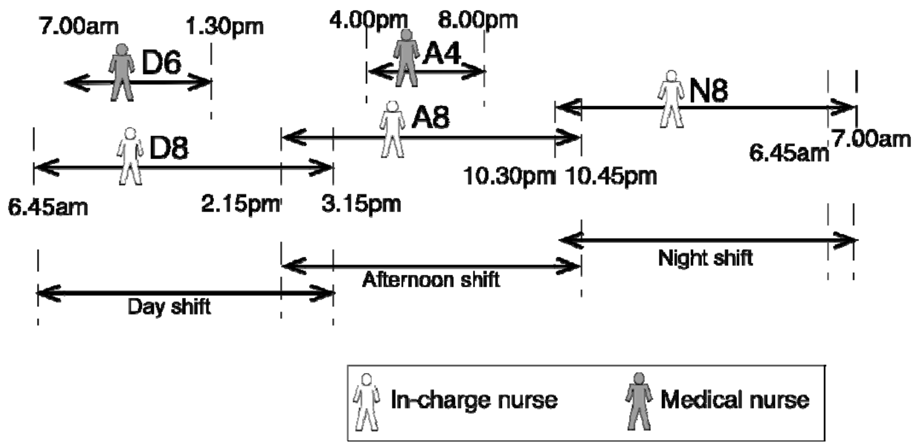


Figure 2.3 Work shifts of nursing care (based on low-care ward)

The second nursing function is ward management process. Ward management is conducted by a head-nurse, a nurse supervisor or a nurse manager (Liaskos and Mantas 2002). The main activities include organising ward and support decision-making process in managing work performances of nurses in the ward. In organising the ward, the head-nurse will distribute and organise clinical care planning, staff scheduling, inventory and supplies management, financial management and personnel management. The organisation style on a ward depends on the culture of the hospital. Thompson et al. (2001) describe that characteristics of a ward management style are unique based on individual organisation care plans. Usually each organisation or hospital has individual instructions that guide nurses in patient care.

The third function of communication process focuses on exchanging of relevant patient information to nurses on duty (Goossen 2002). It is the way to interchange thoughts and information by speech, writing and signs among nurses, doctors and patients. Information may include appointments, orders, care plan, medical treatment, scheduling, and all related messages. In addition to caring of patients, very often nurses have to educate patients and their families on individual medical condition and to promote good health and services to the patients. Thus, communicating with patients' family is also one of the activities in the communication process. Other communication activities include communication to another department such as transferring of x-ray films, lab analysis or contacting the insurance company of the patient. There are several media used in the communication process. These include nurse note and handover sheets which are usually paper-based, doctor orders (paper-based and computer-based), ward's whiteboard, computer technologies (medical software, electronic mail and

Internet) and face-to-face communication (Just 2008).

The fourth function is the education and research. These processes are the supporting functions in the healthcare (Gerrish and Lacey 2010). The health research domain includes basic science, clinical medicine, heath services, preventive medicine and public heath (HIC 2009). Information about nurse instructions, treatment procedures, infection control and documentation of routine functions are given to the nurses as a means of education and research (Cormack 2000). In addition, research domains about how to improve nurse educations for the nursing students is also part of education and research (Broom et al. 2009). It has been shown that nurse practitioners prefer continuing education in using electronic technology (Newman et al. 2009).

## 2. 2 Nursing Documentations and Tools

Nursing information system is part of healthcare and hospital information system (Liaskos and Mantas 2002). Information technologies in healthcare have attracted the attentions of improving patient cares, enhance quality of care services, safety of health-care delivery and provide quality of nursing and clinical assistance (Gedda 2008). It has been shown that applying computerised system in healthcare is able to improve nursing performance of overall clinical workflows (NHS Connecting for Health 2004, Raptis et al. 2009). Goossen (2002) also supports the belief that computerised system can help to improve better nurses communication in the healthcare environment. The more accurate the communication, the more effective the management of workflow is in the ward.

An example of health information technology is the tablet technology by Motion Computing (2007), which has shown to improve nursing productivity and satisfaction.

These technologies have also improved traditional vital signs acquisition workflow in the timeliness and accuracy of clinical documentation. According to Mahler et al. (2007) and Gedda (2008), computer-based systems help to improve the quality of nursing documentary system. Huber (2001) and Ghahramani et al. (2009) also suggest the overall work performances of nurses have increased by using the computerised system in the workplace. However, in many hospital wards, nurses are not familiar with the computerised system and they still rely on paper-based system for documentation purpose. Prinyapol et al. (2010) have shown that current generation of nurses are in the middle-aged group and they think that computers are hard to use. Thus, the level of stress of nurses involved with the computer technologies needs to be considered before computerised system is introduced to the ward (Blair et al. 1999). The observation result conducted at Port Kembla Hospital, NSW shows the similar results which indicate the majority of nurses in the wards prefer to use a very user-friendly interface system (Matnieseir 2010, Tynan 2010).

Generally, different hospitals have different patient care procedures in their workflow processes. However, the majority of information flow and the communication media remain quite the same. For example, the document tools used in the ward are the similar such as nurse note, Kardex panel or medical record. The nursing communication methods are also the same; they are mainly face-to-face and paper-based. In-charge nurse distributes the nursing plans to each nurse and transfer information between shifts during the handover time. Although there are many documentations involved in the patient care process, most of them are paper-based (Just 2008). Generally, nurses have to record everything that has happened in the ward. For example, they have to measure vital sign such as weight, blood pressure, respiration rate, body temperature, and then write these vital sign records by hands into patient's "*medical record*". To ensure correct medical dose is given and to avoid missed dose and to prevent double doses, this medical record is usually attached to the bedside of each patient. "*Nurse note*" or "*narrative nursing progress note*" is a nurse diary that contains all observations, progress notes and everything that relates to health problems of the patients, which are attached into flow-sheets called "*handover sheets*" (Craven and Hirnle 2008). Generally, the handover information includes doctor's orders, primary diagnosis, current problem, action plan, primary care team details, patient's demographics and location, nurse notes, lab requests or results (Raptis et al. 2009). At the end of the shift, this information is transferred and discussed with nurses in the next shift during the handover time (Ram et al. 2009). The nurses will forward special reminder or tasks to be continued to nurses in the following shift, such as "waiting lab results of a patient A". "*Incident report*" is the report of any unusual incident that has happened such as medication error, equipment malfunction injury to a patient, patient's family issues or problems or nursing errors or problems (Craven and Hirnle 2008).

### *2.2.1 Kardex Panel*

The nursing Kardex (see Figure 2.4) is a proprietary piece of stationery, a series of flip cards designed to record to-do-lists of all necessary nursing tasks, nursing care plans, medical treatments and medical history. It is one of the essential recording tools that uses for patient documentation for communication in the ward (Craven and Hirnle 2008). It permits the storing of a given number of patients' nursing and medical records in the ward. In addition, it also collects patient's records such as pertinent demographic data (name, age, physician, admission date, etc), lists of allergies, basic needs (diet, activities, etc), diagnostic tests, treatment plans and daily nursing procedures (dressing change or vital signs) (Craven and Hirnle 2008, Kardex 2008).


Figure 2.4 Medical panel and Kardex panels (RecordSystem.net 2009)

Kardex is a media to ensure continuation of care between nurses from one shift to another, and from one day to the next. Figure 2.5 illustrates the Kardex panels that represent three patients' plan of care using three Kardex cards in the panel. Each Kardex piece contains all tasks required for one patient. If there are twenty-five beds in the ward, a Kardex panel will contain twenty-five flipped cards. The written Kardex pieces will be placed in the nurse station, where every nurse can check their assigning tasks.

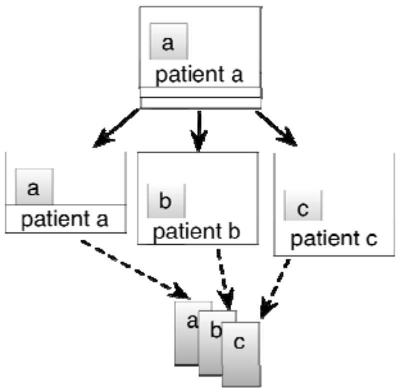



Figure 2.5 The nursing Kardex panels

Kardex is often written in pencil so that nurses can make changes when patients' conditions have changed. If it becomes a part of the permanent patient profiles, it is written in pen. However, both are the responsibilities of nurses who need to initiate and update the record. The next nursing staff can use information in the Kardex to understand patient progress and help doctor for diagnosis and prepare next treatment plan.

In task setting, the in-charge nurse assigns the daily tasks for the medical nurses. The Kardex cards serve as the means for each nurse to take care of each patient. Figure 2.6 shows workflow of Kardex cards in the system and the command hierarchy in the ward in the following steps.

1. The in-charge nurse plans and writes the task schedules for each patient into the Kardex cards according to the doctor's orders.

2. Once the medical nurses receive the assigned tasks (see Table 2.1), they have to check the Kardex cards of their patients. For example, *Nurse 1* takes care of *patient a*, *b*, *c* and handles insurance reports; *nurse 2* takes care of *patient d*, *e*, *f* and handles discharge reports; while *nurse 3* takes care of *patient g*, *h*, *i* and *j*.

3. *Nurse 1* is responsible for *patient a*, *b*, *c*, so she has to check the tasks in the Kardex card of *patients a*, *b*, *c* (see Figure 2.7) and complete the assigned tasks. After she has finished with each task, she needs to put a checked-mark in the Kardex to indicate the tasks have been completed. This helps to enable the in-charge nurse to monitor and organise the ward.

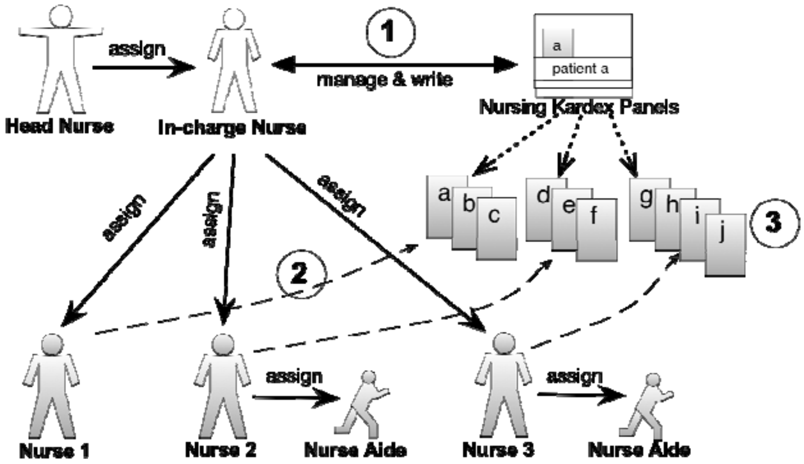



Figure 2.6 The nursing Kardex panels and a chain of commands

In such circumstances, it is clearly seen that patient record-keeping is crucial to support patient treatment processes (Thompson et al. 2001). However, the current method of transferring information during nursing handover meeting can be inefficient and information can be wrongly communicated when paper-based or oral communication between nurses on different shifts is used (Ram et al. 2009, Tucker 2009). Therefore, it is desirable to improve communication method in the ward and it is believed that a computerised system can be used to improve record keeping and communication (Daily 2004).

Table 2.1 An example of function allocation designed for each nurse

| **Date 01/01/2010** | | | |
|---|---|---|---|
| Day shift | ***Nurse 1*** | ***Nurse 2*** | ***Nurse 3*** |
| | Patient a<br>Patient b<br>Patient c<br>Insurance reports | Patient d<br>Patient e<br>Patient f<br>Discharge reports | Patient g<br>Patient h<br>Patient i<br>Patient j |
| Afternoon | ***Nurse 4*** | ***Nurse 5*** | ***Nurse 6*** |

| | | | |
|---|---|---|---|
| Shift | Patient a<br>Patient b<br>Patient c<br>Patient k (*new patient*) | Patient d<br>Patient e<br>Patient f<br>Insurance reports | Patient g<br>Patient i<br>Patient j<br>Discharge reports (*Patient h*) |

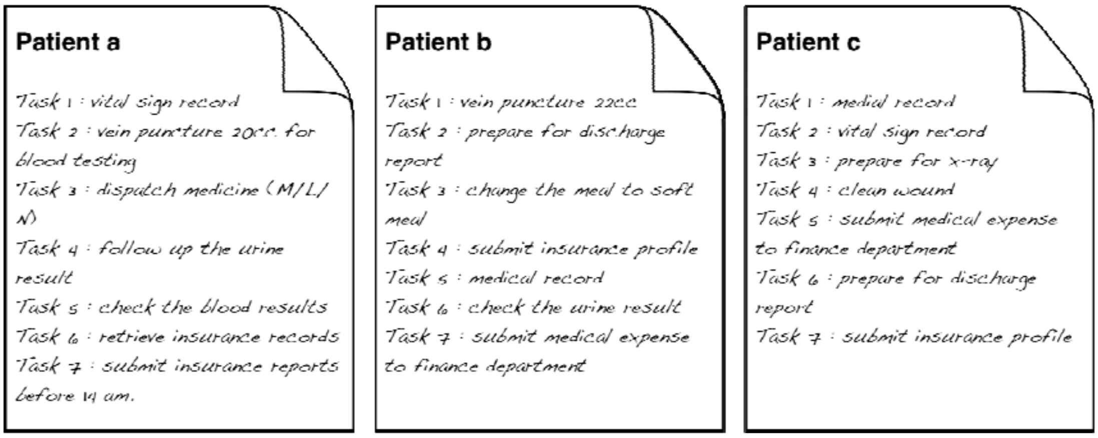



Figure 2.7 An example of assignment of tasks using Kardex cards

## 2. 3 Nursing Scenario: A Children Ward

To help to illustrate a nursing workflow, we will consider a scenario of a children ward in one of the Thailand public hospitals (Prinyapol et al. 2010). In this hospital, there is one head nurse who usually works on the day shift of eight hours long. In the afternoon and night shifts, a nurse supervisor will be given responsibilities to act as the head nurse. The responsibilities of the supervisor include managing general wards in this hospital for afternoon and night shift, order resources such as bandages or medicines, temporary reallocates nurse staffs from one ward to another if the ward is busy. Figure 2.8 shows different categories of Thai nurses working in this children ward.

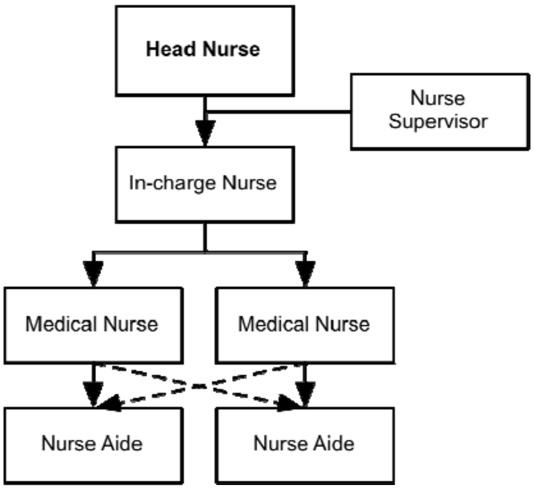



Figure 2.8 Nurse hierarchy in the children ward in a public hospital, Thailand

In the beginning of the shift, the in-charge nurse checks the doctor's orders and makes decisions about clinical routines in the ward. Their responsibilities concern with scheduling nursing care tasks and assigning tasks to each nurse. For example, blood test should be done as the first priority because it takes some time to wait for lab results, whereas requesting medicines from pharmacist or discharging patients can be of lower priority that can be done later.

Nurse aides help medical nurse to take care of patients. Sometimes, nurse aides are recently graduated nurses or apprentice nurses and they are not able to work independently without the mentor nurse (the medical nurse). In this ward, one medical nurse has two nurse aides under her supervision. There are other supporting staffs that can include clerks and patient assistants. The clerks have responsibilities on works related to processing documents, for example processing and contacting patient's insurance company and billings. The patient assistants mainly work to help patients such as helping patient to take a bath, support patient's walking and delivery of meal.

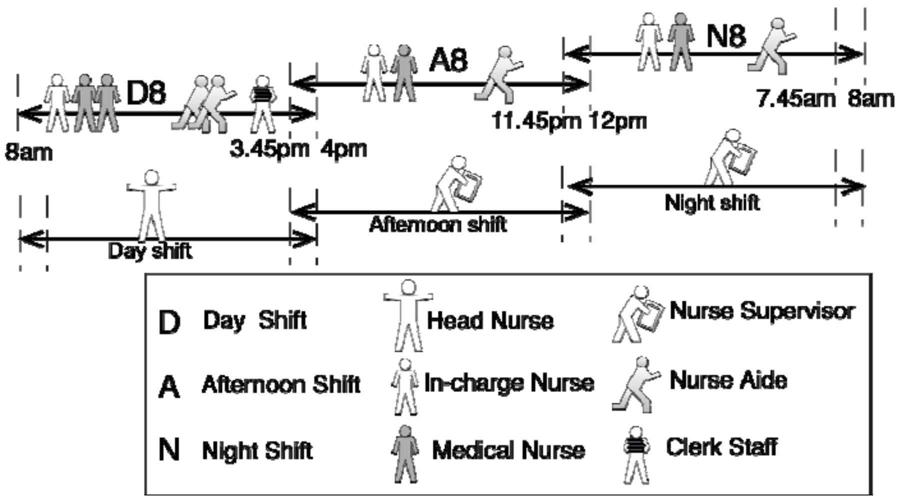



Figure 2.9 Work shifts in a children ward in public hospital, Thailand

The work shifts are divided into three shifts: a day shift, an afternoon shift and a night shift as shown in Figure 2.9. All three shifts are eight-hour long and there are fifteen minutes of overlapping for handover time. In the day shift, three nurses are required; one of them is the in-charge nurse, as well as two patient assistants and one clerk. One medical nurse takes care of eight patients in the day shift (there are twenty-four beds in this children ward). In the afternoon and night shifts, only two nurses and one patient assistant are on duty. More staffs are required in a day shift because the day shift is usually very busy with new patients being admitted to the ward and it is also the visiting time for families. On the other hand the afternoon and night shifts are quieter because these are the rest times for patients and therefore not as busy in term of workloads.

### 2. 4 Information Flow During a Shift: A Children Ward

To deliver nursing care process during a shift, there are four important roles in the workflow; they are doctor, in-charge nurse, medical nurse and pharmacist. Figure 2.10 illustrates the workflow in a general ward when staffs deliver the nursing cares to patients. The workflow can be described in the following steps:

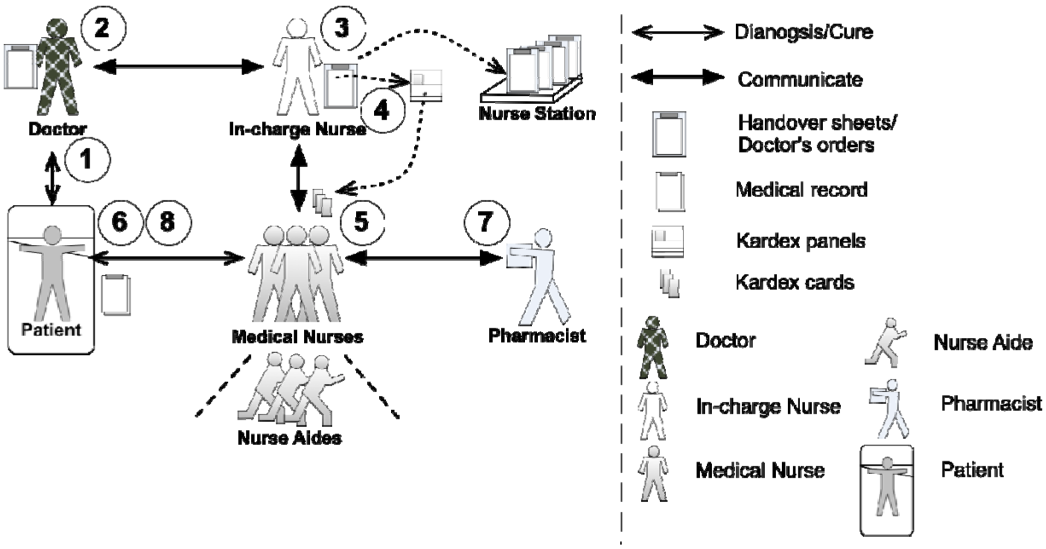



Figure 2.10 General ward workflow

Step 1: When a doctor diagnoses a patient, he writes his diagnosis in the orders (called "*doctor's orders*") in a handwritten form on the paper sheets (called "*patient's chart*" and is attached in the "*handover sheets*" with the "*patient's profile*"). The doctor's orders consist of diagnosis, drugs required and schedule treatment plans for each patient.

Step 2: The doctor handovers the orders to the in-charge nurse (or a head nurse).

Step 3: The in-charge nurse organises and arranges the nursing care into tasks based on doctor's orders.

Step 4: The in-charge nurse arranges the doctor's order into multiple separate tasks and subtasks in a sequential order. The sequence is usually based on the importance of tasks. Tasks are then allocated to individual nurse through Kardex panels. Then the doctor's orders are attached to the patient's charts and keep in the nurse station.

Step 5: The in-charge nurse assigns tasks to each nurse at the beginning of a shift (written in the Kardex) so that the nurses know what tasks they have to do. Nurses have to record the completed task in the Kardex after each task is completed.

Step 6: Nurses follow the allocated tasks to deliver nursing cares to patients. Nurses have to keep checking for new tasks and record any tasks that have been completed. In-charge nurse can monitor the tasks by checking the marked or crosschecked signs of the completed tasks in the Kardex. If any tasks are not completed, the in-charge nurse has to remind the nurses to do them.

Step 7: During each shift, nurses usually request for drugs from the medical store to fill the ward stock. Pharmacists dispense the requested medicines to nurses according to the order charts from the in-charge nurse. Normally, the order charts are handwritten by the doctor or the in-charge nurse.

Step 8: Nurses deliver medicines to each patient in the frequency arranged by the in-charge nurse.

## 2. 5 Information Flow During Handover Time: A Children Ward

As discussed, the major shift is the day shift, which requires more staffs on duty to handle the enormous amount of workload in a day. The night shift requires fewer medical nurses because the patients are mainly sleeping. The nursing care in each shift is different, however, the workflow of the nurses is more or less the same. During the shift, each medical nurse would be assigned patient care tasks and related works. The information flow during a shift has been described previously in section 2.4.

When the day shift finishes, patient information from the day shift will be transferred to the medical nurses in the afternoon shift at the nurse station or during the meeting at the handover time. Similar routine happens during changeover session in each shift. The patients' profiles with diagnosis, treatment plan and nursing care lists are transferred until the patients are discharged. Figure 2.11 shows an example of workflow of the nurses in the children ward.

In this children ward, if there is any patient with incident has occurred, only the in-charge nurse can call the doctor and report the patient observation for urgent diagnosis. After the doctor diagnoses, the doctor will append the new orders in the chart and adjust the new treatment plan. Meanwhile, nursing tasks have to be adjusted and the nursing care jobs rearranged into the Kardex cards. The doctor in this ward will round the ward three times in the day shift (at 7.30 am, 2 pm and 4 pm). After the doctor has rounded the ward, if any patient is allowed to go home then the nurses will prepare the patient discharge documents such as billing.

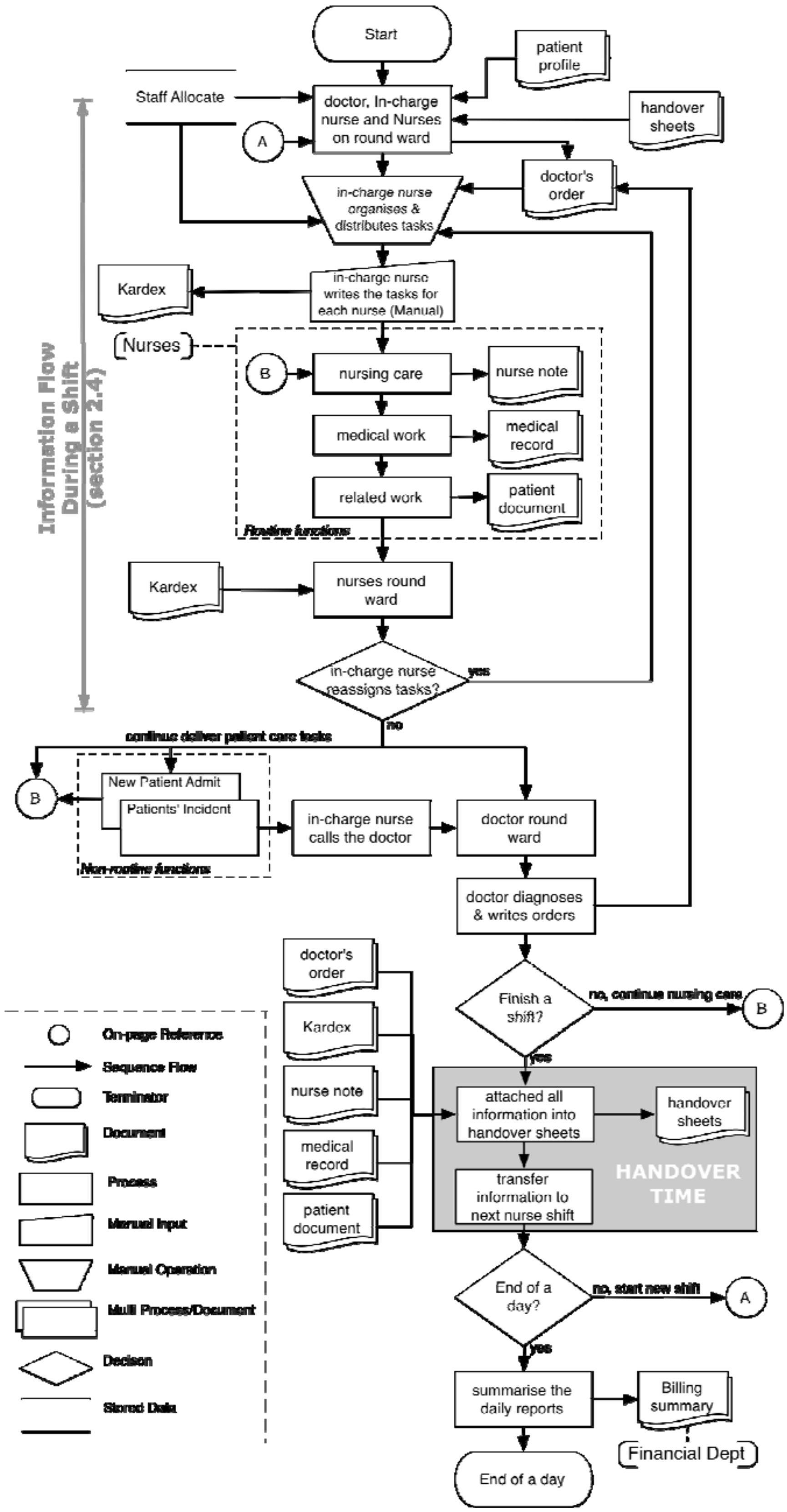



Figure 2.11 The workflow of the children ward in public hospital, Thailand

One hour before the end of the day shift, the nurses will finish their works. The clinical equipment usage will be recorded and concluded for billing purposes. The in-charge nurse will list the special observation cases to nurses in the following shift. Patients' reports will be handover to the next shift in paper-based nurse notes and communicated verbally during the round meeting. The afternoon and night shifts follow the same procedures.

Although the above example is simple, however in reality, there are twenty to thirty patients in the ward with only four or five nurses to deliver nursing care for each patient. The process can become complicated when new patients are admitted to the ward at different time. Therefore, a computerised workflow program for ward management that can improve efficiency of nursing care deliveries is desired.

## 2. 6 Workflow Management System

Over the last few decades, developments in workflow management (WFM) and business process management (BPM) have attracted increasing attention (Schäl 1998, Fischer 2003, Reijers et al. 2003). The WFM and BPM goals are to improve products and services through improving company's performance on systematic management (Chang 2006). Managing the amount of work flowing through workforces and producing maximum allowable services are focus of the most successful businesses (Fischer 2007). Workflow is a way to handle processes in an organisation to become an automated business process by human or automatic executors called agents. It combines not only simple tasks (such as transferring documents), but also complex tasks and activities such as scheduling tasks for employees constraint by times, places, rules, policies and resources (DiCaterino et al. 1997). Workflows are designed to be more efficient in supporting business (Swenson 1995). Examples of workflow systems include booking airline tickets (Jipiaoneta 2008), registering memberships (Kulliyyah 2009), taking care of patients (Ma 2005), providing chemotherapeutic treatment for patients (Lyng et al. 2009), claiming life insurances (van der Aalst and van Hee 2004) and marking of student grades (Plimmer and Mason 2004). In general, workflow consists of work pattern, which has a systematic control of resources, rules and flows of information. The general components of workflow include input descriptions, transformation rules and output descriptions.

Figure 2.12 shows the workflow process components based on literature from researchers such as Curtis et al (1992), Kwan and Balasubramanian (1997), Grefen et al (1999), Basu and Blanning (2000, 2003), Brambilla et al. (2006) and Jablonski (2008).

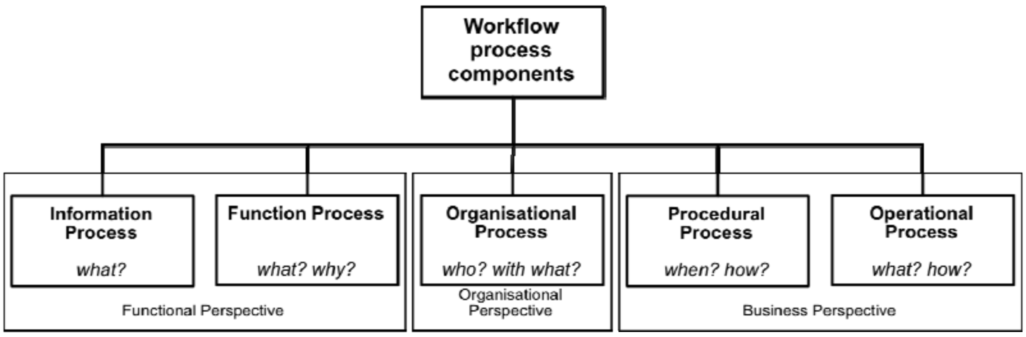



Figure 2.12 Workflow process components

Grefen et al. (1999) analyse the workflow process components into three perspectives: functional, organisational and business. They propose their three perspectives of workflow to separate business activities from functionalities and organisational aspects. Firstly, the functional perspective involves information object and process activities is divided into information and function processes. Secondly organisational perspective is concerned mainly with execution of organisational processes with their agents and roles. Finally the business perspective includes business rules and objectives of the processes.

Functional perspective consists of information and function processes. Information process identifies the information of each activity and answer the ‘what’ question. Function process describes the tasks and subtasks in each activity and requires the conditions of using it so that it can answer ‘what’ and ‘why’ questions.

Other researchers also divide functionalities into information and function processes (Curtis et al. 1992, Kwan and Balasubramanian 1997, Basu and Blanning 2000). They classify the workflow process components into four perspectives, which include informational, functional, organisational and procedural processes. There are methodologies that implement these four workflow models separately (Curtis et al. 1992). On the other hand, there is a new approach to integrate these four components into one single model and form a foundation for supporting workflow management with computer-based tools (Basu and Blanning 2003, Brambilla et al. 2006).

Organisational perspective contains organisational process that identifies workers and their roles. With the organisational process, we know ‘who’ use the system and with ‘what’ activities or tasks belong to these workers. Business perspective consists of procedural and operational processes. Procedural process describes the transactional, behavioural and sequential steps of the activities in the system. It includes methods and constraints of the tasks by organisational policies according to business rules. It can answer the ‘when’ and ‘how’ questions. Another researcher, Jablonski (2008) classifies an operational perspective as being a part of business perspective. Operational process identifies the resources used in each activity and their procedures to complete the tasks.

It can answer the ‘what’ and how’ questions.

In the last few years, the workflow research has focused on continuously applying web-based system to support the WFM, which is an automation of procedures or workflow system that defines, creates and controls a workflow running through a workflow engine governed by procedures or rules (WfMC 1995, Fischer 2003). It consists of a process design, system configuration and process enactment. This includes reducing the risks of costly corrections, time delays, insufficient resources or human errors. One of the goals of the WFM is to manage flows of activities or events. To make effective use of workflow performances, WFM is integrated with other technology such as web services (Walker et al. 2005).

There are many workflow methodologies reported in the literature. These workflow models have been set up to improve efficiency, consistency and quality (Swenson 1995). Figure 2.13 shows the enhanced techniques of developing workflow process based on three workflow methodologies of Vila et al. (2007), Blakely (2001) and Cichocki et al. (1998). Vila et al. (2007) propose the workflow methodology for supporting collaborative design and manufacturing, their models are divided into three parts: (1) describe the collaborative environment, (2) define the lifecycle phrases and organise the workflow and (3) execute the workflow model. Blakely (2001) describes five steps of workflow methodology to expedite the organisation. These five steps are (1) *inform* whole pictures of the new project to users, (2) *discover* the real needs of users and the directions of business, (3) *plan* visual presentations and prototypes for the specific solutions to users, (4) *build* solutions with testing plans and (5) *evaluate* the solutions. Cichocki et al. (1998) describe three main categories of workflow modelling methodologies: (1) communication-based, (2) artifact-based and (3) activity-based. Communication-based methodology is the conversation between a customer and a performer. This action model consists of request, negotiation, performance and acceptance. Artifact-based methodology is object-based modeling. For example, when a product is created, modified and used in the workflow activities, it has to be determined and described in the chain lines such as what are the orders, equipment or who are the users. On the other hand, activity-based methodology plays an important role in specifying the workflow. It focuses on tasks and order of each task in the workflow chains.

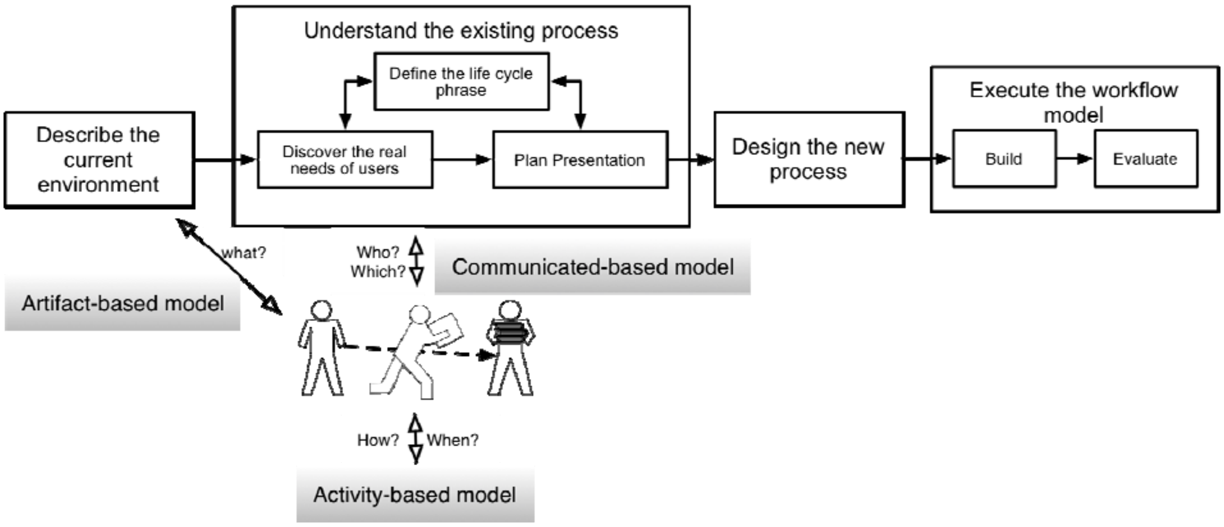



Figure 2.13 Technique to enhance workflow process

A business process is a series of steps or events that focus mainly on business logic. It explains how an organisation produces and delivers the specific products or services to the customers or markets. It consists of human, authority, organisation, policies, rules and procedures to govern the activities ordered in time and space dimensions. There are three logical components of business process. They are management, operation and support processes (Gelinas et al. 2004). Firstly, the management process plans and controls an operation process. Secondly, the operation process is part of a core business such as sales, marketing and purchasing. Finally, the supporting process plays a critical role in supporting all processes to work together.

BPM is known to be an extension of WFM. It includes a diagnosis process which provides methods, techniques and tools to complete a workflow lifecycle (van der Aalst et al. 2003). Figure 2.14 shows the relationships between WFM and BPM. The white arrows represent the components of WFM, which are process design, implementation and process enactment. The gradient arrow represents the last component of the BPM that is the diagnosis process.

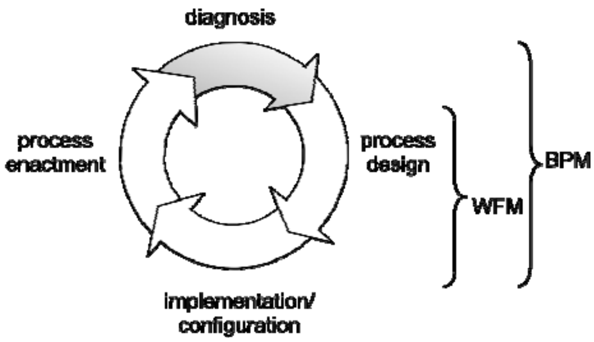



Figure 2.14 Relationship of WFM and BPM (van der Aalst et al. 2003, p5)

Organisations have started to improve their efficiency in business processes. By running the business in a better, faster or cheaper services and able to survive in the marketplace, enhancement of their business processes is considered to be the key competitive weapon to succeed their competitors (TIBCO 2007). Prior to be able to improve the business process, an understanding of the existing business process is essential. Business process discovery (BPD) is a set of techniques or tools used to analyse and understand business process (TIBCO 2007). It is important not only to understand the ‘as-is’ of the processes, but also to discover hidden opportunities, which can help to improve the business processes. To analyse the as-is, the existing process diagrams, standard operation policies, systems and sub-systems and people conducting the processes need to be identified. To discover the hidden processes, walkthrough for observation of the workflow processes is required. Sometimes, the workers may not be able to tell exactly what they have done in their daily tasks, however they know what went wrong or can be improved in the existing processes.

There are several workflow languages and notations that have been developed such as YAWL (yet another workflow language) and business process modelling notation (BPMN). The YAWL is a workflow language based on workflow patterns and offers more flexible features based on Petri net. There are three concepts of YAML, which are not supported in Petri net such as or-join, cancellation sets and multi-instance activities. Figure 2.15 shows an example. Its engine supports the control-flow perspective and data perspective based on Service Oriented Architecture (SOA). It is commonly applied in film/TV production, BPM and healthcare sector.

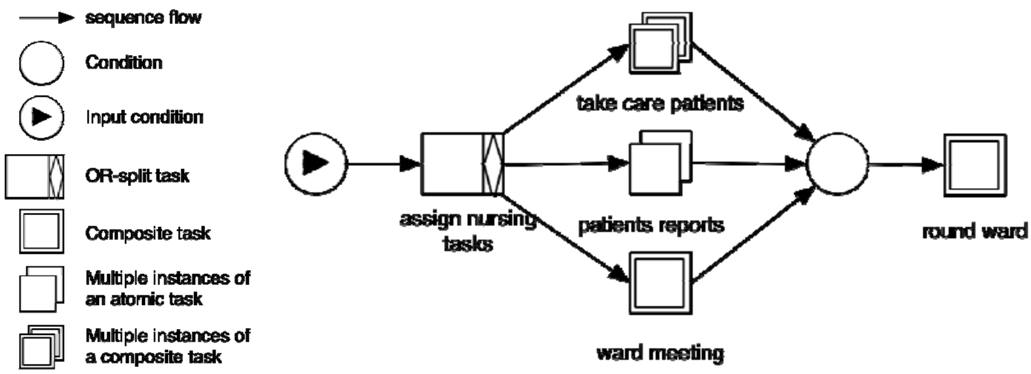



Figure 2.15 YAWL representation of assignment of tasks to nurses in a general ward

BPMN is a standard graphical notation that supports business process. It was developed based on flowcharting and is similar to activity diagram and unified modelling language (UML). Figure 2.16 shows an example.

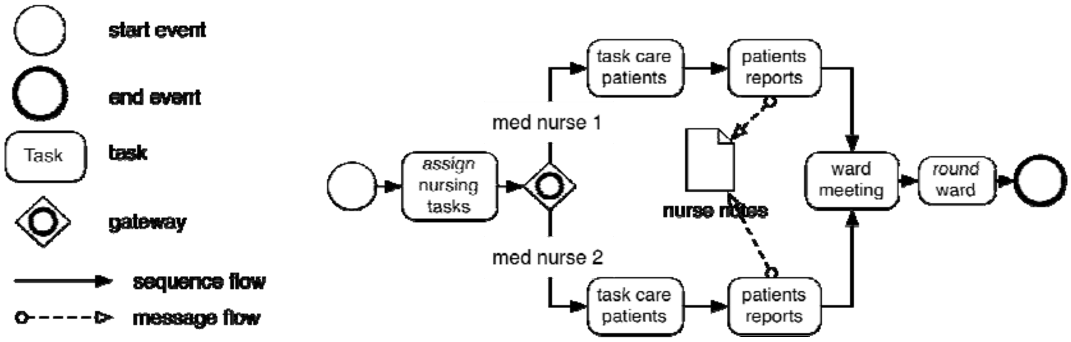



Figure 2.16 BPMN representation of assignment of tasks to nurses in a general ward

Services, in the business perspective, are the business activities traditionally communicating from human to human (Baida et al. 2004, Gordijn et al. 2006). E-service is an extended version of a traditional commercial service over the Internet (Baida et al. 2005, p211). It consists of business activities and information system and allows human-to-application communications. Recent studies show that many researchers have focused on application-to-application communications (Mancini et al. 2006, Chukmol 2008, Papazoglou 2008). These automated communicating technologies called web services can enhance business processes and provide a more flexible and dynamic interoperability among partner organisations (W3C 2006).

## 2. 7 Web Technologies

According to Zimmermann (2003), the behaviour of the organisations which use web services to do their business has changed. The communications from humans-to-applications to applications-to-applications is an ideal platform to implement web services (Berardi et al. 2005). Zimmermann (2003, p4) explains that the requirements of web service technologies should be automation through application clients, connectivity for heterogeneous worlds, information and process sharing, interface agreements, reuse and flexibility, dynamic discovery of service interfaces and implementations and business process orchestration without programming.

Definitions of web services are not always interpreted with the same meaning (Barry 2003, Alonso et al. 2004, Baida et al. 2004). Web service is a web application that provides available capabilities to other applications (Alonso et al. 2004) or other web services (Chatterjee 2004) over a network. Web services consist of software components that are designed to support automation interactive between machines over the network (IBM 2001, W3C 2004). It allows another web service consumer to invoke the application via its interface (Chatterjee 2004). The two protocols of building web services are Web Service Description Language (WSDL) and Service-Oriented Architecture Protocol (SOAP). WSDL is an XML-based language that provides a model to describe web services as the collection of network endpoints to other software agents (Zimmermann 2003, Chatterjee 2004). It includes description on how to access web services and which message format can be sent among servers (W3C 2001).

Web services based on XML standards are platform and language independent. It offers many benefits such as interoperability, usability, reusability and deploy-ability. Web service developers provide their service solutions through that Internet so it is more beneficial to web services to have longer life span for service consumers to request. The usability of web services provides business logic of many different systems to be exposed on the Internet. There is more freedom in choosing the needed web services by only including specific business logic into the client-side of the applications. The reusability of web services provides an easy way to develop application. It is a component-based model that makes it easy to reuse web service components. The deploy-ability of web services makes web services possible to deploy over standard Internet technologies and also over the firewall or over any proven community standards. The web service can provide service for one task or it can link with other services that can perform a series of tasks to deliver a composite results based on the requirement and the complexity of the job requested. As a result, all web services can be used to assemble and design different types of application based on the functional specifications required in solving the problem.

Figure 2.17 illustrates the web services interoperability between service providers and service requesters. Firstly, the service providers publish WSDL documents and register them in UDDI (Universal Description, Discovery and Integration) as a service broker. The UDDI provides an XML-based registry into service repositories. Then, when the service requester calls for web services, the WSDL request is sent to find the matching web service in the service directory. Finally, the SOAP messages are retrieved from the service providers that have the matching web service with the requests based on the description in the WSDL document.

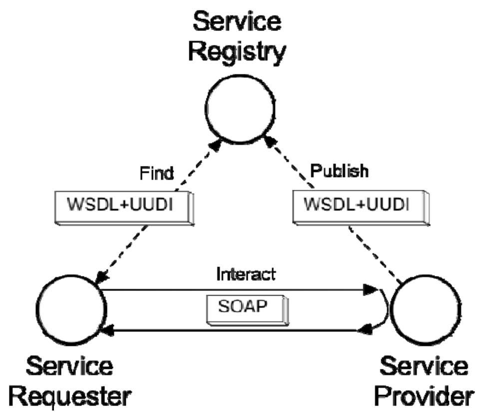



Figure 2.17 Web service architecture (IBM 2001, Vaughan-Nichols 2002, W3C 2004)

SOAP is an interoperability protocol to exchange XML-based protocol over the Internet

(W3C 2004, Newcomer and Lomow 2005). It consists of three parts: an envelope that defines what is contained in a message, how to operate it and who provides it; an encoding rule that can be used to define instances exchanging of defining application data types and represent procedure call (W3C 2000).

Figure 2.18 illustrates a basic communication of SOA. Firstly, a requested service is sent to a service provider. Each of the events can trigger the signal to be sent to the service provider and the signal will be stripped and be interpreted by the XML parser. The information carries the instructions to retrieve the web service application/object where information from the signal will be used to execute web service response. Then, the service provider returns a service response message back to the service consumer.

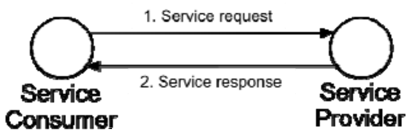



Figure 2.18 A basic service-oriented architecture (W3C 2004)

SOA is an improved architecture integrated with a programming model in designing a software system, supporting services to other applications over a network (Channabasavaiah et al. 2003, Panchenko and Zeier 2008). SOA has reduced the cost of maintenance and improved the return of investment of a system because SOA does not require re-writing of full-scale system.

## 2. 8 Technologies for Web Services

A markup language describes a document by placing tags in the document. Hypertext markup language (HTML) is a well-known example of a markup language. HTML comprises a finite set of tags, however it does not contain functions, conditional logic or loop (Livingston 2002). Extensible markup language (XML) is a specification for creating its own mark-up language (Riehl and Sterin 2003). While HTML is responsible to provide the static contexts, XML displays dynamic contexts. It allows web developers to create special tags, which are not only given a meaning to stored services but also to display contents of the webpage (Holzner 2004). Moreover, XML is a machine-readable language that allows XML to use on all platforms. XML is flexible to use, easy to work with, low start-up cost, open standard and Internet-friendly. Its primary purpose is to facilitate sharing of data across different platforms. Therefore, XML has been integrated into several development platforms and create more efficient web applications. Many languages work with XML such as Perl (Riehl and Sterin 2003), Java (Harold 2003), or C/C++. Existing database is converted to XML standard to use in the proposed system. Then, it is converted to the original format when transfer back to the database.

Another powerful cross-platform scripting language that enables developers to work with XML is PHP: Hypertext Preprocessor (PHP) (Gutmans et al. 2005). It is an HTML-embedded scripting language that composes of dynamic web pages and a server-side application software (Morgan 2007). PHP5, the latest version, is improved to better support for XML. It helps the web developers to write dynamically generated pages more efficiently. ASP.NET is another server side scripting technology that is the next generation of Active Server Pages (ASP) to better support for developing web applications (Bellinaso 2006). The ASP.NET file is the same as an HTML file. It contains HTML, XML and scripts that able to execute on the server.

Asynchronous JavaScript and XML (AJAX) is used to create simple user interface and provide dynamic exchange when users retrieve information asynchronously in the background without interrupting their existing activities. AJAX has the advantages from both JavaScript and XML. Hence, XML not only changes the way to communicate among application platforms, but it also helps to enhance the interactive nature of web applications. JavaScript is a cross-platform and object-oriented scripting language that is the lightweight of Java-based languages. JavaScript processes the data in the background and renders it through a document object model (DOM). XMLHttpRequest is an object that lets clients to submit and retrieve XML data from a server without refreshes it again via loading web pages. Therefore, AJAX enhances an existing web application, faster interaction with users and reduces the lag times of retrieving information.

## 2. 9 Chapter Summary

Literature review shows that the current state of the ward management is very complicated, nurses in the hospital spend a lot of time dealing with day-to-day work responsibilities. They use paper-based reports that can be inefficient (Rajeev et al. 2010). However, nursing staffs usually do not want to use computerised system if it requires intensive trainings (Tynan 2010). It is difficult to design and develop a computerised workflow system that is easy to use and meet the individual needs of each ward.

Current research shows that web services technology is an appropriate technology for communicating between any types of application on any platforms. Web services allow business logic of many different systems to be exposed on the Internet. They also allow web developers to use programming languages independently and provide a component-based model that is easy to reuse. Many prior studies have reported using web service technologies in the workflow management (Alonso et al. 2004, Chatterjee 2004, van der Aalst and van Hee 2004). However, practical applications of web services in particular for the ward workflow management in hospital have not been thoroughly investigated.

In this research we will be proposing a nursing ward workflow management system using web services technology. There are two main features in our proposed research approach of applying web services; they are customisation and recompilation. Customisation feature allows users to modify their function services to suit nursing tasks based on change of requirement. Dynamic recompilation allows automated recompilation of web services from multiple web service repositories. As there are not many research mechanisms emphasising on web service workflow recompilation in the nursing care environment, we will investigate a framework for workflow management system to manage the workflow in a hospital ward to help nursing staff to improve his/her job efficiency.

# CHAPTER 3: DYNAMIC PLATFORM DEVELOPMENT FOR WORKFLOW MANAGEMENT

This chapter describes the proposed conceptual model of Dynamic Platform for Workflow Management (DPWFM) for ward management system in a healthcare environment. Four main web service repositories of the DPWFM will be discussed: web profile service, function service, function allocation service and recompilation service. The customisation and dynamic recompilation features of the DPWFM will be described.

This chapter is organised as follows. Section 1 discusses the proposed conceptual model of the DPWFM. Four web service repositories of the DPWFM and their relationships will be described. Section 2 describes the mechanism deployments of the proposed DPWFM. Section 3 illustrates the recompilation feature of the DPWFM model. Section 4 illustrates the deployment of DPWFM services. Section 5 explains the three layers of the DPWFM for three user roles. Section 6 concludes the chapter.

## 3. 1 Proposed Conceptual Model of the DPWFM

As discussed in chapter 2, it is very difficult to design and develop a ward management system that is easy to use and one that fits requirements of any ward and can be applied to a standard service model. Therefore, we should aim to allow users to customise their requirements by designing and developing a system that can be dynamically recompiled and at the same time fulfils service requirements of the users. To achieve an outcome toward this direction, web service technology is proposed. The dynamic recompilation feature of the system will not only be easy to use, at the same time it enables nurse supervisors to customise the ward management based on individual ward circumstances. The two important features of the proposed system are customisation and dynamic recompilation. These features aim to allow nurses to customise individual requirement by designing and developing a system that can be dynamically recompiled and at the same time fulfils service requirement of individual nurses. Survey results by Prinyapol et al. (2010) show that although younger generation of nurses are familiar with computer technology, a large number of older nurses still find computer technology unfamiliar.

Customisation allows users to modify functions within the web service repository to suit individual tasks based on their current situations. On the other hand, dynamic recompilation allows services from multiple web service repositories to recompile and arrange into a new set of dynamic functional services for the users. Furthermore, advances in web technology such as AJAX technology provide an easy platform to develop such a web service application. Literature review from chapter 2 also shows that little research has been conducted in the deployment of web service application for healthcare information system, in particular in the area of ward management. In this research, we propose a dynamic recompilation service framework for ward management. A workflow management system consists of web services based on web services repository will be described. The purpose of developing the DPWFM model using web services technology is to provide a standard platform for interoperability and is best suited to work across multiple platforms. In addition, web services are reusable and have the flexibility to be platform independent (Chukmol 2008).

As previously discussed in chapter 2, workflow management can be improved by considering the following three perspectives: organisational, functional and procedural processes (Grefen et al. 1999, Basu and Blanning 2003, Brambilla et al. 2006). In this research, we propose a framework of the DPWFM model based on these three perspectives. Figure 3.1 shows the conceptual framework of interactions between the three components to generate result as an agenda workflow. Organisational perspective describes the characteristics of the nurses such as organisational roles, qualifications and job descriptions. The functional perspective describes task-related requirements such as task descriptions, task constraints and sub-task descriptions. The procedural perspective describes operational setting, business logics and business rules. These three components will interact to generate results in a form of workflow using a recompilation process taking into consideration inputs from organisational, functional and procedural perspectives.

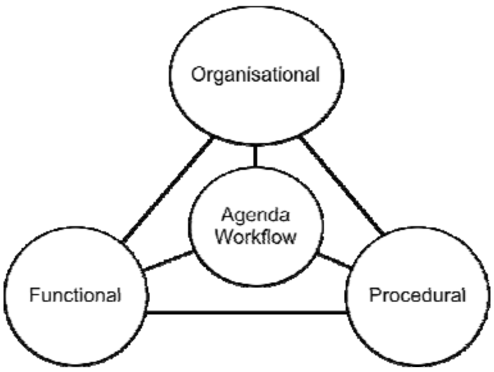



Figure 3.1 Conceptual workflow components for the DPWFM

The proposed conceptual framework will be developed using web services and dynamic recompilation being its main feature. The recompilation of web service refers to the reorganisation capability of workflow when job functionalities, processes, operations and organisational changes occur in the work environment. The proposed framework is particularly useful on workflow assignment and scheduling in situations where supervisors are responsible for assigning workflow-related tasks to their subordinates such as supervisor assigning projects to subordinates and manager assigning tasks to the workers.

Figure 3.2 shows the proposed DPWFM architecture consisting of four web service repositories that include: work profile service (WPS), function service (FS), function allocation service (FAS) and recompilation service (RS).

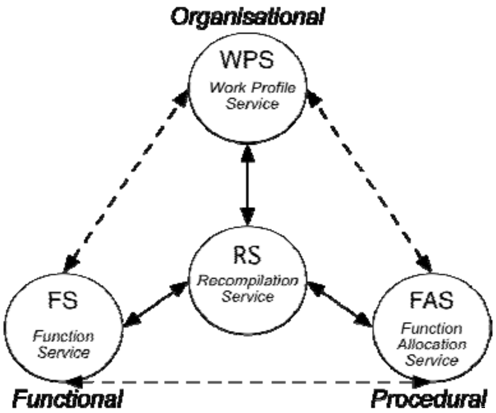



Figure 3.2 Web service repositories of the DPWFM

We consider the WPS as the organisational aspect, FS as the functional aspect and FAS as the procedural aspect. The organisational aspect of the WPS defines who will take which roles in the department. The functional aspect of FS describes the activities and services that include pre-conditions and predecessor ordering of tasks. The procedural aspect of the FAS provides the allocated services that are customised by a supervisor for each subordinate. Table 3.1 summarises the description of these four components.

Table 3.1: Four web service repositories

| Service repository | Descriptions |
|---|---|
| Work Profile Service (WPS) | Web service repository that describes and stores worker descriptive information such as level of qualification, personal data from existing database, career positions, job description, main responsibilities, work experiences and specialist skills. |
| Function Service (FS) | Web service repository that identifies and classifies the business process and company procedures into function and sub-function services such as task and subtask details including restriction, predecessor and condition and pre-condition of each function service. |
| Function Allocation Service (FAS) | Web service repository that gathers and customises the assigned functions based on supervisor's decisions. |
| Recompilation Service (RS) | Web service repository that recompiles appropriate scheduled function services to users. |

Figure 3.3 shows different types of users who will be involved in the DPWFM model. Different user roles determine the way users control their workflow processes in the department and the functionalities that each user has accessed to. The administrator is the system administrator who is responsible to maintain the system such as extracting information from existing databases and organisational activities processes and transforming the information into web service repositories. The supervisor is able to customise tasks and workflows using web service repository and the nurses use the web application that has been authorised to fit into individual work processes.

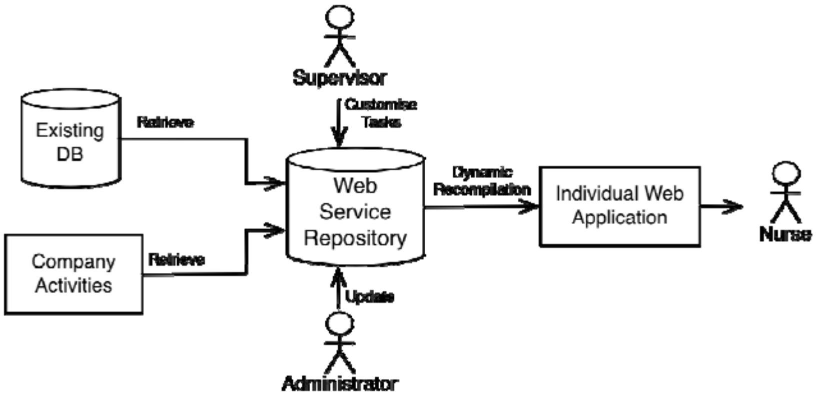



Figure 3.3 Different user role models for the DPWFM

### 3. 2 Proposed Mechanisms of DPWFM Model

One advantage of the DPWFM model with different user roles is a feature of customises necessary functional services and presents to the targeted users. For example, supervisor can select and customise tasks based on information setup by the worker profiles in WPS and function requirements in FS (see Figure 3.4). Here, the supervisor assigns nurses the tasks by retrieve information of worker profiles and determines related function requirements. Thus, the requirement inputs are entered into relevant web service set. This customising feature enables nurses to retrieve the relevant information for their assigned tasks. Therefore, a ready-to-use function services can be customised for individual nurses.

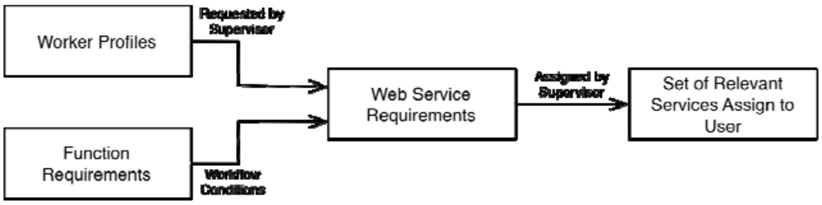



Figure 3.4 Proposed mechanism of customise web services

Let us apply the DPWFM model to a nursing ward management. In this case, the WPS is linked to a database that stores information on responsibilities, roles and tasks of each nursing worker such as RN, EN or nursing supervisor. Their individual job responsibilities and task functions associated with their levels of employment are recorded in the WPS. The FS consists of a set of web services that describes each function associated with the workflow system. In the case of ward management, it may include processes such as patient check-in procedure, which may includes subtasks such as name tag identification, bed allocation, temperature taking, height and weight measurement. The third component is the FAS that contains task assignment. The task assignment refers to allocation of appropriate tasks by a supervisor to each subordinate. The task assignment is based on individual job requirement, roles and responsibilities. Supervisors can access and modify the assignments at any time. Finally, the RS is the assignment and allocation of work schedule using inputs from the WPS, FS and FAS and the resultant web application provides a recommended schedule or sequence for the staff.

The main contribution of the proposed framework is the recompilation capability of web services that support dynamic changes of workflow based on individual work environment. It organises the relevant function services for each nurse into a systematic arrangement in the form of web-based application. Figure 3.5 shows the proposed mechanism of dynamic recompilation services. The type of input interface of each function service has been determined at the design state. An interface-customise generator has the interface design capabilities to suit different devices such as computer, portable device or PDA. The predefined interface components are integrated with the relevant function services such as uses checkboxes for selecting the relevant tasks or uses drop-down lists for the services allocation. Therefore, the interface generator can be designed and used to provide suitable interface components to the regular work assignments. Next, a set of function allocation services and the interface components are rearranged and output as an assignment scheduling. Then, an agenda task list is generated after dynamic recompilation and the nurse is presented with the scheduled activities to perform.

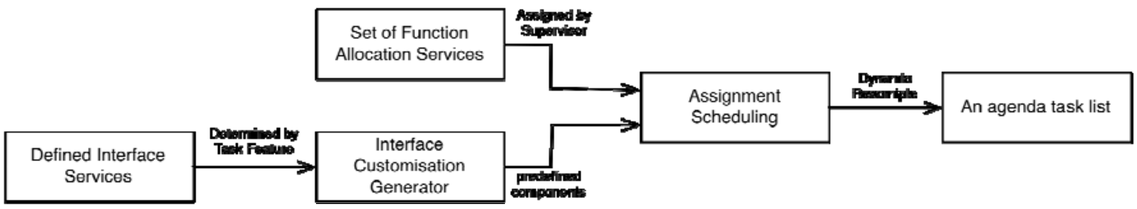



Figure 3.5 Proposed mechanism of dynamic recompile services

Figure 3.6 shows the resultant recompiled services for the DPWFM based on the above descriptions. Rather than retrieving the existing database and extracting company activities directly into the web service repository, the WPS and FS are predefined into the web service repository of DPWFM.

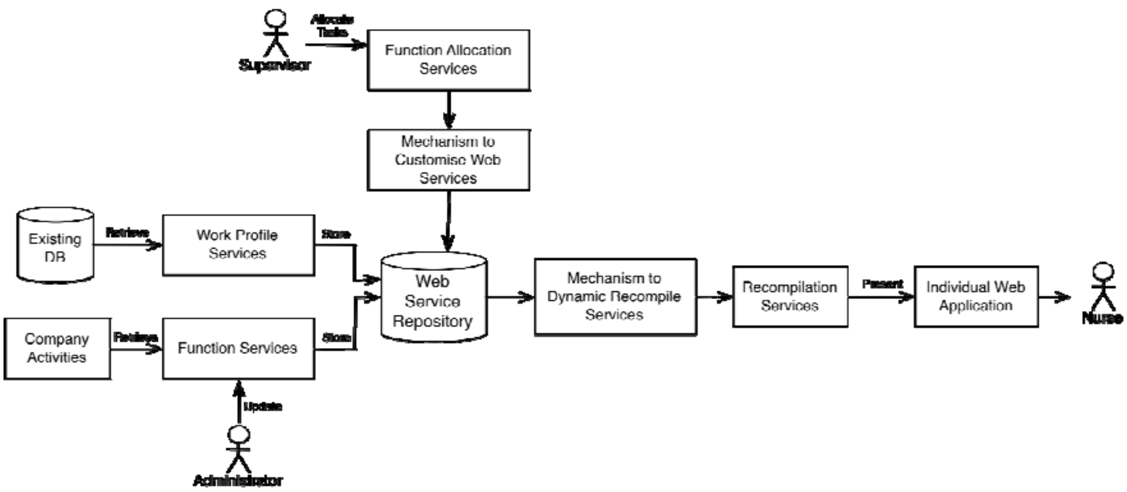



Figure 3.6 Diagrammatic representation for the DPWFM model

The proposed DPWFM model can be applied to any different types of workflow environment. The web services repository framework means any organisation with different setting of databases and process flows can be used. The existing databases can communicate with the DPWFM server by translating raw data to the standard format of the proposed DPWFM framework. The DPWFM server is for retrieving and executing the web services based on the requirements of the workflow functions needed for each individual ward. Figure 3.7 shows the database model for the DPWFM framework. The information of nurses are converted and kept into the profile database set of the WPS repository called ProfileDB. The profile of nurse, role responsibility and role description are predefined for the WPS repository (see Figure 3.7a). The tasks information and rules are stored into a set of the FS repository called FunctionDB. It contains priority, procedure and time sequence of each task. This includes subtasks associated with each task and function (see Figure 3.7b). The function tasks of the FS and nurses information of the WPS are assigned to the tasks in the FAS by supervisor (see Figure 3.7c). The FAS will use the information setup by the WPS to work out a planned priority schedule for the work in a form of tasks list. The supervisor can customise the tasks and record assignments of tasks into the FAS repository for each nurse. If the assigned tasks are related to the patient, the patients' profiles will also be retrieved into the FAS. The priority task information of the FS and the FAS support information to the RS and then the agenda workflows will be generated for individual nurse (see Figure 3.7d).

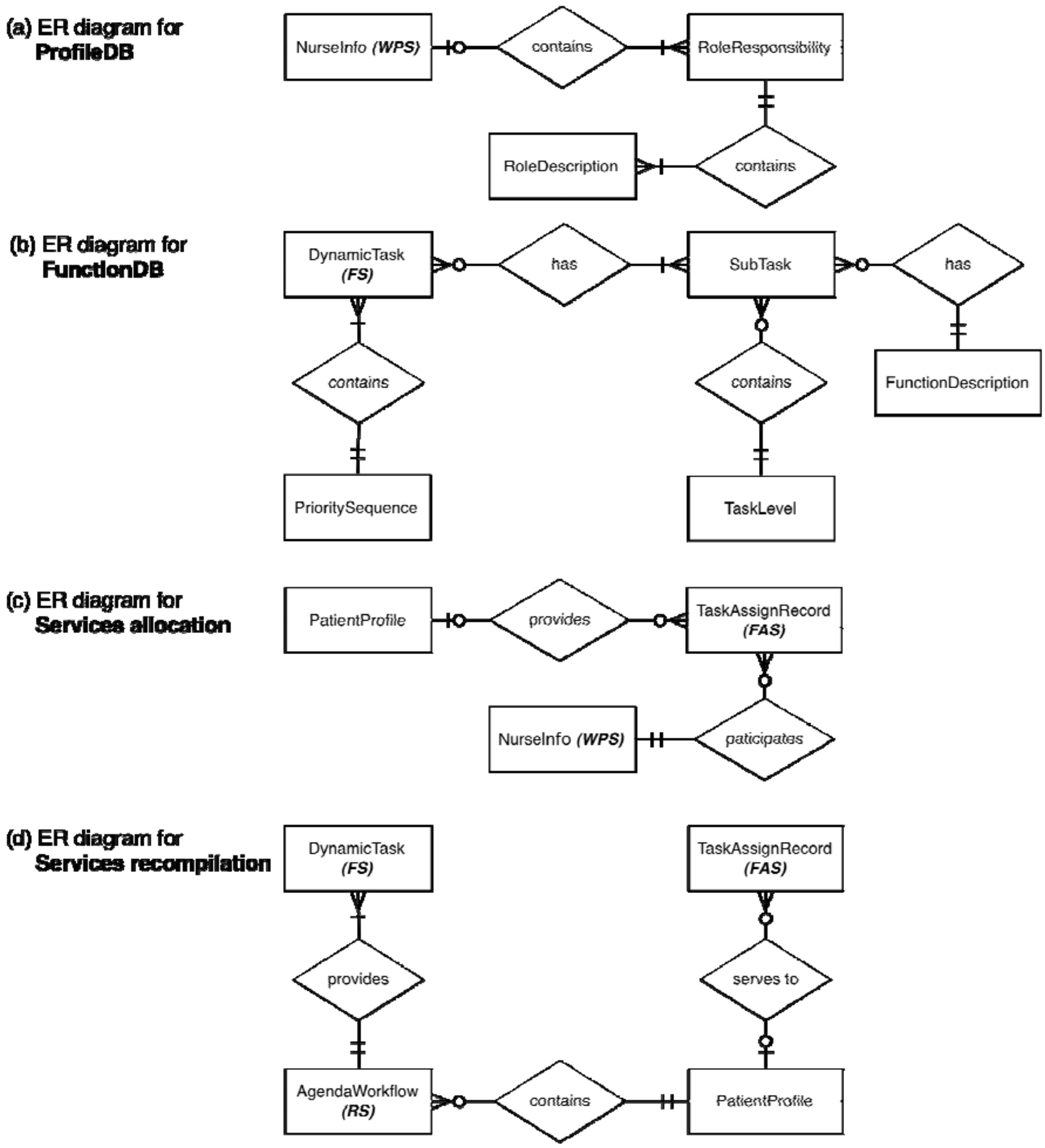



Figure 3.7 Database relationship model of the DPWFM

The following describes how web services can be established in the four web service repositories. To identify users of the system, job descriptions of the nurses are stored and described in the WPS. It includes roles, qualifications, access authorisations, career positions, main responsibilities, main tasks and specialist qualifications. Supervisors and managers will determine what role is suitable for each subordinate based on departmental rules. For example, a nurse can have more than one job functions in a ward such as a general nurse who has the role of ‘medical nurse’ and can be assigned as ‘an in-charge nurse’ for a particular shift at the same time. In another example, the role of ‘medical nurse’ can have different levels of authority to do the tasks depending on their job descriptions such as ‘junior’ or ‘senior’ position.

The FS records business functions into the DPWFM server. The business processes are translated and stored in the function components of the FS and the process relationships are described into the FS predecessor information. The FS manages and collects these business service functions into web services. Each service combines a set of sub-functions or collections of tasks. For example in a nursing care, the function service of ‘writing a report for patient A1’, consists of five sub-tasks such as (a)‘record vital signs of the patient’, (b)‘fill-in dose of drug’, (c)‘fill-in frequency of drug’, (d)‘collect lab result’ and (e)‘note special requirements’. These sub-tasks are reusable for another function service such as ‘discharge a patient process’.

The FAS gathers inputs from supervisors. The supervisors allocate the related FS to their subordinates via web-based application. The web application provides the appropriate task list (such as presenting the FS services with the combination of the checkboxes) to allow supervisors to assign specific tasks to each nurse by checking the checkboxes to retrieve each related FS service. Simple user-interface has been generated for individual nurse, which consists of the assigned FS and the predefined interface components. This interface generator makes it easy to use and reduces confusion in interpreting assigned tasks to individual nurse, thus allowing the supervisors to allocate appropriate tasks to their subordinates. The customisation paradigm provides a more flexible and manipulable interface when retrieving the FS and able to meet nurses’ requirements.

The RS performs the operation of comparing, sorting and prioritising the task list to prepare for an agenda workflow after the FAS completes selecting the assigned task list. In the ward management, the RS can use the Bubble sort to perform the sorting operation and recompilation of the FAS services. The selection of which sorting algorithm to use depends on the complexity of the workflow scheduling problem related to each individual organisation, for example, Selection sort, Quick sort or the sophisticated sorting algorithms such as Neural Net scheduling, Machine Learning algorithm like A* algorithm, Constrains-based Scheduling algorithm or Genetic algorithm.

### 3. 3 DPWFM Recompilation Feature Demonstrated

Web services reusability is the main feature in developing web applications (Inverardi and Tivoli 2007, Chukmol 2008). For that reason, the proposed DPWFM enables the recompilation of current FS to automatically create a new set of FAS. In the FS, sub-functions can be regrouped, separated or modified to become a new set of FS. In the FAS, the set of previously assigned FS can be reused in another similar job. For example, if it is the same assigned service, then the FS service can retrieve it from the web service repository. The supervisor can modify additional components and use it again. This provides flexibility and a greater control in organising work schedules in individual function unit.

The dynamic recompilation is applied to generate the RS workflow. The systematic steps of the RS can be described as follows:

1. The RS carries out the matching function between the FS and the FAS with the WPS.
2. The RS crosschecks information with the WPS in order to assign a matched FS to a designated nurse.
3. During the compilation, the RS evaluates the appropriate workflow schedule based on the assignments from the FAS.
4. The RS performs compilation of all selected web services from the FS into an individual web application for nurses.

As a result, the set of recompilation services are presented into individual web application of each nurse in a schedule that has been organised. Each nurse is presented with a unique sequence of tasks.

Figure 3.8 shows the four web service repositories. This example shows the function services of nursing care task in the FS and the FAS collects the assigned WPS with the chosen FS. The RS is applied to generate the agenda workflow after recompilation arrangement of the FS, FAS and WPS. In the example, ‘recording vital signs’ is a F1 function. It shows that F1 is a set of FS, which consists of F1a, F1b, F1c and F1d. The WPS contains instances of worker profile, which determines the qualification and level of authority of nurses. In the DPWFM framework, the FAS allows a supervisor to assign the FS to each nurse in the WPS and the framework provides supporting information and conditions for helping supervisor when assigning the FAS. The FS can be assigned separately (such as F1a and F2b) or assigned as a set of function (such as F1). For example, the function of F1: ‘recording vital signs’ consists of F1a: ‘record weight’, F1b: ‘record blood pressure’, F1c: ‘record respiration rate’ and F1d: ‘record temperature’. In cases where only selected sub-functions are required, then the supervisor can select only the required sub-functions. For example, selects F1a, F1b and F1d, but not F1c.

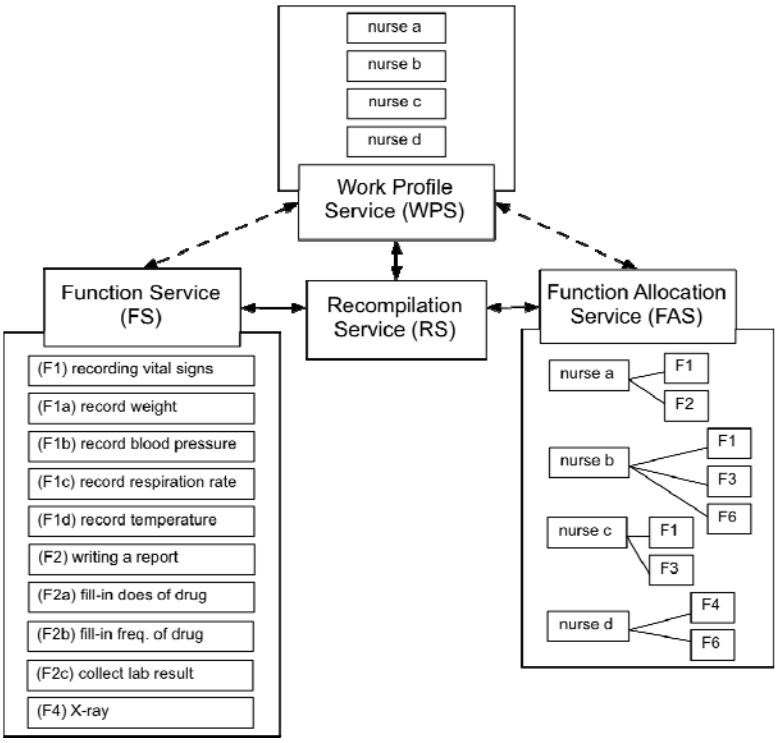



Figure 3.8 Four web service repositories of nursing care scenario

### 3. 4 Illustration of DPWFM Deployment

The following steps describe how the proposed DPWFM will be deployed. Firstly, the WPS and FS are initialised into the web service repositories. As described in

Table 3.1, the WPS stores information related to the profile of nurses such as job descriptions, work qualifications, job duties and responsibilities; whereas the FS stores task descriptions associated with each task, process and function. In most instances, the administrator often predefines information stored in the FS and WPS prior to the deployment of the DPWFM system. The third component, the FAS, deals with work assignment for a particular instance of time period. The supervisor assigns it at run-time such as during a shift. Then, the RS recompiles the work schedule by rearranging work assignments stored in the FAS and information gathered from the two components of the WPS and FS. Finally, a scheduled service of recompiling work assignment is produced as an individual web application to a user.

Figure 3.9 shows a sample of related function services that have been selected by the supervisor. Once the supervisor has selected the required FSs (in this example F1, F2, F4, F6, F8, F9, F10, F14, F15, F16, F17, F20, F21 and F24) into the FAS, then the proper sequence of FSs are reordered and regenerated to individual agenda workflow by the RS services. The RS performs the Bubble sorting operation to generate the agenda workflow services using time and priorities procedures depending on the ward rules.

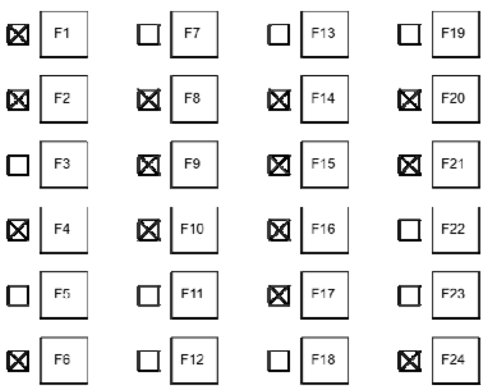



Figure 3.9 Function service repositories

Figure 3.10 shows the new sequential series of recompiled services for a nurse (in this case F1, F4, F2, F14, F8, F9, F10, F24, F20, F21, F16, F17, F6 and F15). The nurse has to complete the group of high-priority tasks or pre-requisite tasks (for example in Figure 3.10, this is the F1) and then performs the tasks that are of lower priorities until all tasks have been completed. The recompiled services show the sequence and order of tasks that need to be performed and to be completed. As explained previously, the schedule or sequence is a result of recompilation based on inputs from the FAS.

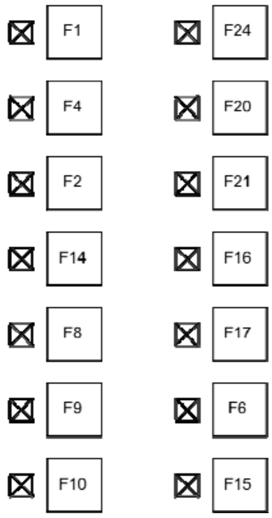



Figure 3.10 Recompile services

Figure 3.11 shows the elements in the function service repositories including the predefined interface components. This example describes five sets of FS repository that are F1, F2, F3, F4 and F5. Function F1: *recording vital signs* consists of sub-functions 1a: *record weight*, 1b: *record blood pressure*, 1c: *record respiration rate* and 1d: *record temperature*. Function F2: *writing a report* consists of sub-functions 2a: *submit dose of drug*, 2b: *submit frequency of drug* and 2c: *collect lab result*. Function F3: *medical record* consists of sub-functions 3a: *identify bed number* and 3b: *restriction setting*. Function F4: *X-ray* consists of a combination of F1: *recording vital signs*, sub-functions

4a: *describe lab diagnose-1*, 7c: *describe lab diagnose-2* and 6e: *describe diagnose-3*. Function F5: *prepare for surgery* consists of sub-functions 1a: *record weight* and 3b: *restriction setting*.

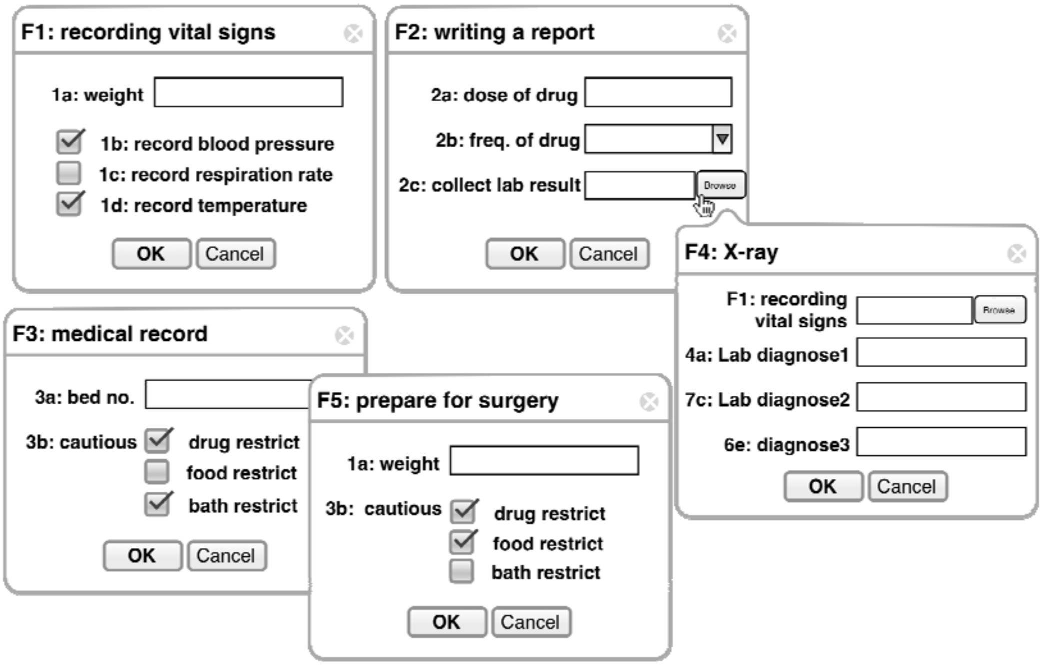



Figure 3.11 Elements in function service repositories

The example above shows that some sub-functions can be reused into another function set. For example, function F4: *X-ray* combines several sub-functions from other functions. Furthermore, some sub-functions can be linked, retrieved or accessed from another functions. For example activity 2c: *collect lab result* can be linked to the X-ray result of function F4.

Figure 3.12 shows the scheduled function and tasks that have been recompiled and presented in a systematic sequence to the user. The user can easily follow and do each task by following the proposed sequence. The supervisor can also customise each function, for example, in F1: *recording vital signs*, the supervisor can customise by selecting only some sub-functions such as 1a: *record weight*, 1b: *record blood pressure*, 1d: *record temperature*. Let us have another look at the example of the interface. The results show that after the task 1b, 1d, 4a, 7c and 6e have been completed. Then, the web services recompile and produce a new schedule of the result as shown in Figure 3.13.

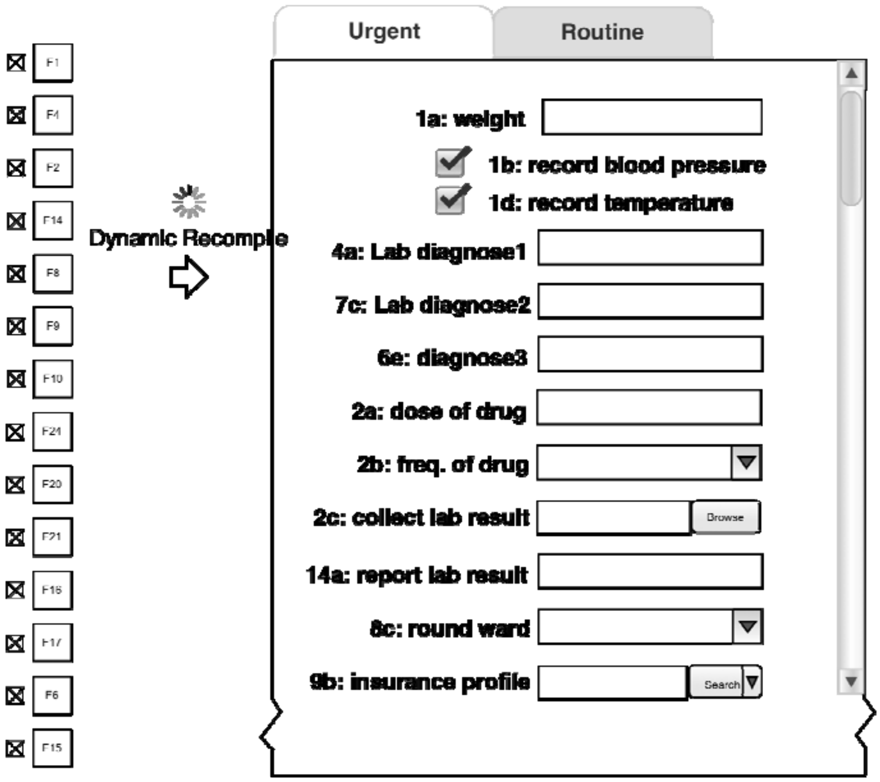



Figure 3.12 Recompilation services of the assigned function services

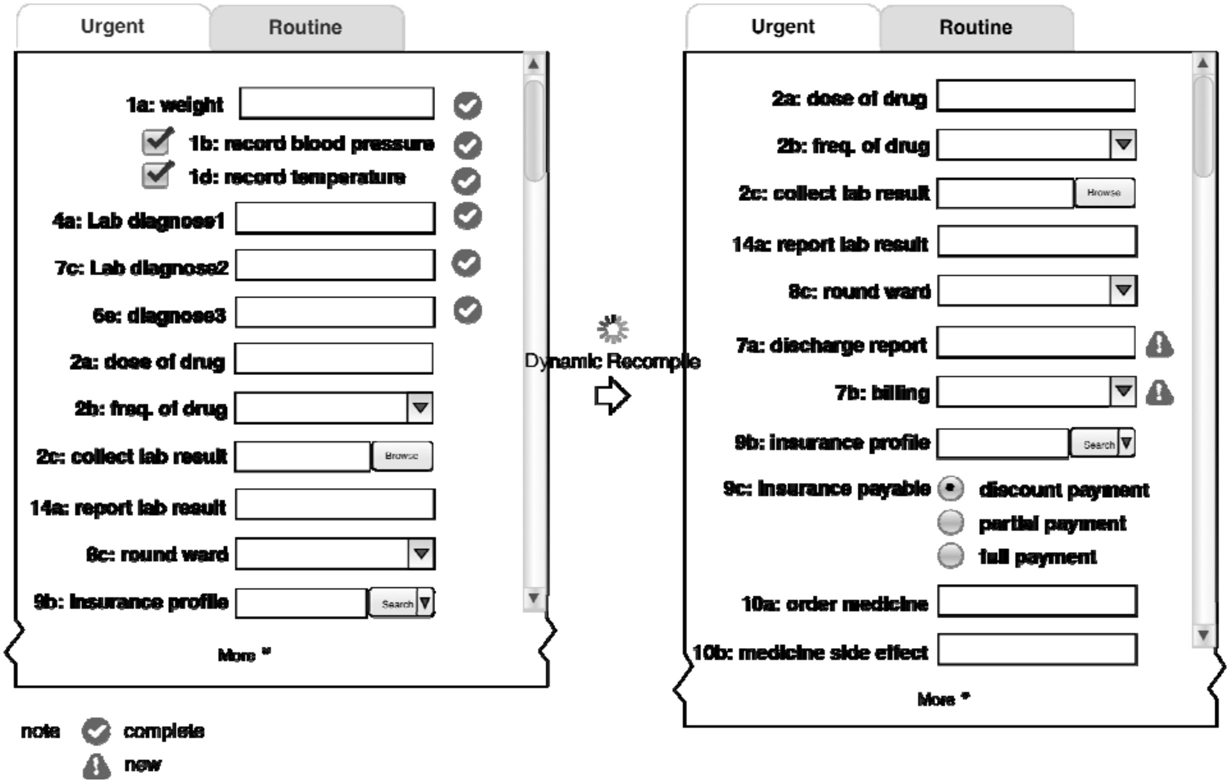



Figure 3.13 Dynamic recompilation after 1a, 1b, 1d, 4a, 7c and 6e have completed

In addition, the dynamic recompilation process is operated whenever the supervisor creates, modifies or reorganises new assignments of tasks. In summary, every time new assignments are inserted or changed, the workflows will be recompiled and a new schedule is produced.

### 3. 5 Three Layers of DPWFM

The proposed DPWFM model designs dynamically recompile workflow to generate web applications over the Intranet or Internet. The development focuses on using available open-source resources, so that it can communicate seamlessly with any operating platforms. To utilise better resources, organisation can continue to use the existing database servers. The DPWFM designs three layers of architecture: presentation, business logic and data layers.

Firstly, the presentation layer is the graphical user interface layer that provides facilities to customise and provide a suitable web application for each user level. Information on the web applications can be displayed in XML/HTML format. Figure 3.14 shows the presentation layer that provides functionality interfaces to nurses. Different roles have been provided with different function interfaces. For example, the supervisor has the authority to organise and maintain the FAS repository, while nurses can only retrieve, browse, update and enter information at the task levels.

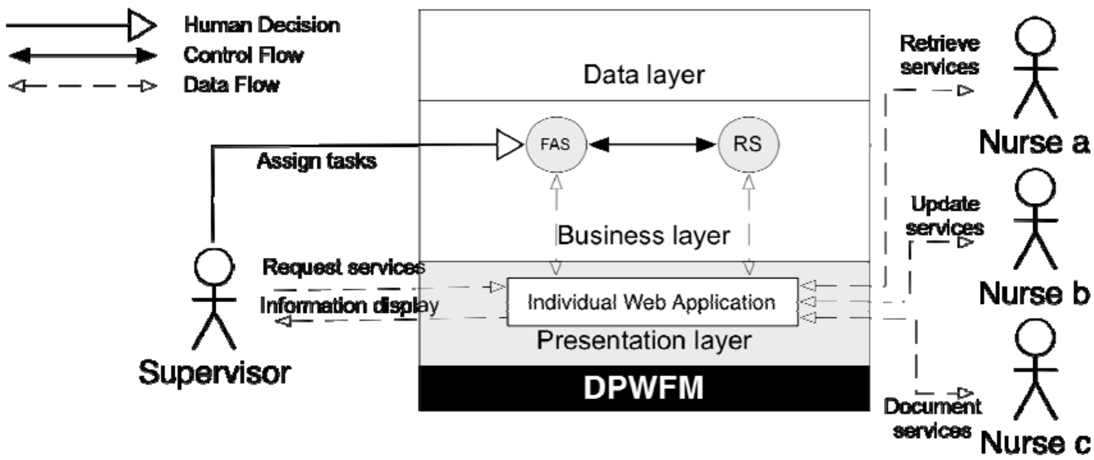



Figure 3.14 The presentation layer

Secondly, the business logic layer consists of business process that encompasses the WFM and is integrated using the SOA. Using the WFM with the SOA platform allows organisation ways to increase flexibility and speed up integration of applications for business advantage (Vaughan 2009). In this business logic layer, the SOA plays an important role to organise web service repository of the DPWFM. Figure 3.15 shows the business layer diagram. Based on business processes, suitable workflow schedule of everyday jobs are generated and presented as individual web application for each nurse.

The administrator initially predefines the WPS and the FS at the design state and then the supervisor can access the WPS and the FS in their day-to-day work. In this business layer, dynamic recompilation manager is applied to manipulate and control the FAS during execution time when supervisor requests the reallocation service as described previously.

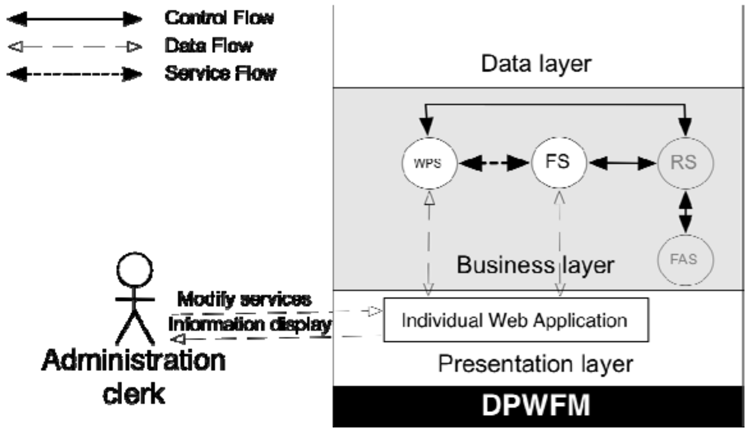



Figure 3.15 The business logic layer

Finally, the data layer consists of SQL based database management system (DBMS) and the XML parser. The DPWFM platform uses the standard XML to incorporate web services and to retrieve information from the existing database. The DPWFM data layer is designed to transform existing database into the XML format that suits the DPWFM server setting. Figure 3.16**Error! Reference source not found.Error! Reference source not found.** shows the data layer of the DPWFM server that connects with existing databases. The proposed framework performs information exchange with the DPWFM server in the form of XML and using the XML parser to translate data. This data layer is the responsibility of system administrator who setup web service repository. If there is any additional service required in the future, the administrator is also able to transform it into the DPWFM database.

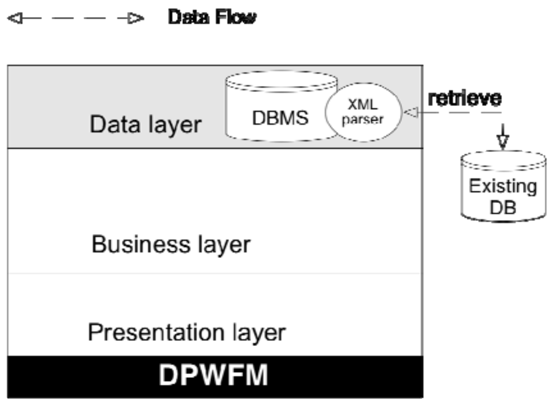



Figure 3.16 The data layer

### 3. 6 Chapter Summary

The DPWFM model for ward management has been described. The proposed platform enables dynamic recompilation and customisation features to support nurses to manage workflow in the ward. Individual ward has its own process, procedures and workflow culture. Day-to-day patient care does not remain static; in fact patient care is dynamically change depending on the patient's diagnosis and treatment plan. The dynamic recompilation feature in the proposed framework is important to reflect the dynamic nature of ward management.

The proposed DPWFM model has the capability to customise the ward workflow processes of the ward as the situations have been changed in the ward. There are three layers in the DPWFM architecture. The data layer translates existing database to XML format. The business logic layer organises business processes for ward workflow and the presentation layer provide the user-friendly interface that interacts with each user. The DPWFM provides four web service repositories to develop systematic reallocation of workflow for ward management. The WPS repository contains ProfileDB for manipulate the information related to the nurse description, his/her role and responsibility. The FS repository contains the tasks and subtasks information and their relationships or priority sequence into FunctionDB. It contains the business process, rule and procedure to explain how the tasks can be processed based on the organisation workflow. The FAS repository collects the assigned tasks from the supervisor and then the agenda workflow provides a recommended schedule or sequence has been recompiled by the RS. The result is the recompilation services for individual web application to help the nurses in managing ward workflow. In the next chapter, we will discuss the prototype design of the DPWFM model.

# CHAPTER 4: PROTOTYPE DESIGN

This chapter discusses the prototype design for the DPWFM model. The development design and technology used will be described. We will discuss the design based on three layers in the DPWFM model. The data layer relates to the database design, the presentation layer deals with the user interaction with the DPWFM system and the business logic layer will describe the dynamic recompilation design of the DPWFM model.

This chapter is organised as follows. Section 1 discusses the designing environment that suit for the DPWFM platform development. Section 2 describes the interaction interface designs of different users of the DPWFM system. Section 3 describes the physical design and architecture of the DPWFM model. That includes the design issues of the data, presentation and business logic layers. Section 4 concludes the chapter.

## 4. 1 Development Environment

The DPWFM model will be developed using web services technology. Web services can communicate using open protocol; they are self-contained and can be used by other applications. This framework utilises Internet as the backbone for delivering services. We use different web-based technologies that include PHP: Hypertext Preprocessor (PHP), ASP.NET, MySQL, XML/HTML, Apache Web Server and AJAX. The different markup and scripting languages are used to develop the web applications and the web services are stored in the web services repository. PHP5 is used to support component-based design, MySQL and XML. ASP.NET is the major part of the Microsoft .NET framework environment for building, deploying and running web applications and web services. MySQL database management system is for storing, manipulation and maintenance of data. To provide connectivity between AJAX and MySQL within the prototype, Apache version 2.0 is used to provide the engine to deliver the web interfaces and web services execution for the system. AJAX is used to support dynamic recompilation of web services.

Figure 4.1 shows the sequence diagram of the interaction between *UserInterface* using *AjaxEngine* to transmit data. The nurses send the request information through the function interface. That request is submitted to *UserInterface* and then the *AjaxEngine* (XMLHttpRequest object) will call the JavaScript functions for information operated in *BusinessLogicLayer* to retrieve the XML data from the *DataLayer*. After that the system will display the required information for the nurses.

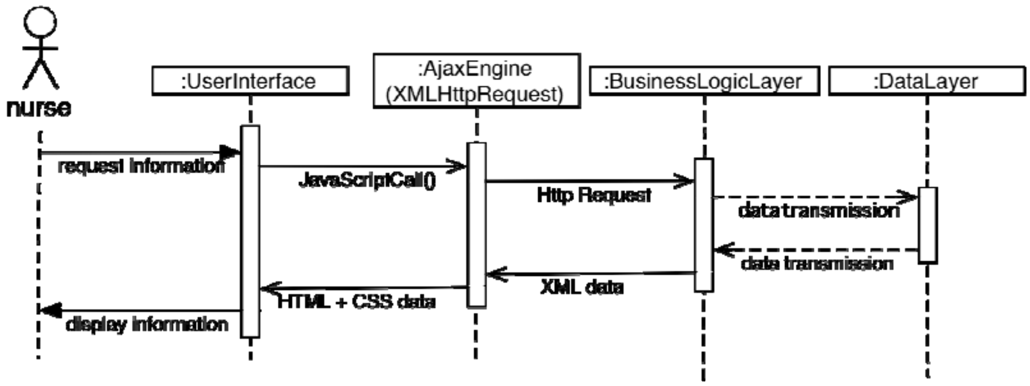



Figure 4.1 Using AJAX with the DPWFM

### 4. 2 Interaction Design Issues

There are three main function interfaces to support three types of interaction in the DPWFM platform. Figure 4.2 illustrates the relationships of these three functional interfaces with the three layers of the DPWFM platform. There are (1) administrator, (2) supervisor and (3) general nurse. These functional interfaces provide facilities to cater for different levels of usage, access authorisation, security clearance and different functional tasks that are required in the workflow management.

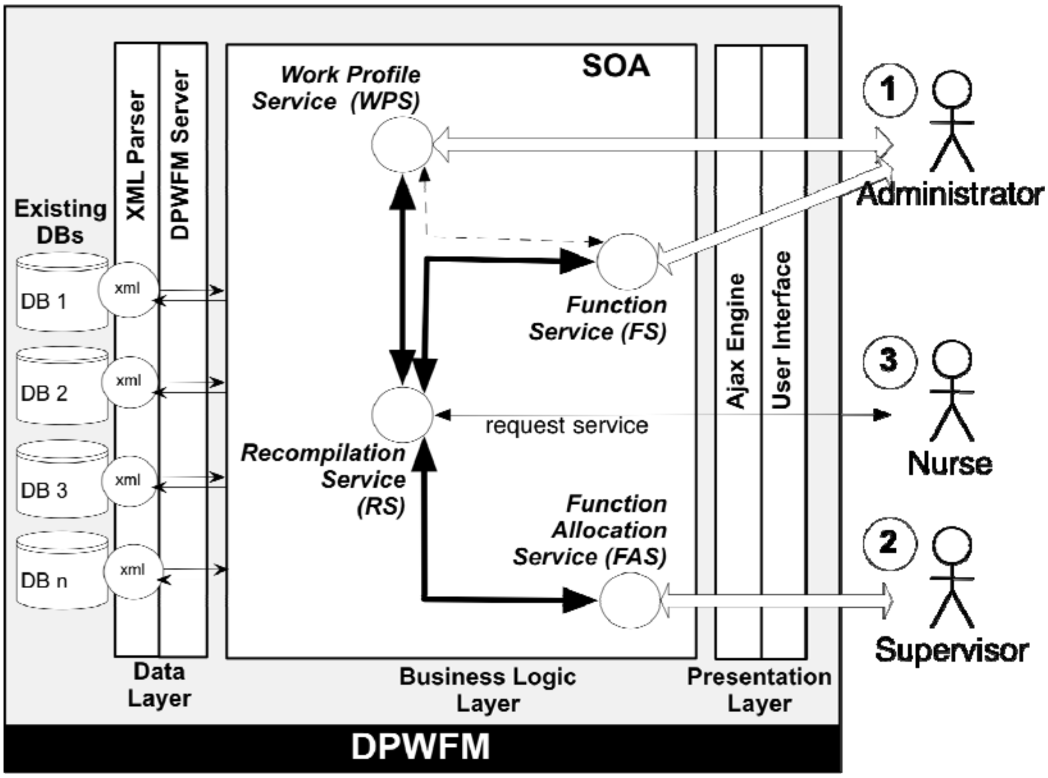



Figure 4.2 Three architecture layers of the DPWFM

The first type of interaction is performed by the administration staffs that are responsible for system installation, system configuration, resources assignments, general configurations, supports and maintenance. An administrator has authorised access to maintain information in the database, including assigning different user security access levels to the users of the system, maintenance of various functions and components of the web server, maintenance of various indexes and components in the web services repository, carry out routine system maintenance and restoration, perform system configurations, setting up user accounts, and maintain security of the system.

The proposed DPWFM platform allows the administrator to create custom security measure by segmenting which nurse can browse and interact with specific data from the database, accessing certain type of information, configuring different functional elements and advance filtering mechanism based on the roles of the nurses. The administrators can manipulate data from existing hospital database and manipulate any data from the database using SQL or XML script using the DPWFM server.

The second type of interaction is at the supervisor level which includes supervisor, head nurse, nurse unit manager and in-charge nurse. Their roles are to perform supervision and management of the flow of business functions, assign tasks, information control and accessibility of the system for each nurse. The proposed DPWFM allows supervisors to manage workflow processes as different tasks according to the requirements of the workflow specifications.

Finally, the third type interaction is at the nurse level. Nurses can sign in to the DPWFM system to obtain workflow information that contains the sequence of tasks assigned by the supervisor.

Figure 4.3 shows the different options of functional interfaces of administrator, supervisor, and user. *Administrator function interface* allows administers to create and maintain web service repository and also convert existing database to be used in the prototype. *Supervisor function interface* allows supervisors and managers to manage and organise the workflow in their departments. *Nurse function interface* allows nurses to access and update task completion information in the prototype.

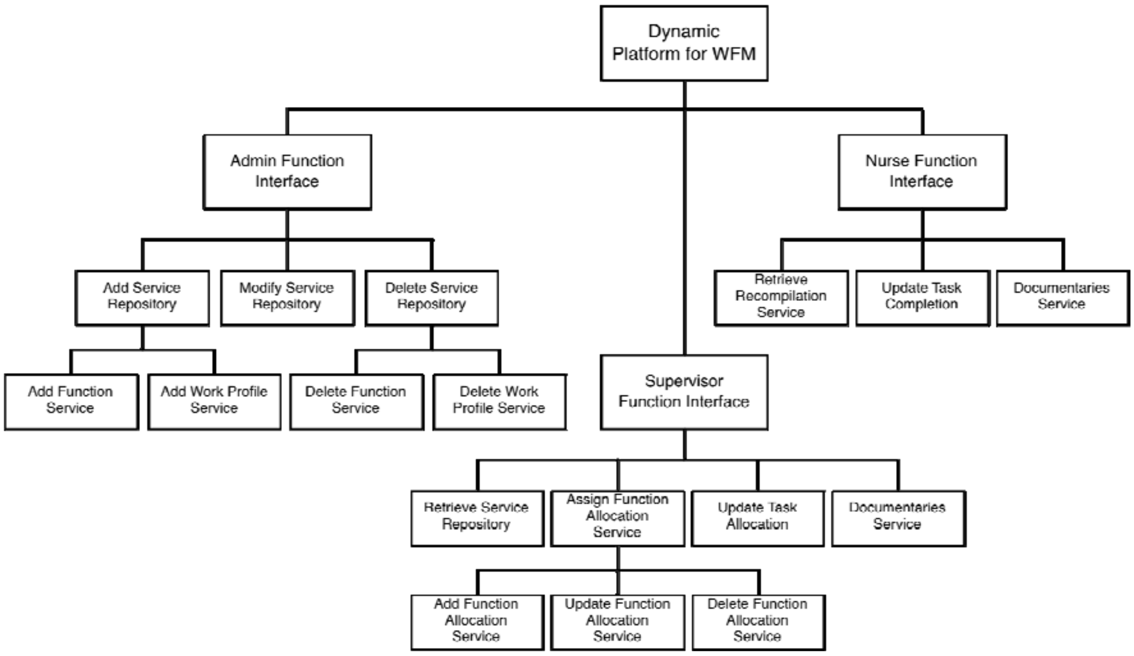



Figure 4.3 Functionalities of the prototype

## 4. 3 Physical Design and Architecture of the DPWFM

The DPWFM model is based on client-server architecture. The server will perform all the processing and execution of functions requested by the client, the client receives output from the server and results will be displayed on the client system. The DPWFM server architecture consists of three layers: Presentation, Business Logic and Data Layers. All layers store and retrieve information in form of XML and HTML scripts. The Presentation layer contains the web services that provide the interface to interact with the nurses who will view/retrieve the information in form of HTML data in their browser. Therefore, the nurses do not need to visit a server to use the system. They are only using their client applications by sending the request services via their web browsers. The information will compile and organise in the Business Logic layer by using data from the Data layer, after the information is recompiled and the priority function services rearranged into the workflow sequence for the nurses at the Business

Logic layer. As a result, the agenda workflow services were created. The Presentation layer represents the agenda workflow services in the form of their pre-setting logical components. The example of the pre-setting presentation logical components of individual function services is illustrated in Fig. 3.11, page 60. In conclusion, the nurses will use the system via their web browser (client) while the Presentation layer will compile/recompile the arranged function services in the form of the UI components for the nurses to use. Figure 4.4 shows the overview and the relationships of these three-layers in the prototype's architecture.

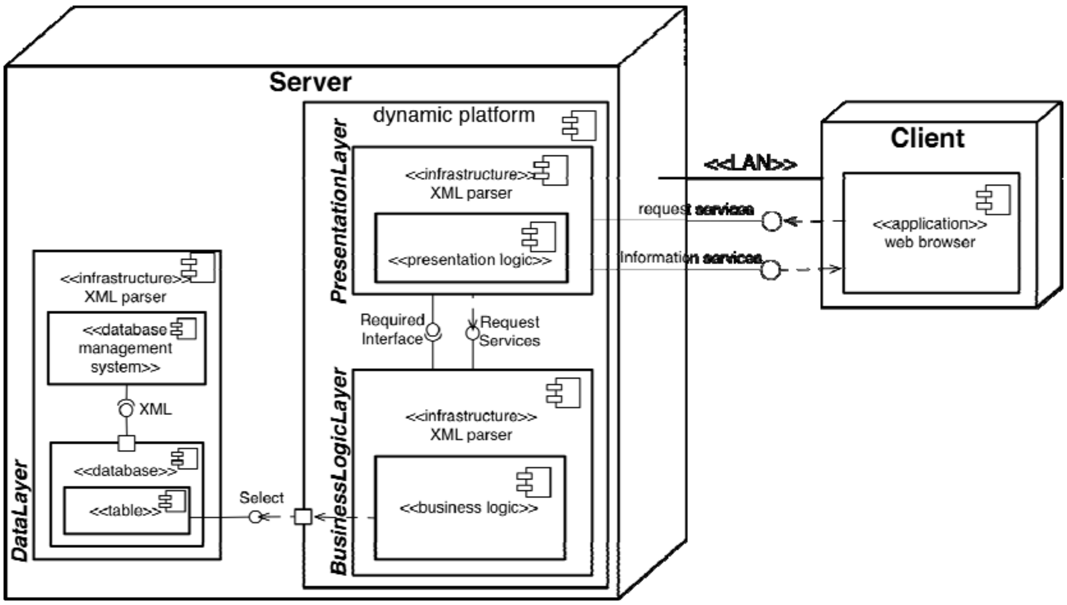



Figure 4.4 Overview of the prototype's architecture

### *4.3.1 The Presentation Layer*

The DPWFM system can be accessed using web browser. This is via a simple URL link http://localhost:port/DPWFM where the localhost is the name of hostname that houses the organisation's DPWFM server. The "port" refers to special communication channel number that will be used to interact with this system. This allows special server-based security to be implemented to protect the system from unauthorised access or malicious attack.

In order to verify the user's function interface, user needs to enter username and password for security verification. Various input devices can be used to allow more elaborate security protocol to integrate with the DPWFM system. Such devices can be security smart key card, bar code reader or fingerprint scanner. Figure 4.5 shows a simple sample of login page to access the DPWFM system and the example of a simple interface using fingerprint verification as the verification device.


Figure 4.5 Login web page of the prototype

The following is a web service that performs the task of the password verification using information stored in the profileDB.

```
//Example of password verification code
<input type='text' id='nurseName' name='nurseName'>
<input type='password' id='nursePass' name='nursePass'
onblur='validateNurseWPS()';>

function getXmlHttpRequestObject() {
 if (window.XMLHttpRequest) {
   return new XMLHttpRequest(); }
 else if(window.ActiveXObject) {
   return new ActiveXObject("Microsoft.XMLHTTP"); }
 else {alert("Your Browser is old!\nPlease upgrade"); } }

var xmlHttp =                       getXmlHttpRequestObject();
```

```
function validateNurseWPS() {
   var nurseName = escape(document.getElementById('nurseName').value);
   var nursePass = escape(document.getElementById('nursePass').value);
   var url = 'validate?nurse=' + nurseName + '&pass=' + nursePass;
   xmlHttp.open("GET", url);
   xmlHttp.onreadystatechange = Nursevalidate;
   xmlHttp.send(null);}

function Nursevalidate() {
 if (xmlHttp.readyState == 4) {
   if (xmlHttp.status == 20) {
      //request the access right of nurseName in WPS
      //if nurse password is wrong, set fail
      //if nurse password is correct, recompilation services for nurse
   } } }
```

The instruction issued by the client-side is captured and interpreted by the web server. If the instruction is to activate a web service stored on the web services repository, then that instruction will be passed to the DPWFM repository. The web service object will be retrieved and executed in the web server and the resulting outcome is constructed and posted on the nurse browser screen. The request generated from the user interaction can be in the form of signal (signal can include instructions and data in the form of XML scripts) and triggered by pressing a button, selection from a drop-down menu, choosing a radio button or checking on a checkbox.

In order to activate the FS to verify the password, the instructions can be written in the form of HTML script, PHP script, ASP.NET script or AJAX script. These scripts can perform the necessary interaction with the DPWFM on the client-side. Information or instructions that are needed to communicate with the DPWFM server will be transmitted using the standard Internet communication protocols that are in-built in the web browser software. When the DPWFM server receives that information, the web server will interpret it. If it is a normal web server instruction, then the DPWFM will carry out the instruction using the standard web server operation. If the instruction is to fetch a web service from the web service repository, then the script will be submitted to the XML parser to be stripped and be interpreted. The instruction will be carried out by the web service repository to fetch the right web service and return the web service function to the web server and be executed. If the web service function needs to access the database for information, the instruction will be issued by the web service function and the operation will then be carried out using the database server. In the example, the password verification instruction will be transmitted to the DPWFM web server using the client-side of the web browser.

Once the instruction is received by the web server, the information related to web service password verification function will be sent to the web service repository XML parser. The parser will strip down all the information needed to fetch the web service password verification function, the information needed by the function that was provided from the user (for instance the username and password provided on the client side) and any other information that are needed by the web service function for a successful execution of the function (for example the number of times a user can request to perform the password verification function). Once the web service function is successfully parsed, the web service repository will fetch the web service password verification function. When the web service function is successfully executed, the outcome will be constructed into HTML script again and be transmitted back to the user's web browser for display.

### *4.3.2 The Business Logic Layer*

This layer consists of the DPWFM repository server that performs primarily interpret the requests from the user interaction and deliver web services applications that were pre-designed, tested and stored into the web services database by an administrator according to the requirements of the ward management.

When an in-charge nurse assigns daily tasks to a nurse on the DPWFM web application. The in-charge nurse will select the tasks and the nurses that she requires to perform and assign to for the day. At the end of the tasks selection, the in-charge nurse will submit the tasks list to the DPWFM server. The server will then elicit information from the submission using the XML parser and fetch the corresponding data in the ProfileDB and the FunctionDB from the database and also fetch the web services from the DPWFM repository.

The first web service that will be fetched from the web service repository is the FAS repository. The FAS performs the job of a controller for instructing or directing the web server in performing various operations/programs in completing the task at hand. In this case the recompilation of the tasks list submitted by the in-charge nurse and return the resulting output in a planned schedule tasks list according to the priority, precedence and job descriptions of each task. The FAS starts by fetching the second web service, the WPS repository, where information from the tasks list will be stripped down using the XML parser into individual nurse information stored in various data fields or variables. The WPS then uses the information from the tasks list to access and fetch information in the ProfileDB from the database. After that the FAS fetch the third web service, the FS repository to determine priority and requirements of each task that have been assigned and fetch information in the FunctionDB. This information will be stored in various data fields and variables to perform the necessary operations in recompilation. The FAS will perform the following steps:

1. Perform consistency check of the WPS with the FS to verify the role responsibility

and task level. For instance, a nurse who is working in the night shift cannot work on a task that is only being done during the day shift. A junior nurse should not be allocated any task that can only be done by the qualified RN. If there is an error, the recompilation operation terminates and error statements will be generated for the in-charge nurse to rectify the situation and reassign tasks.

2. If there is no error, then the FAS will initiate and fetch the FS to determine the priority sequence from the FunctionDB. After that, the FAS will fetch the RS to reconstruct the assigned information and performs the operation of comparing, sorting and prioritising the tasks list as submitted by the in-charge nurse. The operation can be carried out using sorting algorithms such as Bubble sort.

3. The RS will read each of the entries in the sorted tasks list and construct the information according to the requirements of the tasks by fetching information from data fields in the ProfileDB and the FunctionDB in the DPWFM database. It constructs the relevant AJAX script, PHP scripts, ASP.NET script or HTML scripts by fetching relevant script templates stored in the FunctionDB and then inserts the right information from the data fields and variables that were processed from the FAS.

4. The outcome is an agenda workflow schedule that will be compiled into a HTML scripted file. The information recorded in the scripted HTML file will also be stored in the data fields of the agenda workflow database.

5. The above processes will be repeated whenever the in-charge nurse changes any information related to the assignment of tasks to her subordinates.

A significant feature of the proposed DPWFM is the design and use of dynamic recompilation, which allows web services to recompile the changing workflow assignments from the nurses with respect to the ward workflow requirements. This approach provides a well-organised workflow assignment services to nurses. The recompilation operation will be activated when the supervisor submits changes in workflow assignment request and it will trigger the FAS to execute the recompilation operation request from the web server. Figure 4.6 shows the state diagram of dynamic recompilation operations.

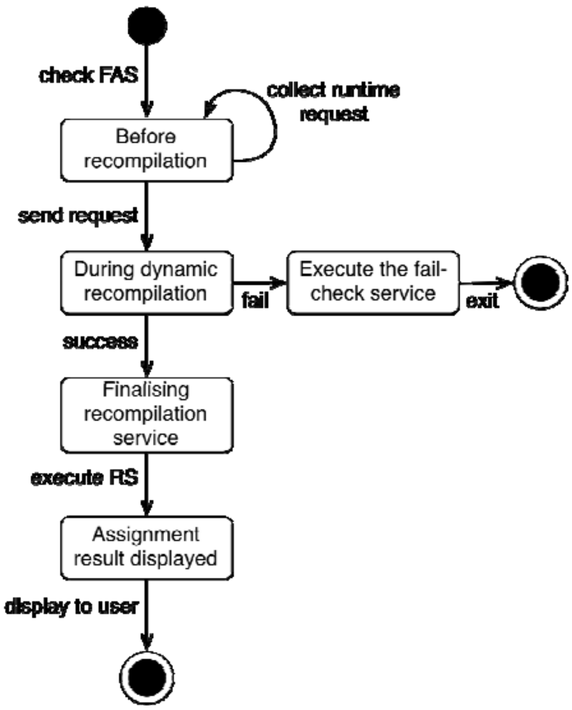



Figure 4.6 Top level state diagram of dynamic recompilation system

The dynamic recompilation operations work according to different states of the recompilation operation as described below:

**Before recompilation:** Check the predefine setting of FAS and XMLHttpRequest handles for listening the service allocation requests from the supervisor (see Figure 4.7).

1. Verify the user authorisation with the WPS repository in “verify authorisation with

WPS” state. At the same time, check the FAS if verification is valid, the state will be changed to “load web application”. It provides the user function interface for enquiry request.

2. Change the state into “runtime request service enquiry”. In this state, supervisors are allowed to set the FS and allocate tasks to subordinate in the FAS. At the same time, the state “loads Ajax engine” has been activated to continue collecting request at execute time.
3. “Access FS” and “access FAS” will be activated after selected FS for the subordinate user into FAS.
4. Then change the state into “create FAS” when there is a new function assigned, or “modify FAS” when loading the current set of FAS repository.
5. The new settings of FAS are saved into the FAS repository. Then change the state into “during recompilation” and exit.

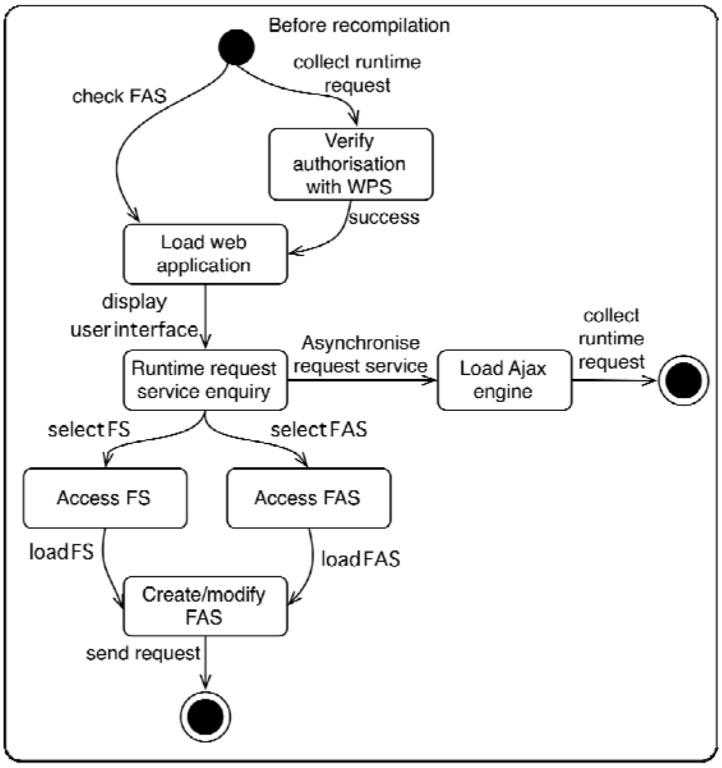



Figure 4.7 Before recompilation sub module

The preparation of information processed before recompilation is described in the following pseudo codes.

```
// Note: Italic style refers to a database model in Figure 3.7
State: “verify authorisation with WPS”

//Verify the user authorisation with the WPS repository
If (supervisor) Then
      {allow to access and maintain the services in FS and FAS;
      State changes to “load web application;}
      Else ø

//Load individual web application of the authorised supervisor
State: “load web application”
Provide individual user interface for enquiry request to nurse;
State changes to “runtime request service enquiry”;

//XMLHttpRequest listen to the supervisor response and run
State: “runtime request service enquiry”
While (supervisor)
      {request supervisor requirement;
      If (set or access the FS, TaskDescription) Then
      {retrieve service ‘modified’, ‘inserted’, ‘cancelled’ and
      ‘deleted’};
      If (set the FAS) Then
      {allocate tasks into FAS for each nurse;
      State changes to “create FAS” or “modify FAS”;}
      }
```

**State:** “create FAS”
**While** (supervisor)
      {**If** (a new task is assigned) **Then**
            {nurse responsibility of the task is required and record
            the task in FAS, *TaskAssignRecord*;
            send request to the RS, *AgendaWorkflow*;};
      State changes to “service manipulate”;
      }

**State:** “modify FAS”
**While** (supervisor)
      {**If** (a new task is modified/reassigned) **Then**
            {the FAS set the modified requests to retrieve web service
            database from *TaskAssignRecord*;
            send request to the RS, *AgendaWorkflow*;};
      State changes to “service manipulate”;
      }

**During recompilation:** Ask the FAS for service request and classify the recompilation controller after retrieving services (see Figure 4.8).

1. If there is a service modification, change the state into “service modified”, send the executed result and exit.
2. If there is a new service assigned, change the state into “service inserted”, send the executed result and exit.
3. If there is a service cancellation, change the state into “service cancelled”, executed the original FAS and exit.
4. If there is a service deletion, change the state into “service deleted”, send the executed result and exit.

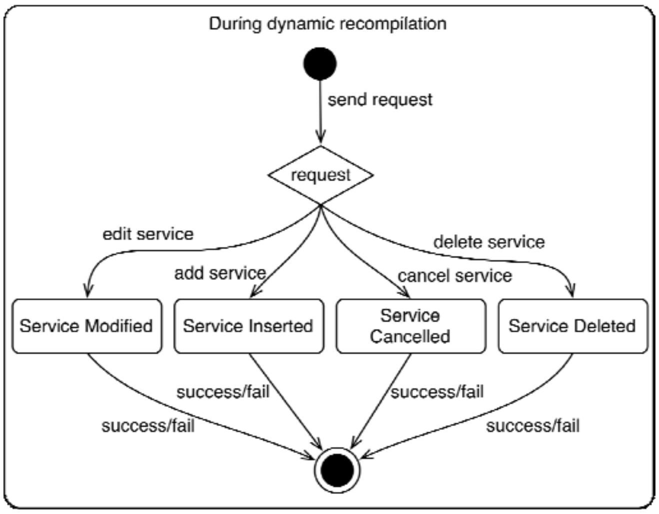



Figure 4.8 During dynamic recompilation sub module

The following pseudo codes are described the services classification and requests.

```
// Note: Italic style refers to a database model in Figure 3.7
State: “service manipulate”

//supervisor requests to modify services and then system recompiles
Service Modified:
If (a patient information is modified) Then
      {the FAS retrieves web service database from PatientProfile;
```

```
      modifies the patient information and store in PatientProfile;}
If (a new task is modified/reassigned) Then
      {the FAS set the modified requests to retrieve web service
      database from TaskAssignRecord;
      recompile and store the modified service into the RS,
      AgendaWorkflow;}

//supervisor requests to insert services and then system recompile
Service Inserted:
If (a new patient is admission) Then
      {the FAS adds inserted requests of the patient to web service
      database of PatientProfile;
      status is set to ‘admission’ in PatientProfile;
      nurse(s) is/are assigned to the new patient in TaskAssignRecord;
      new task activities are set for the patient in TaskAssignRecord;
      recompile and store the new task activities into the RS,
      AgendaWorkflow;}
If (a new task is assigned) Then
      {nurse responsibility of the task is required and record the
      task in FAS, TaskAssignRecord;
      recompile and store the new task activities into the RS,
      AgendaWorkflow;}
If (a new nurse is employed) Then
      {record the new nurse in WPS, ProfileDB, NurseInfo;
      new NurseInfo and RoleDescription are enquired;}

//supervisor requests to cancel services and then system recompile
Service Cancelled:
If (a patient discharges is cancelled) Then
      {rollback task activities and reset for the patient in
      TaskAssignRecord;
      recompile and store the new task activities into the RS,
      AgendaWorkflow;}
If (a task is cancelled) Then
      {rollback the task activities in FAS, TaskAssignRecord;
      recompile and store the new task activities into the RS,
      AgendaWorkflow;}

//supervisor requests to delete services and then system recompile
Service Deleted:
If (a patient is deleted) Then
      {the FAS retrieves the patient information from PatientProfile;
      status is set to ‘discharge’ in PatientProfile;}
If (a task is deleted) Then
      {retrieves the FAS, TaskAssignRecord;
      modify the assignment of nurse related to the task;
      recompile and store modified service into the RS,
      AgendaWorkflow;}
```

**Finalising recompilation service:** Execute the success recompilation RS (see Figure 4.9). Rearrange and load the previous FAS subroutine with the FS predecessor setting and exit.

1. If service requests successfully recompiled, change the state to “verify request service”. Then identify the request and execute it.

2. If there is a request for editing service, the state is changed to “modify RS”, execute the modification of the selected FAS and update in the RS repository.

3. If there is a request for adding service, the state is changed to “insert RS”, determine the predecessor sequence of a new service in the RS repository.

4. If there is a request for deleting, the state is changed to “delete RS”, the selected FAS has been deleted.

5. If there is no request, change the state into “load current RS”, retrieve the current RS. If the RS list is empty, create the new RS in “create RS” state.

6. After access RS, change the state into “load FS predecessor arrangement”.

7. Reallocate FAS setting and change the state into “define RS predecessor sequence”, execute RS and exit.

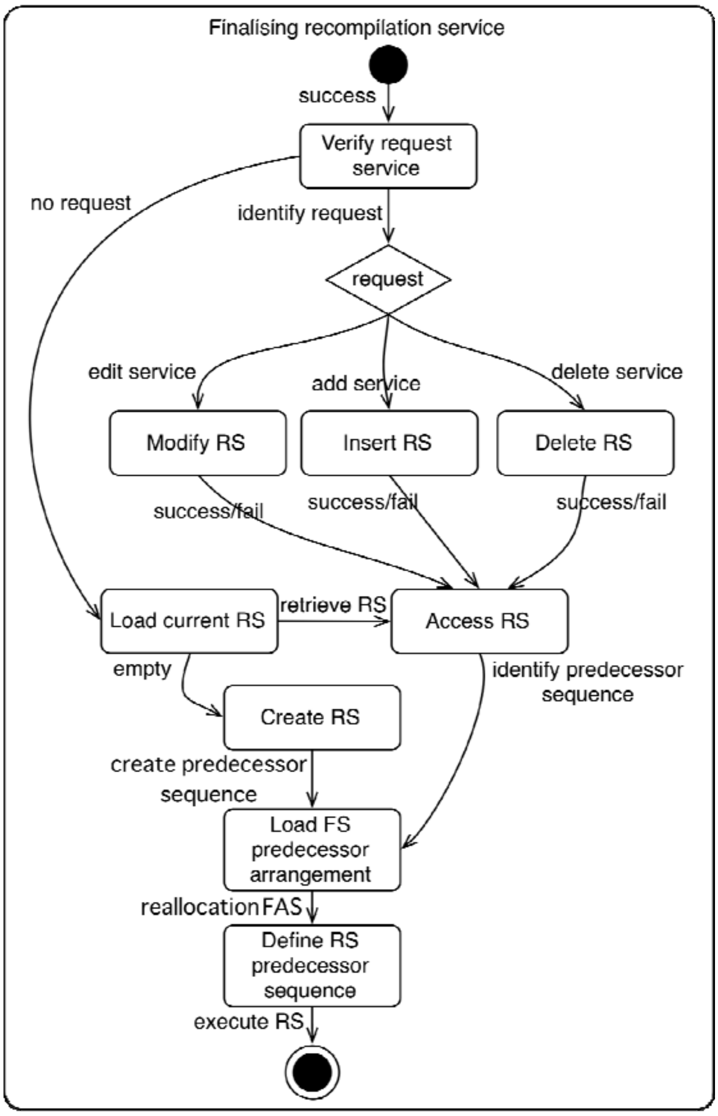



Figure 4.9 Finalising recompilation service sub module

The RS will read each of the entries in the sorted workflow tasks list, build information according to the requirements of the task by retrieving information from data fields in the ProfileDB and FunctionDB, build the relevant AJAX scripts, PHP scripts, ASP.NET script or HTML scripts by fetching the relevant script templates stored in the FunctionDB and inserting the right information from the data fields and variables that were processed from the FAS. This information will compile into a HTML scripted file and this file will display for individual nurse and all information of the workflow tasks list will be presented on the screen in the recompiled sorted format.

The following pseudo codes describe the manipulation of the RS when the supervisor reassigns the FAS.

```
// Note: Italic style refers to a database model in Figure 3.7
State: "dynamic recompilation"

//create AgendaWorkflow in the RS
While (supervisor request the RS services)
   {If (FAS service modify) Then
      {access the FS, FunctionDB
      retrieve TaskDescription;
      retrieve TaskAssignRecord;
      retrieve PatientProfiles;
      retrieve NurseInfo;
      access the WPS, ProfileDB;
      query records in TaskDescription that match to NurseInfo and
      RoleResponsibility;

      If (TaskDescription has SubTask) Then
         retrieve SubTask and TaskLevel that match with
      TaskAssignRecord;

      //load FS predecessor arrangement
      query predecessor ordering in TaskDescription;
      query the processing times of the tasks and starting time of the
      tasks and ending time of the tasks in TaskAssignRecord and
      TaskDescription;
      sequence and rearrange the tasks in TaskAssignRecord by time
      constrains of tasks in TaskDescription;
      rearrange the tasks in TaskAssignRecord by predecessor ordering
      in FS, PrioritySequence;
      set the higher predecessor at the top list;
      recompile AgendaWorkflow of PatientProfile with the new
      arrangement of TaskAssignRecord;

      //request confirmation of new assignment setting from supervisor
      If (confirms for FAS service modify) Then
         {recompile the TaskAssignRecord;
         State changes to "display web results";}
         Else cancel and exit;
      }
   }
```

**Execute the fail-check service:** If for any reason that the execution of the recompilation operation fails, the FS will generate an error statement to inform the nurse of an unexpected error occurred during recompilation and the user may need to resubmit the request again.

**Assignment result displayed:** Send the executed RS to generate the GUI components to display to individual nurse (see Figure 4.10).

1. The executed RS was sent into the state "load UI web service" for selecting GUI that used to represent the selected FS. At the same time, the state "rearrange RS" is activated, provide the sequence upon the predecessor setting and execute.
2. Change the state into "scheduled service displayed", generate the user function interface of using proper GUI of FS and exit.

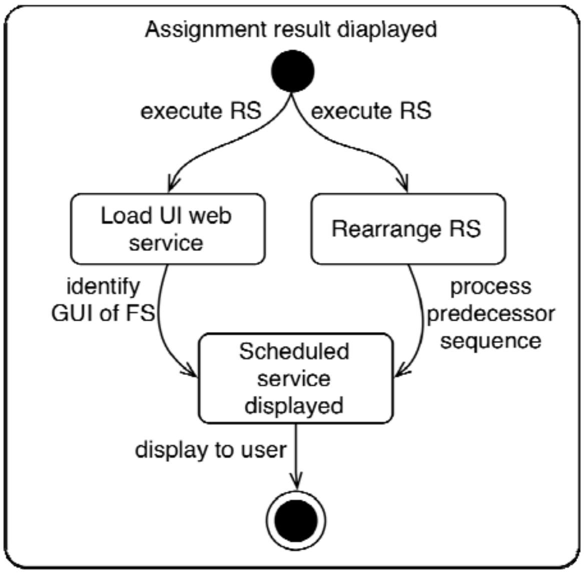



Figure 4.10 Assignment result displayed sub module

```
// Note: Italic style refers to a database model in Figure 3.7
State: "display web results"
If (supervisor confirms for FAS service modify) Then
   {retrieve the level of access of nurse in RoleDescription;
   verify NurseInfo and RoleResponsibility with the matching records
   between RoleDescription and TaskDescription;
   retrieve the recompiled TaskAssignRecord;
   create AgendaWorkflow that match the recompiled TaskAssignRecord
   with both NurseInfo and RoleDescription;

   //create an individual web application for each nurse
   retrieve the recompile function interface;
   create web service components of each task according to the
   recompiled AgendaWorkflow for each nurse in NurseInfo;
   }
```

### 4.3.3 The Data Layer

The database management system (DBMS) is for managing raw data, tables, functional services with embedded procedures and script-objects. The XML parser translates any XML instructions into SQL for the interaction needed with the DBMS. Examples of instructions include creation, manipulation, storage and maintenance of data tables.

PHP is a very versatile web programming language with good capabilities, however in this thesis, it was only good to build the system engine and structure of the web services. The incorporation of XML is for portability. The use of MySQL is for ease of information retrieval and management of the DBMS. The incorporation of Ajax is to allow interaction and retrieval of data from the server asynchronously in the background without interfering with the display and behaviour of the web services and Ajax allows dynamic interfaces to display on web pages.

The information that is obtained from the XML parser can also be instructions for the DBMS to carry out database operations include retrieving and storing data to execute the stored procedures and objects. Most of the database operational functions are in-built to the DBMS. However, the structures of the data tables will need to be designed based on the required functions of the ward workflow.

There are different tables in the DBMS. Figure 4.11 shows some of the sample tables. The maintenance of the database can be carried out using the tools provided by the DBMS.

| Table Name | Type | Row Format | Row | Data Length |
|---|---|---|---|---|
| BedInfo | MyISAM | Dynamic | 24 | 1.29 k |
| DynamicTask | MyISAM | Dynamic | 150 | 4.89 k |
| FunctionDescription | MyISAM | Dynamic | 49 | 1.73 k |
| Login | MyISAM | Dynamic | 13 | 312 |
| NurseInfo | MyISAM | Dynamic | 39 | 2.12 k |
| NurseShiftSchd | MyISAM | Dynamic | 600 | 33.45 k |
| PatientProfile | MyISAM | Dynamic | 18 | 1.52 k |
| RoleDescription | MyISAM | Dynamic | 7 | 164 |
| RoleResponsibility | MyISAM | Dynamic | 13 | 380 |
| SubDescription | MyISAM | Dynamic | 7 | 280 |
| SubTask | MyISAM | Dynamic | 176 | 4.72 k |
| TaskAssignRecord | MyISAM | Dynamic | 23 | 772 |
| WardDescription | MyISAM | Dynamic | 3 | 220 |
| WorkerTimeTable | MyISAM | Dynamic | 3 | 60 |

Figure 4.11 Examples of tables

On the front-end of the DPWFM system, there are two sets of data tables interacting the database within MySQL. The first set relates to the profile of each user types called ProfileDB. The second set of tables relates to all functional specifications of the ward workflows and all application scripts for the web services that will be used by the web server. This set of tables is called FunctionDB. Figure 4.12 shows the ProfileDB stores information such as nurse profiles.

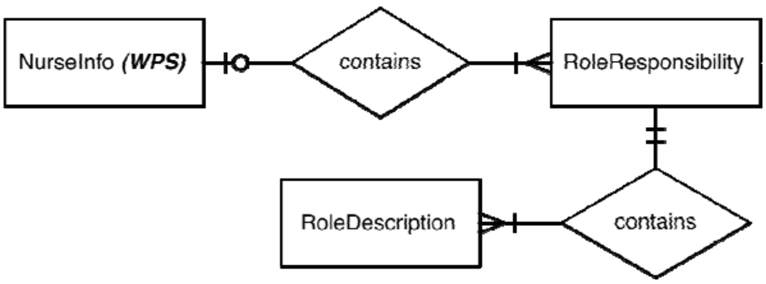



Figure 4.12 ProfileDB

ProfileDB includes nurse information, role responsibility and role description. The following shows the data elements of these nurse profiles.

Nurse Information:

```
CREATE TABLE  'ward'.'NurseInfo' (
  'WorkerID' varchar(5) NOT NULL DEFAULT '',
  'Name' varchar(20) DEFAULT NULL,
  'Lastname' varchar(20) DEFAULT NULL,
  'StartWorkDate' varchar(20) DEFAULT NULL,
  'Speciality' varchar(50) DEFAULT NULL,
  'Position' varchar(50) DEFAULT NULL,
  'SecurityLevel' varchar(3) DEFAULT NULL,
  'Status' varchar(15) DEFAULT NULL,
  PRIMARY KEY ('WorkerID'),
  UNIQUE KEY 'id' ('WorkerID')
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Role Responsibility:

```
CREATE TABLE  'ward'.'RoleResponsibility' (
  'WorkerID' varchar(5) NOT NULL DEFAULT '',
  'RoleID1' varchar(5) NOT NULL,
  'RoleID2' varchar(5) DEFAULT NULL,
  'RoleID3' varchar(5) DEFAULT NULL,
  'RoleID4' varchar(5) DEFAULT NULL,
  'RoleID5' varchar(5) DEFAULT NULL,
  PRIMARY KEY ('WorkerID'),
  UNIQUE KEY 'id' ('WorkerID')
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Role Description:

```
CREATE TABLE  'ward'.'RoleDescription' (
  'RoleID' varchar(5) NOT NULL DEFAULT '',
  'Description' varchar(255) DEFAULT NULL,
  PRIMARY KEY ('RoleID')
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

The FunctionDB stores information related to functional specifications such as task description (see Figure 4.13).

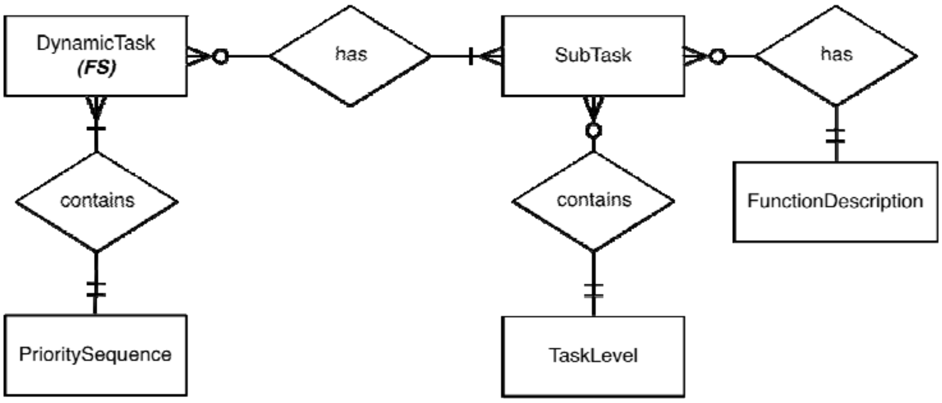



Figure 4.13 FunctionDB

FuncitonDB includes tables of task description, subtask, task level and priority sequence. The following shows the data elements of task information.

Dynamic Task:

```
DROP TABLE IF EXISTS 'ward'.'DynamicTask';
CREATE TABLE  'ward'.'DynamicTask' (
  'TaskID' varchar(6) NOT NULL,
  'PatientID' varchar(5) DEFAULT NULL,
  'SuperID' varchar(5) DEFAULT NULL,
  'NurseID' varchar(5) DEFAULT NULL,
  'Remark' varchar(255) DEFAULT NULL,
  'Status' varchar(15) DEFAULT NULL,
  PRIMARY KEY ('TaskID')
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Subtask:

```
CREATE TABLE  'ward'.'SubTask' (
  'TaskID' varchar(6) NOT NULL DEFAULT '',
  'Time' varchar(5) DEFAULT NULL,
  'SubID' varchar(6) NOT NULL,
  'FunctionID' varchar(8) NOT NULL,
  'Description' varchar(255) DEFAULT NULL,
  'LevelID' varchar(2) NOT NULL,
  PRIMARY KEY ('SubID')
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Function Description:

```
CREATE TABLE  'ward'.'FunctionDescription' (
  'FunctionID' varchar(8) NOT NULL,
  'FunctionName' varchar(50) DEFAULT NULL,
  'Description' varchar(255) DEFAULT NULL,
  'Predec' varchar(6) DEFAULT NULL,
  'Logical' varchar(6) DEFAULT NULL,
  'TaskLevel' varchar(2) DEFAULT NULL,
  'Note' varchar(255) DEFAULT NULL,
  'PrioritySeq' varchar(6) NOT NULL,
  PRIMARY KEY ('TaskID')
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Task Level:

```
CREATE TABLE  'ward'.'TaskLevel' (
  'LevelID' varchar(2) NOT NULL,
  'WorkerName' varchar(50) DEFAULT NULL,
  PRIMARY KEY ('LevelID')
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Priority Sequence:

```
CREATE TABLE  'ward'.'PrioritySequence' (
  'TaskID' varchar(6) NOT NULL,
  'PrioritySeq' varchar(6) NOT NULL,
  'ServiceType' varchar(50) NOT NULL,
  'ServiceSource' varchar(50) DEFAULT NULL,
  'ServiceDesc' varchar(50) DEFAULT NULL,
  'ProrityTime' varchar(25) DEFAULT NULL,
  'ProrityTask' varchar(25) DEFAULT NULL,
  'ProrityNurse' varchar(25) DEFAULT NULL,
  'ProrityConstrain' varchar(50) DEFAULT NULL,
  'Comment' varchar(255) DEFAULT NULL,
  PRIMARY KEY ('TaskID','PrioritySeq')
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Other related tables include task-assigned record, patient profile, nurse shift schedule, ward description and login tables. The following are the data elements of these tables.

Task-assigned Record:

```
CREATE TABLE  'ward'.'TaskAssignRecord' (
  'TAssignRecID' varchar(3) NOT NULL,
  'TaskID' varchar(6) NOT NULL,
  'ShiftID' varchar(6) DEFAULT NULL,
  'WorkerID' varchar(5) NOT NULL,
  'PatientID' varchar(6) DEFAULT NULL,
  'Comment' varchar(50) DEFAULT NULL,
  PRIMARY KEY ('TaskID', 'TAssignRecID')
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Patient Profile:

```
CREATE TABLE  'ward'.'PatientProfile' (
  'PatientID' varchar(5) NOT NULL,
  'Name' varchar(20) DEFAULT NULL,
  'Lastname' varchar(20) DEFAULT NULL,
  'CaseDesc' varchar(50) DEFAULT NULL,
  'EnrollDate' varchar(10) DEFAULT NULL,
  PRIMARY KEY ('PatientID')
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Nurse Shift Schedule:

```
CREATE TABLE  'ward'.'NurseShiftSchd' (
  'WorkerID' varchar(5) DEFAULT NULL,
  'ScheduleID' varchar(5) DEFAULT NULL,
  'theDate' varchar(10) DEFAULT NULL,
  'Weekday' varchar(10) DEFAULT NULL,
  'WhichShift' varchar(2) DEFAULT NULL,
  'StartTime' varchar(5) DEFAULT NULL,
  'EndTime' varchar(5) DEFAULT NULL,
  'WhatRole' varchar(5) DEFAULT NULL,
  'WardID' varchar(5) DEFAULT NULL,
  'TAssignRecID' varchar(3) DEFAULT NULL,
  'Comments' varchar(50) DEFAULT NULL,
  PRIMARY KEY ('WorkerID')
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Ward Description:

```
CREATE TABLE  'ward'.'WardDescription' (
  'WardID' varchar(5) DEFAULT NULL,
  'Description' varchar(255) DEFAULT NULL,
  PRIMARY KEY ('WardID')
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Login:

```
CREATE TABLE  'ward'.'Login' (
  'WorkerID' varchar(6) NOT NULL DEFAULT '',
  'UserName' varchar(20) DEFAULT NULL,
  'Pass' varchar(20) DEFAULT NULL,
  'Description' varchar(255) DEFAULT NULL,
  'SercurityLevel' varchar(6) DEFAULT NULL,
  'PermissionLevel' varchar(6) DEFAULT NULL,
  'usertype' varchar(2) DEFAULT NULL,
  PRIMARY KEY ('WorkerID'),
  UNIQUE KEY 'id' ('WorkerID')
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

## 4. 4 Chapter Summary

This chapter describes the prototype design of the DPWFM. The user function interfaces interacting with the DPWFM system were also described. There are three architecture layers of the DPWFM model. First is the data layer that provides the database design model of the proposed DPWFM. This layer consists of the DBMS and XML parser for communicating and translating data between the existing database server and the DPWFM server. The DPWFM data was designed and divided into two sets that are ProfileDB and FunctionDB databases. Second is the presentation layer. The interface with the users and the interactive devices has been discussed. The AJAX technology helps to generate the simple interactive and user-friendly interface for the nurses. Third is the business logic layer. This layer provides the mechanisms of the dynamic recompilation that operate in the DPWFM service between the four DPWFM repositories which are the WPS, FS, FAS and RS repositories. We will illustrate the walkthrough of DPWFM using a nursing ward scenario in the following chapter.

# CHAPTER 5: ILLUSTRATIVE SCENARIO OF USING DPWFM

In this chapter, we will describe how the DPWFM model works using a nursing care workflow scenario. We will first present the scenario in a paper-based ward environment. Then, the scenario will be described when the DPWFM model is applied.

The chapter is organised as follows. Section 1 describes introduction of nursing ward management. Section 2 describes the nursing care scenario with paper-based workflow. Section 3 illustrates the nursing workflow using the DPWFM platform. Section 4 gives the chapter summary.

## 5. 1 Nursing Ward Management

Each ward is managed by a head nurse. The head nurse allocates nurses for three shifts per day. Usually this is planned up to one month in advance. The in-charge nurse is the supervisor in each shift managing and organising all activities in the ward. Head nurse and in-charge nurse assigns tasks to the nurses in the ward. The assignments of tasks are documented into nursing tools such as nurse note, medical record and handover sheet. Figure 5.1 shows the use case of a nursing workflow in a hospital ward. The head nurse, in-charge nurse and nurses are responsible for patient cares, rounding the ward and documenting all nursing activities. There are other staffs such as administration clerks are responsible for doing non-nursing activities such as billing.

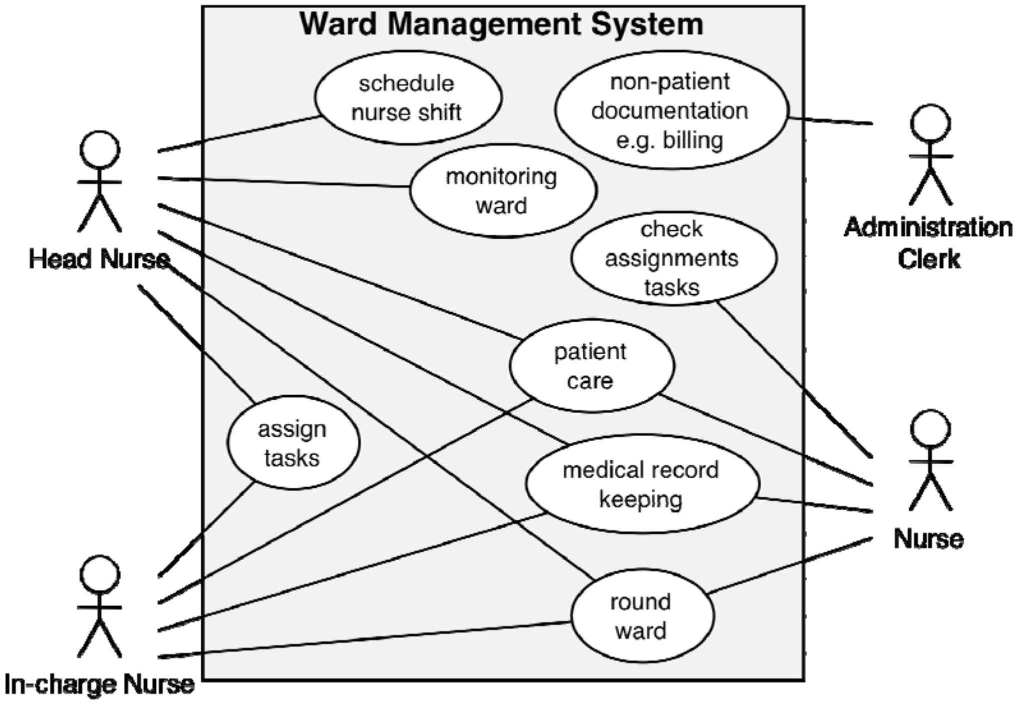



Figure 5.1 Use case diagram of nurses in a hospital ward

We will use the following scenario to illustrate the paper-based nursing workflow and a workflow system using the DPWFM framework.

## 5. 2 Paper-Based Nursing Workflow Scenario

On a Friday day shift after the morning round ward, *Jenny*, the head nurse of an operation ward, assigns several tasks to *Amanda*. *Jenny* writes the tasks of each patient in the Kardex and writes down other assigned tasks in the main white board at the nurse station. There are five patients to task care of today. Two of the patients require blood tests and the samples are to be sent to the Haemoglobin Lab. Another patient is to be prepared for a surgery in the late morning; information, appointments for doctor and theatre room need to be scheduled. *Amanda* checks her tasks that she has been assigned in the nurse station and prepares to deliver patient care to her patients. She will re-check the main board from time to time to check if any new tasks have been assigned. Some ad-hoc tasks may be assigned verbally by the head nurse, so she needs to remember these new tasks and completes them too.

In the late morning, another two new patients are admitted and are assigned to be under *Amanda*'s care. *Jenny* also assigns *Amanda* another new task that is to process the discharge documents for all patients in the ward who will be going home that afternoon. *Amanda* is very frustrated because she has to prioritise the tasks and do the most urgent tasks and at the same time she needs to retrieve her new patients' profiles. As a first task she decides to prepare for blood sample because the Haemoglobin lab requires a few hours before the blood results can be reported back to the ward. At eleven o'clock, *Jenny* asks *Amanda* to help *Lucy*, her colleague, to distribute medicine to *Lucy*'s patients when she also delivers the medicine to the patients under her care. This is because *Lucy* has to take care of seven seriously ill patients and is kept occupied with a patient who is heavily sedated at that time. Amanda has to check the drug plans of her patients and that of Lucy's patients and dispatches the medicine to each patient before lunchtime. This produces high level of stress for *Amanda* and she feels frustrated and do not know what she should do first.

When she has some free time later, she goes back to the nurse station to check the blood test results and process the documents for patient discharge reports. *Amanda* informs *Lucy* about medication reaction of *Lucy*'s patients because one of *Lucy*'s patients had an adverse reaction to the medication and has vomited. Afterward, *Amanda* needs another fifteen minutes to recall what she has done in the morning and writes down those details in her nurse notes, fill in the usages resource report and the medical records report for her patients. In normal circumstances, she should have recorded everything immediately or as soon as she completes the tasks. However, it has been a very hectic morning and she had not have any time to write the details down. She had only written short notes to remind herself and plans to rewrite the full details in the handover sheet before the next shift. During the shift, *Jenny* is also monitoring performance of her subordinates. *Jenny* needs to check the Kardex cards and main board for tasks that have been completed. Sometimes she needs to ask the patients about taking medication because the nurses had not recorded them on the main Kardex.

### 5. 3 A DPWFM Walkthrough for Nursing Workflow Scenario

We will now apply the DPWFM framework to the above scenario. The web service repository performs the job of a controller for directing the DPWFM server to complete the tasks. Figure 5.2 shows the RS service in the DPWFM server that connects to web service repository such as the WPS, FS and FAS. These web service repositories will help in performing the sequencing, ordering and scheduling of the submitted requirements into a planned priority schedule for the agenda workflows. The RS recompiles and generates these agenda workflows for individual web application after executing of allocated requests in the FAS. *Jenny* and her subordinates' details (*Amanda* and *Lucy*) are stored in the WPS that contains database scripts to fetch the information of the nurse profiles (name, role, priority and function assign and manage). The FS includes examples such as writing report, dispatch medicine, deliver patient care and Kardex arrangement.

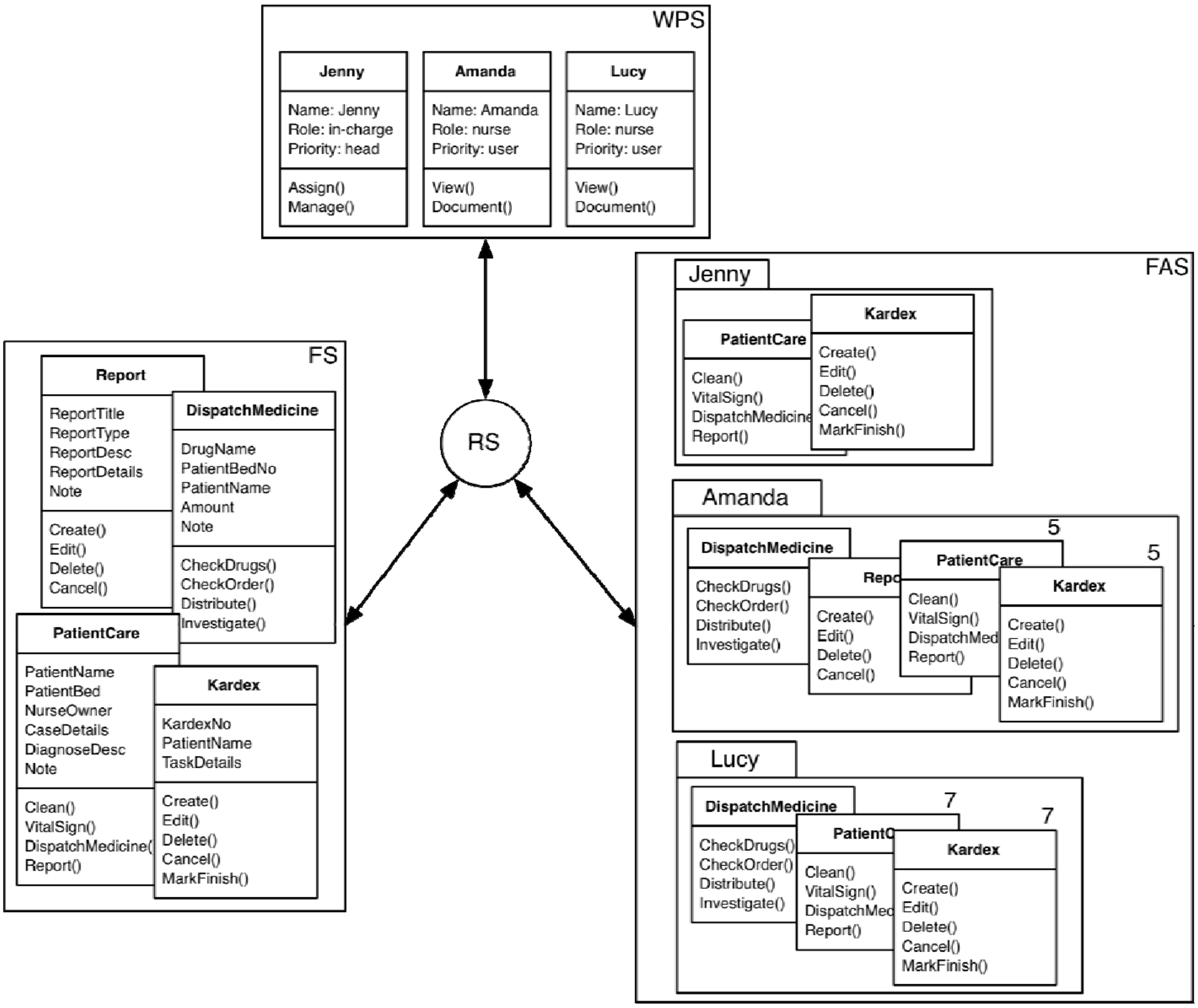



Figure 5.2 Examples of nursing care services

Let us now have a walkthrough for the DPWFM system using the same scenario described previously. When the supervisor, *Jenny*, access to the DPWFM system, a login authorisation screen is shown (see Figure 5.3).

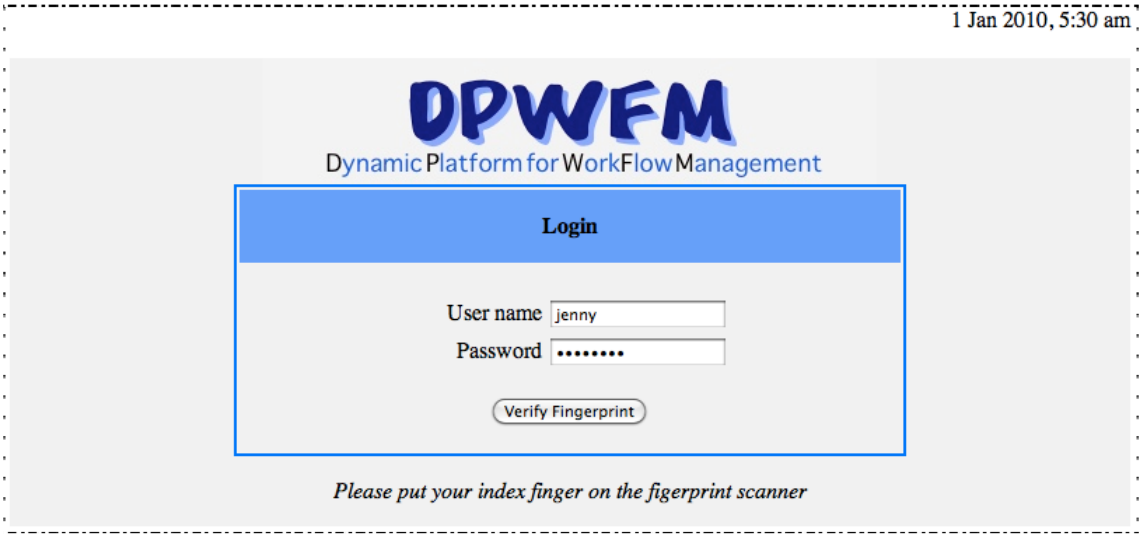



Figure 5.3 *Jenny* login page

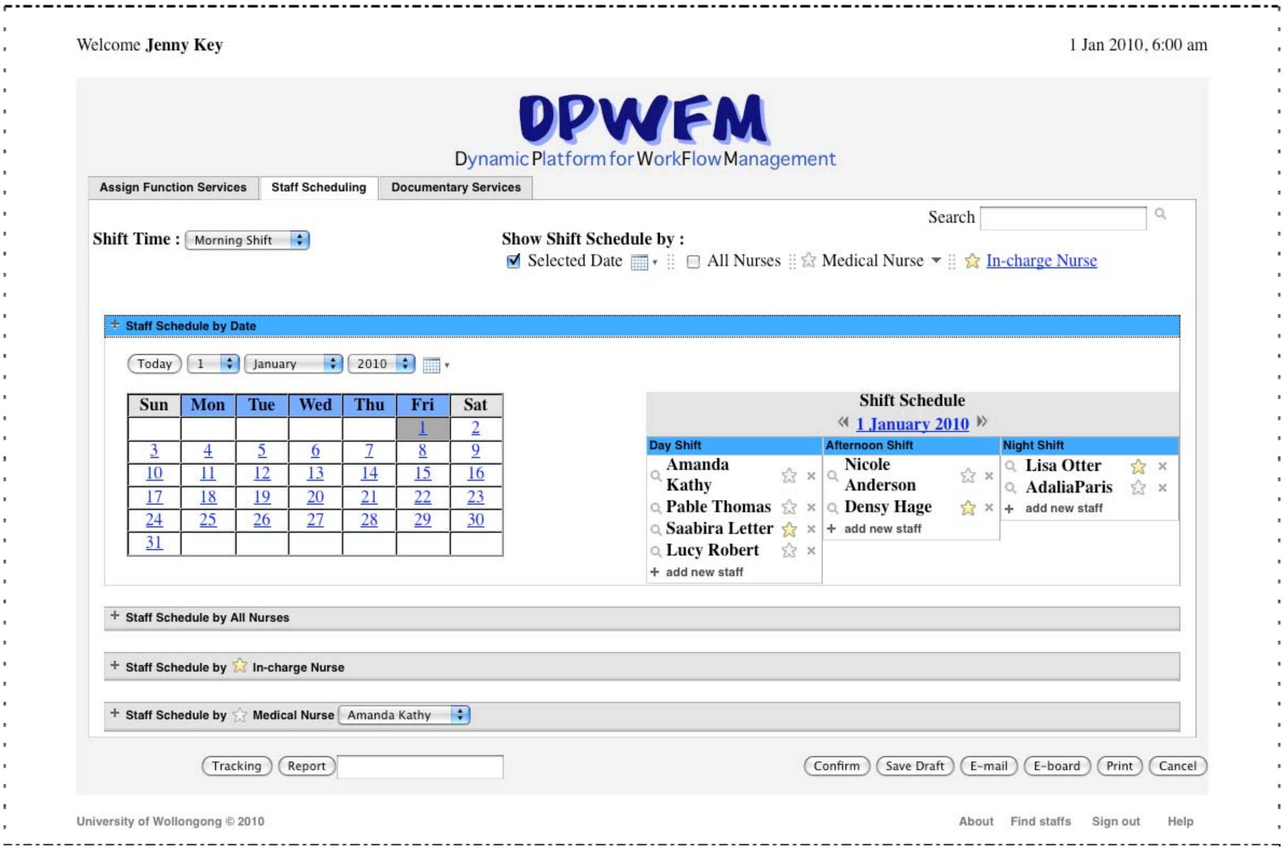



Figure 5.4 Assigning shift schedule viewed by *Date*

After *Jenny* enters the correct identification keys, she can assign tasks in the FAS to *Amanda*, *Lucy*, *Pable* and *Saabira*. Generally, the schedule of the shift is planned one month in advance. *Jenny* can customise the shift allocation based on a specific date (see Figure 5.4). In addition, *Jenny* has the flexibility to view the shift allocation by day as shown in Figure 5.5.

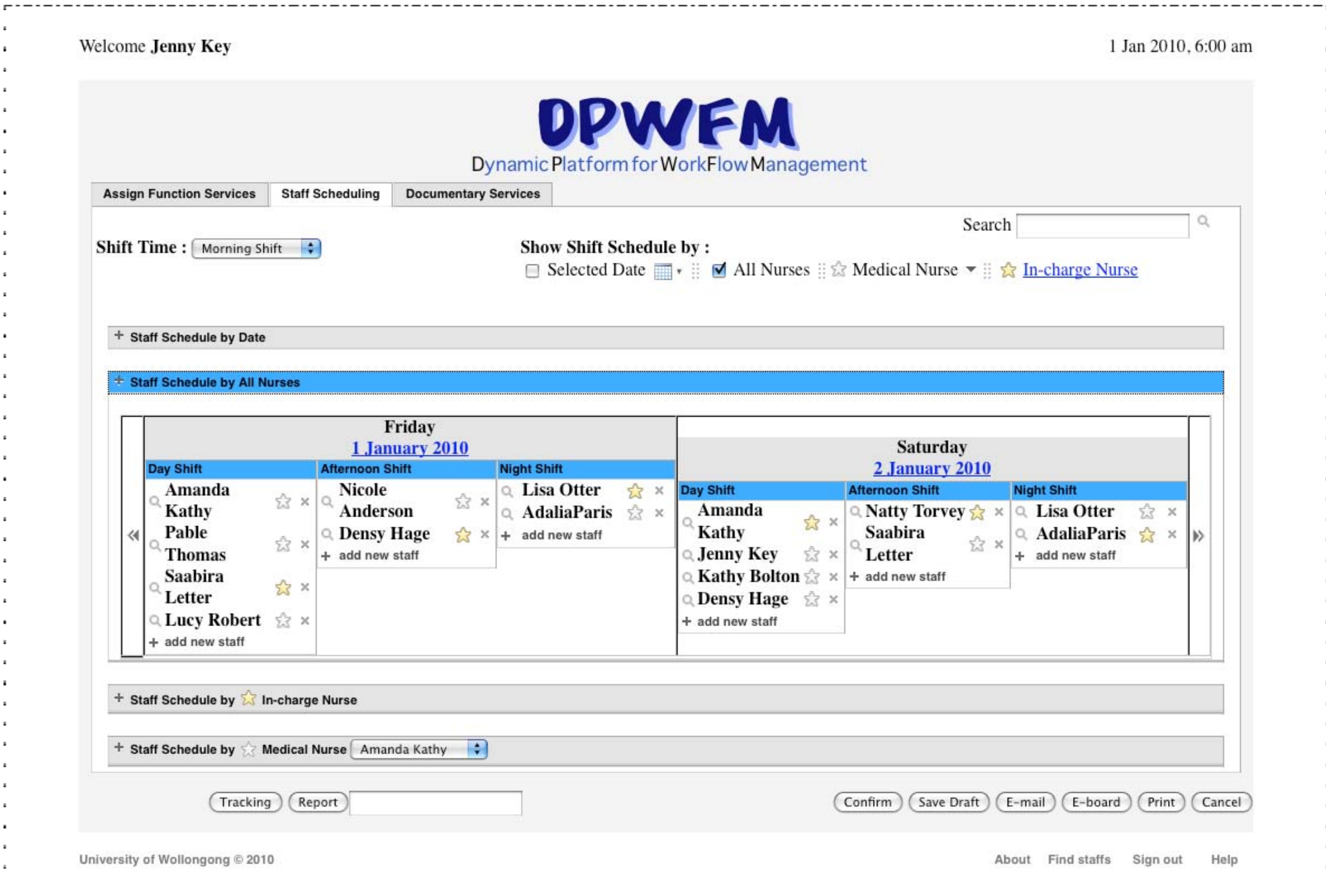



Figure 5.5 Assigning shift schedule viewed by *All Nurses*

Figure 5.6 shows the in-charge nurse schedule of each shift. For example, on 1st January, *Saabira* works as an in-charge nurse in a day shift, *Densy* works on the afternoon shift and *Lisa* works on a night shift. Another view of nurse-shift schedule that shows a specific nurse such as *Amanda* is shown in Figure 5.7.

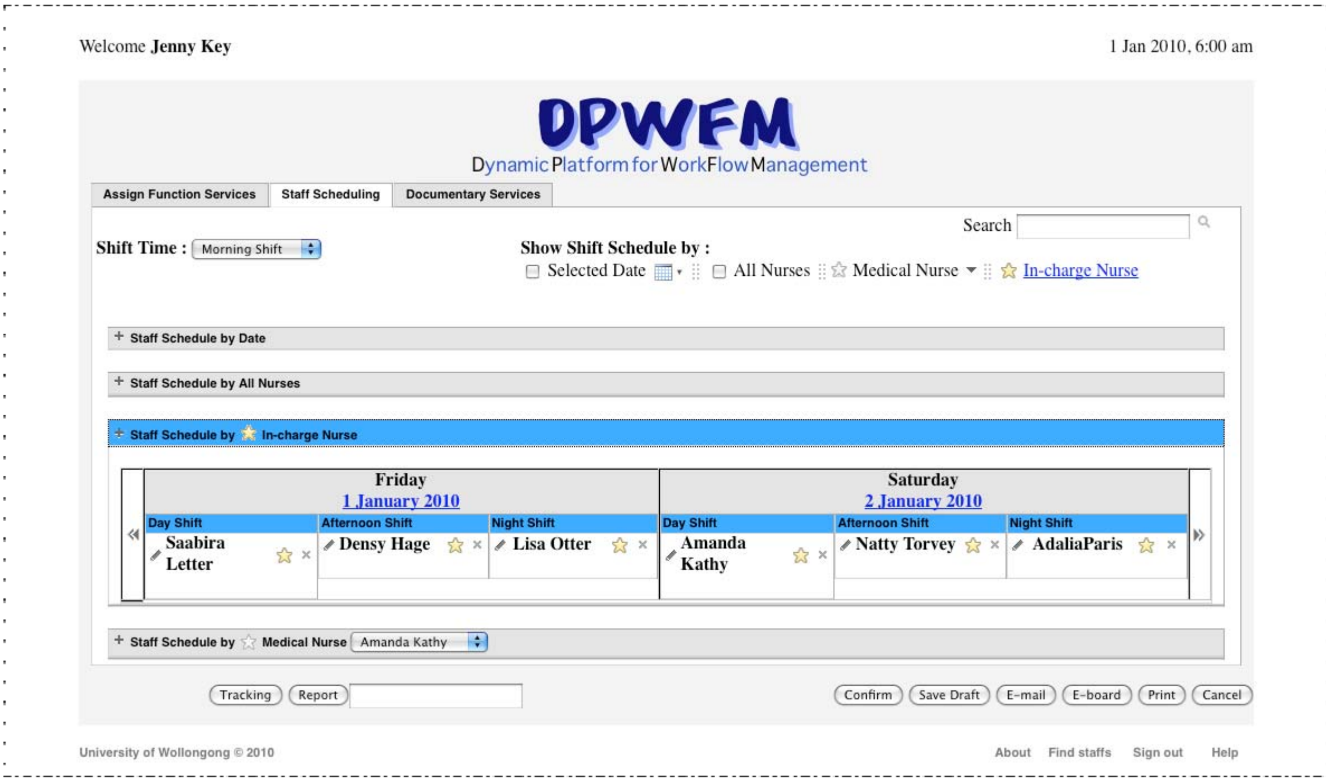



Figure 5.6 Assigning shift schedule viewed by *In-charge Nurse*

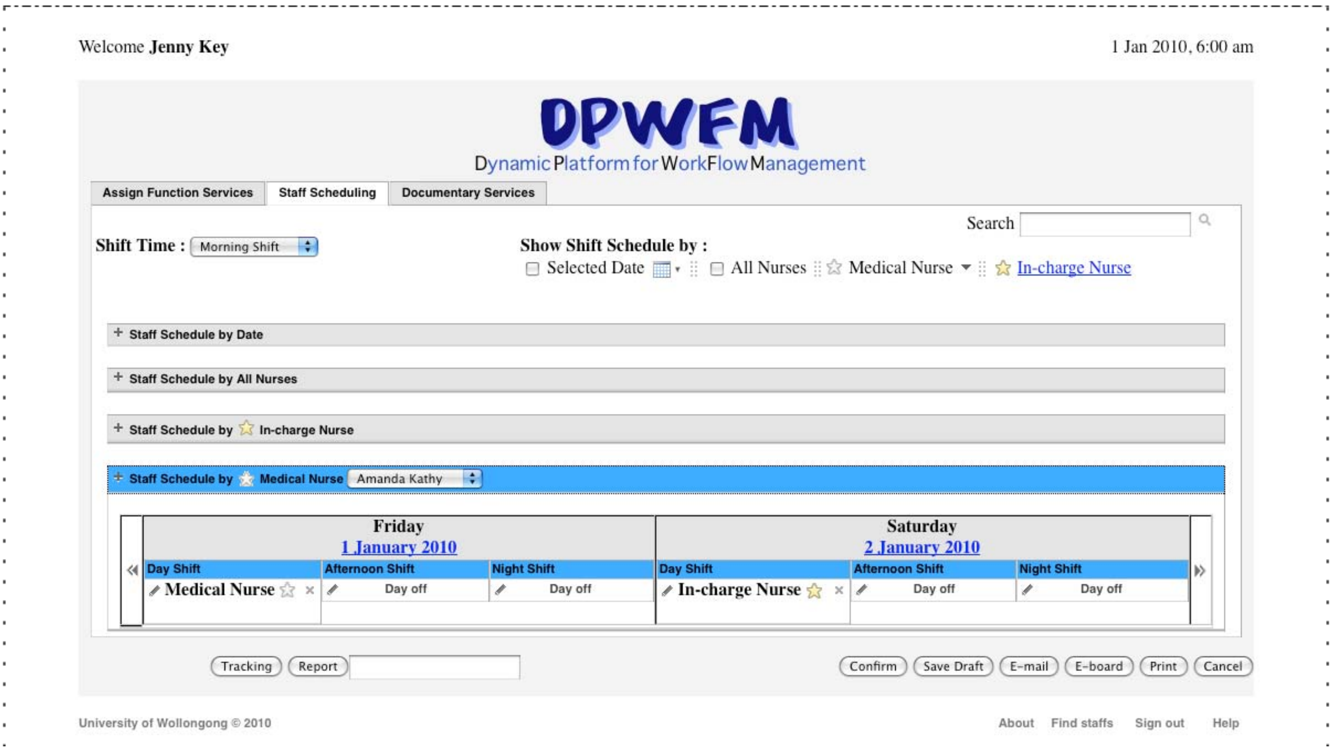



Figure 5.7 Assigning shift schedule viewed by *Specific Nurse*

Figure 5.8 shows *Lucy*'s seven patients that were assigned by *Jenny*. In this example, these seven patients are *Dachelle*, *Vaanda*, *Babra*, *Abbie*, *Jan*, *Janet* and *Dacian*. During the shift, *Jenny* may handover the patients to another nurse if unpredictable circumstances occur such as the severity of one patient may require urgent care of one nurse.

Figure 5.9 shows *Jenny*'s web application that displays all four nurses on duty at the upper part of the screen. Consider *Amanda* being assigned to take care of five patients and *Lucy* has seven serious cases on hand. Thus, she assigns five patients to *Amanda* by checking the boxes of these five patients. In this example, these five patients are *Desiree*, *Michael*, *Isabel*, *Saacha* and *Lazier*. Figure 5.9 also shows the patient profiles and a specific nurse. The patient information includes patient ID, patient name, bed location, case descriptions (based on the doctor's orders) and allergic history.

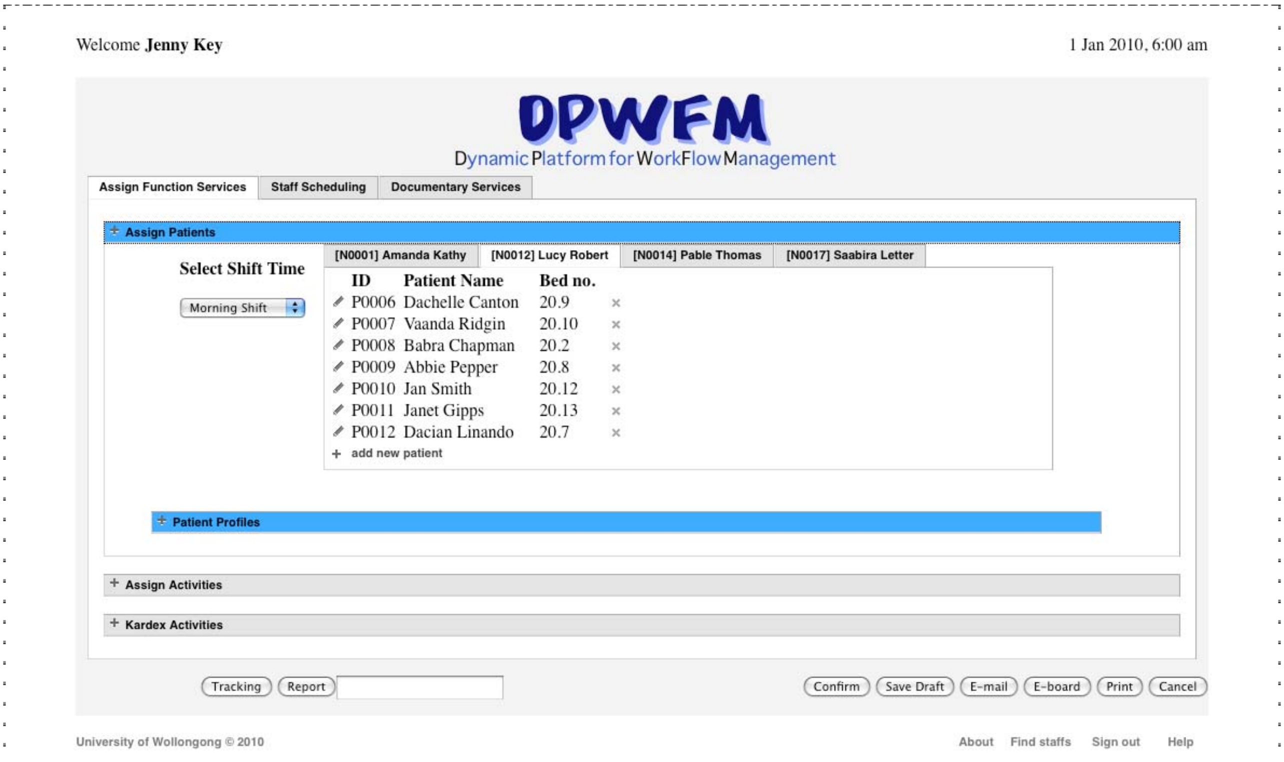


Figure 5.8 *Jenny* assigns seven patients to *Lucy*

Figure 5.9 *Jenny* assigns five patients to *Amanda*

The DPWFM provides the tasks arrangements in a systematic manner; the high priorities tasks are highlighted and listed at the top part of the Kardex lists. After *Amanda* logs in to the DPWFM, she can follow the scheduler services that are shown on her personal web application to carry out her tasks.

Figure 5.10 shows all Kardex activities of *Amanda* for her five patients. The high priorities tasks in *Amanda*'s Kardex list shows that two of *Amanda*'s patients (*Desiree* and *Michael*) need to have blood test and another patient (*Saacha*) need to be prepared for surgery in the late morning and vein puncture must first be organised.

The surgery has been decides by the owner doctor. A surgery appointment has been made based on availability of surgeons, anaesthetists and surgery facilities (surgery theatre and surgery equipment needed). After surgeons and anaesthetists confirmed the surgery appointment with the DPWFM, the nurse can prepare the patient to be ready for surgery. Then, the nurse will inform the surgery doctor for the operation again and confirm the booked theatre before a surgery takes place.

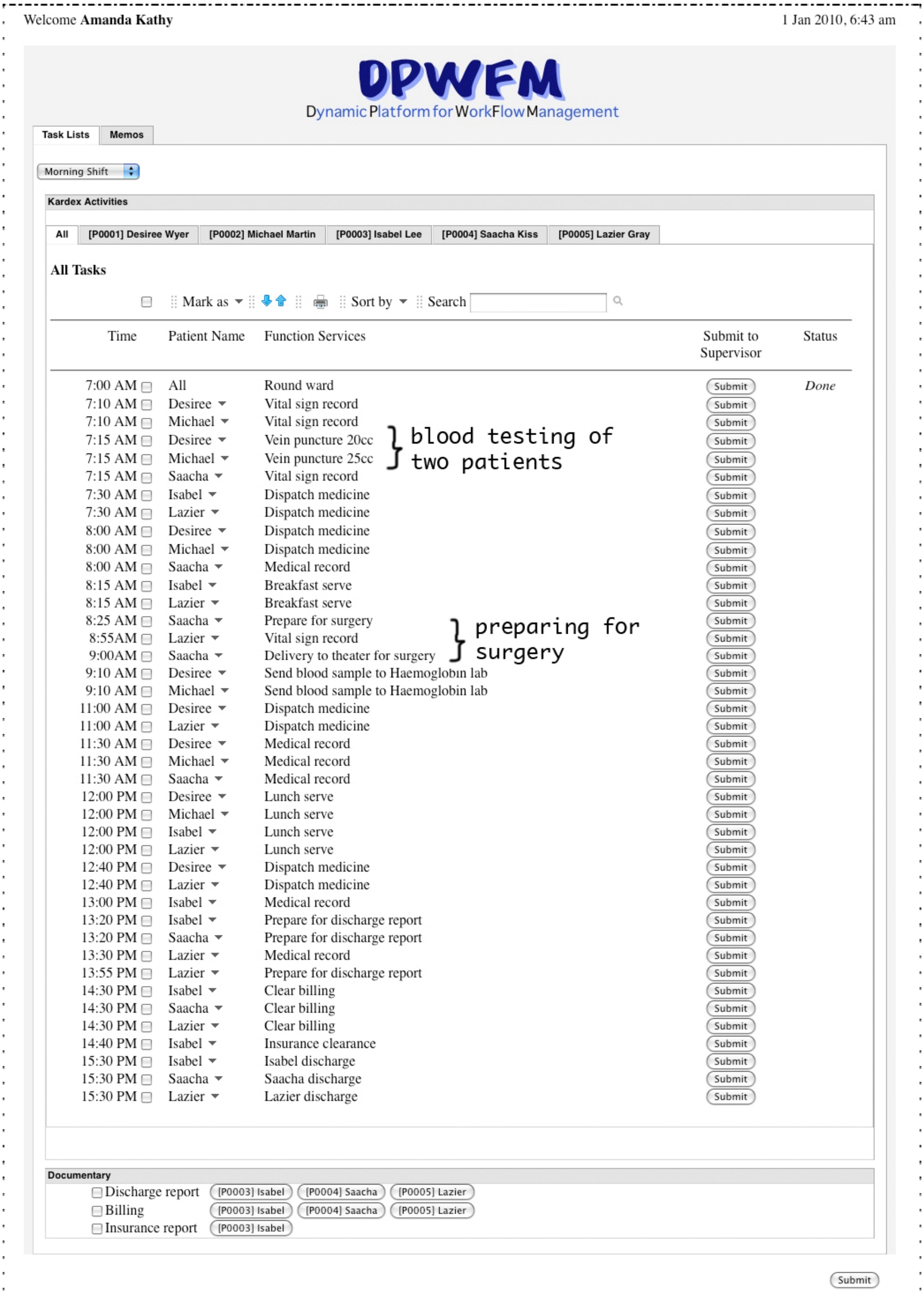



Figure 5.10 Kardex activities for all patients for nurse *Amanda*

Then, in the morning, *Amanda* checks a surgery appointment of surgical operation time from a surgery doctor (Dr. Kim) and checks the booked notice of a theatre room (room 1A) via the DPWFM services. The operation times will be marked on the doctor's schedule by the DPWFM since the surgery time has been confirmed. The available theatre room is also booked for this patient (*Saacha*). Figure 5.11 shows the selection of the service of "prepare for surgery" by click the checkbox in the task list. Then, the service window is pop-up for *Amanda* to view the details of booking the surgical operation time and surgeon.

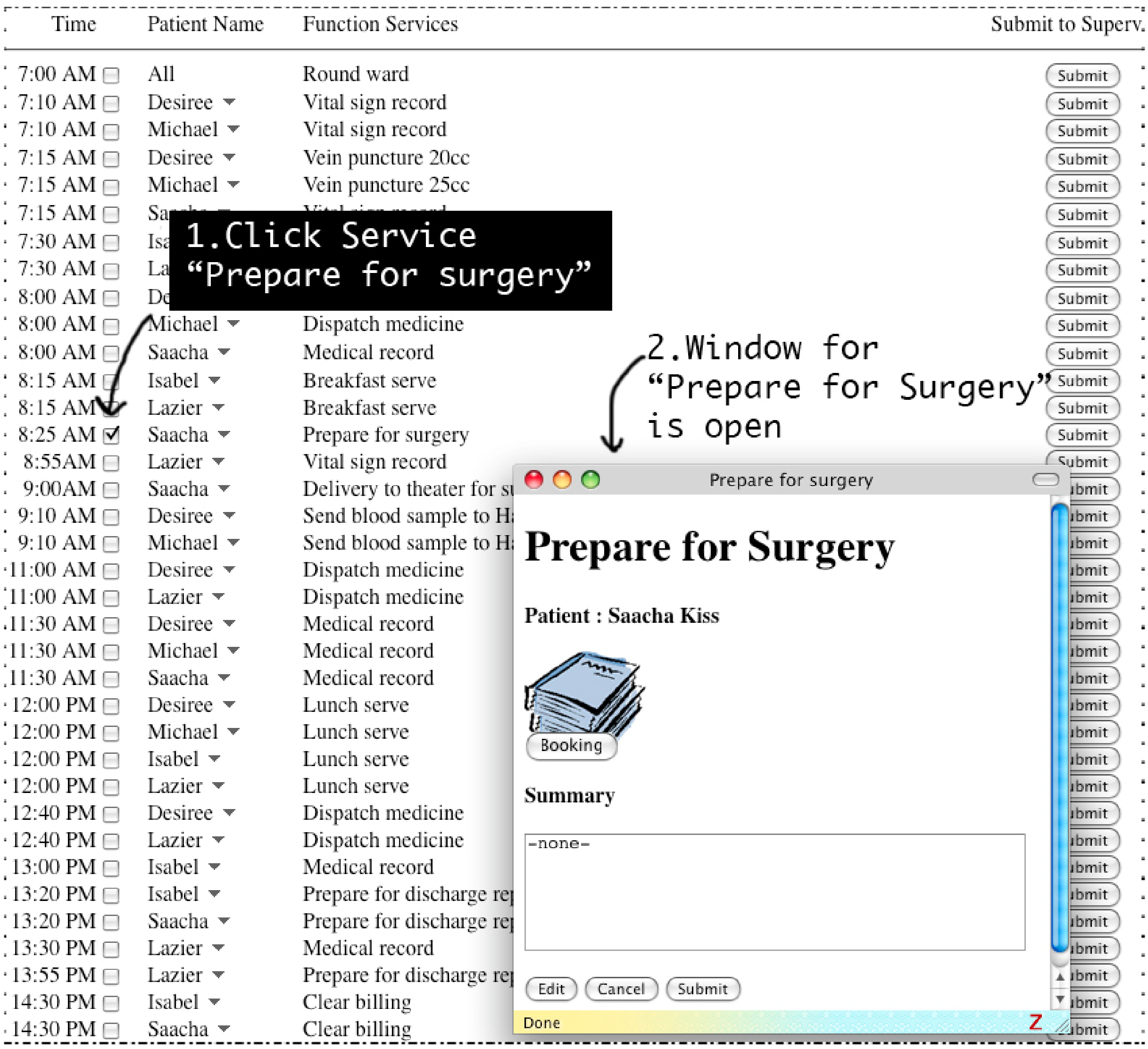


Figure 5.11 Select the service of prepare for surgery

Next, *Amanda* clicks the button  to check the booking time/date for surgery. Figure 5.12 shows the procedure steps of completing the preparation for surgical operation. The first step is to check the doctor name that is Dr. Kim, view Dr. Kim's schedule by click at the calendar button . Then, the schedule of Dr. Kim is displayed. The 1st January has been booked for *Saacha*'s operation in Dr. Kim's schedule is displayed and the time of operation is 09:00am.

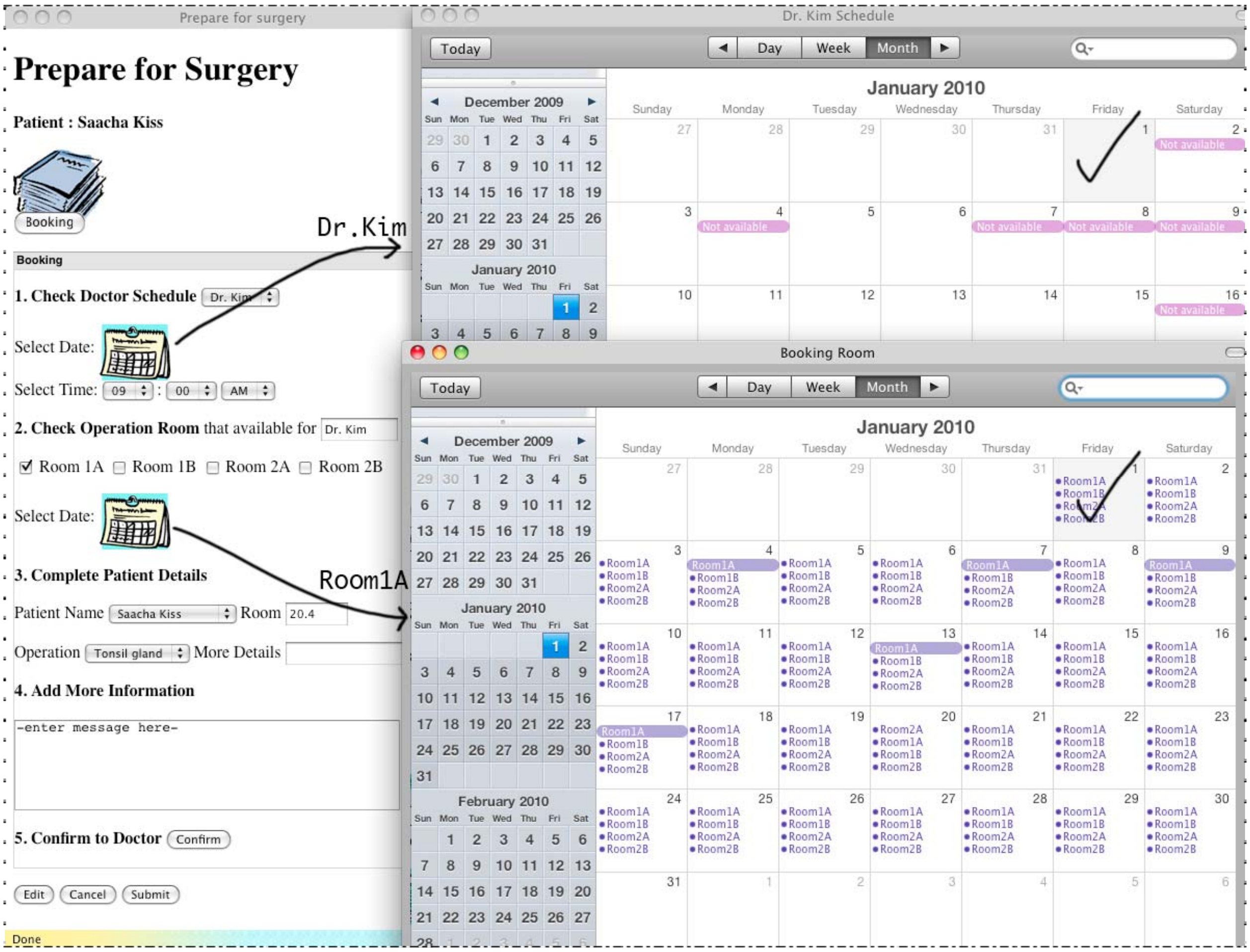



Figure 5.12 Confirm the appointment for surgical operation

In this hospital, there are four rooms in a theatre (room 1A, 1B, 2A and 2B). The second step is to check the operation room. *Amanda* clicks the calendar button  to view the theatre room, then the calendar window pops-up. After that the DPWFM shows "room 1A" on the 1st of January, which is the date that Dr. Kim is available, is booked. The third step, *Amanda* checks the patient details such as name, room, operation type and add more information in the form. Then *Amanda* sends the confirmation notice to Dr. Kim again. Dr. Kim will receive the information of surgical operation of *Saacha* in his task list. Within a few minutes, the task to prepare for surgery, confirm the appointment of the operation is completed. Moreover, *Amanda* can view the Kardex activities of each patient by selecting individual patient. Figure 5.13 shows the Kardex for *Desiree*. Here the tasks are organised in time sequence.

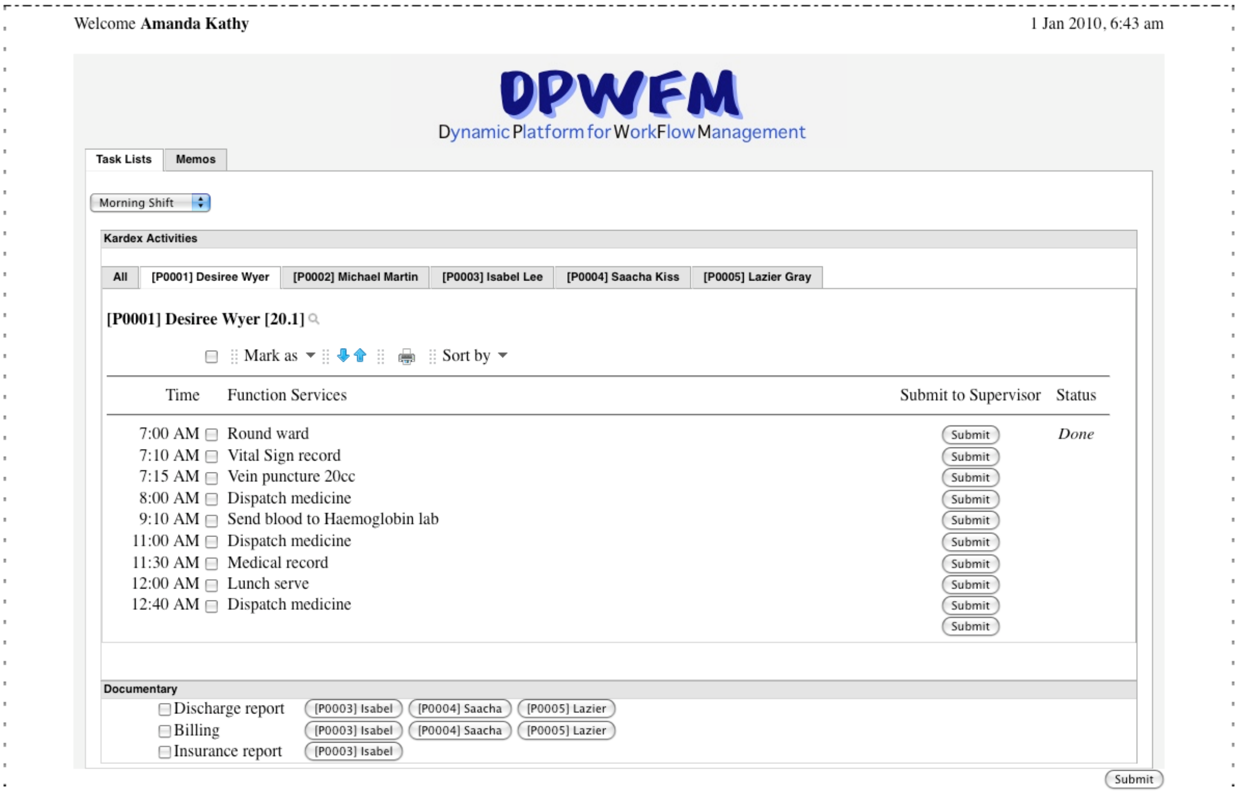



Figure 5.13 Kardex card of patient *Desiree* for nurse *Amanda*

Figures 5.14, 5.15, 5.16 and 5.17 show the Kardex activities for *Michael*, *Isabel*, *Saacha* and *Lazier*, respectively.

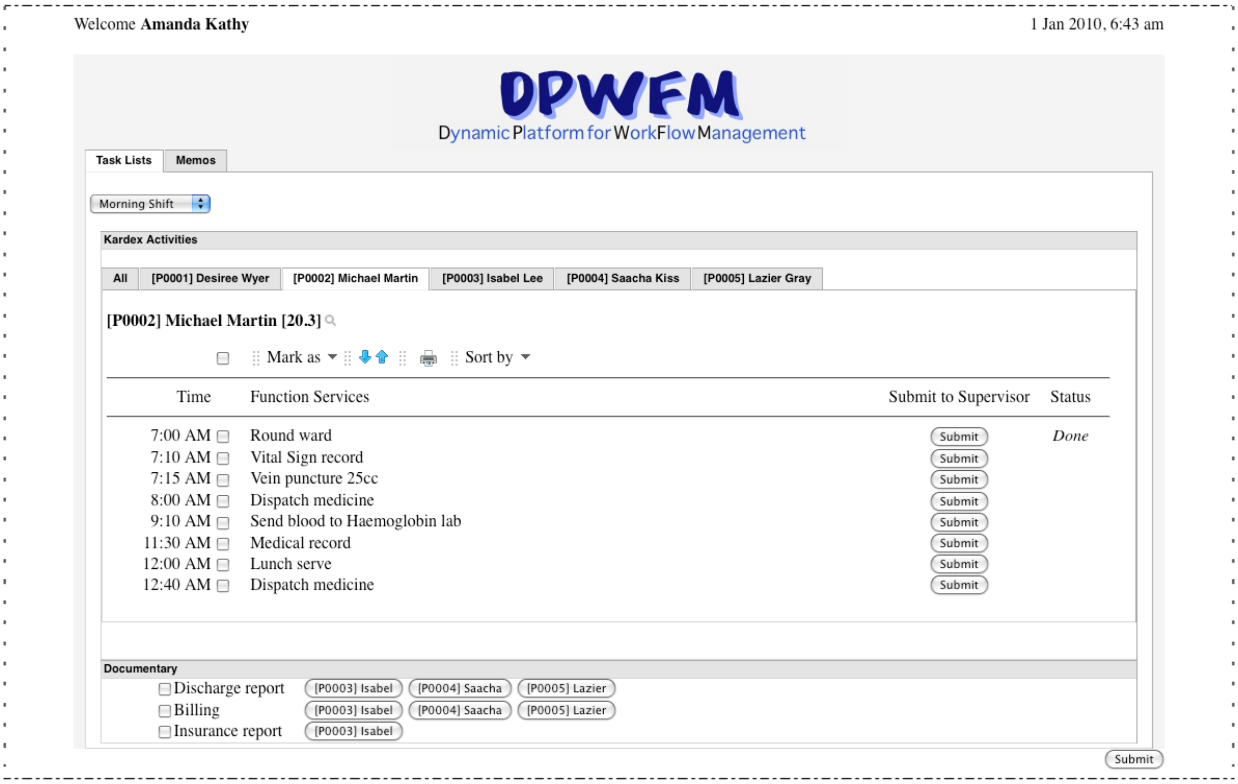



Figure 5.14 Kardex card of patient *Michael* for nurse *Amanda*

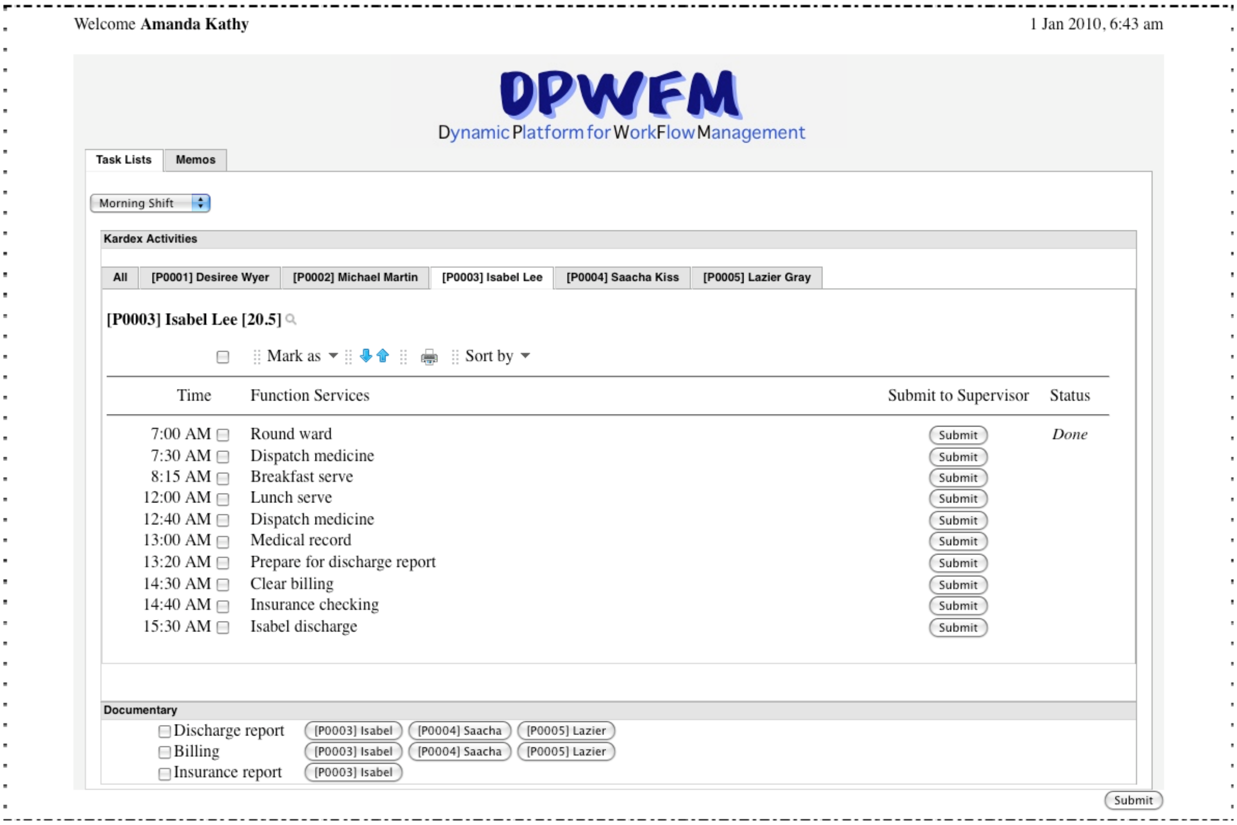



Figure 5.15 Kardex card of patient *Isabel* for nurse *Amanda*

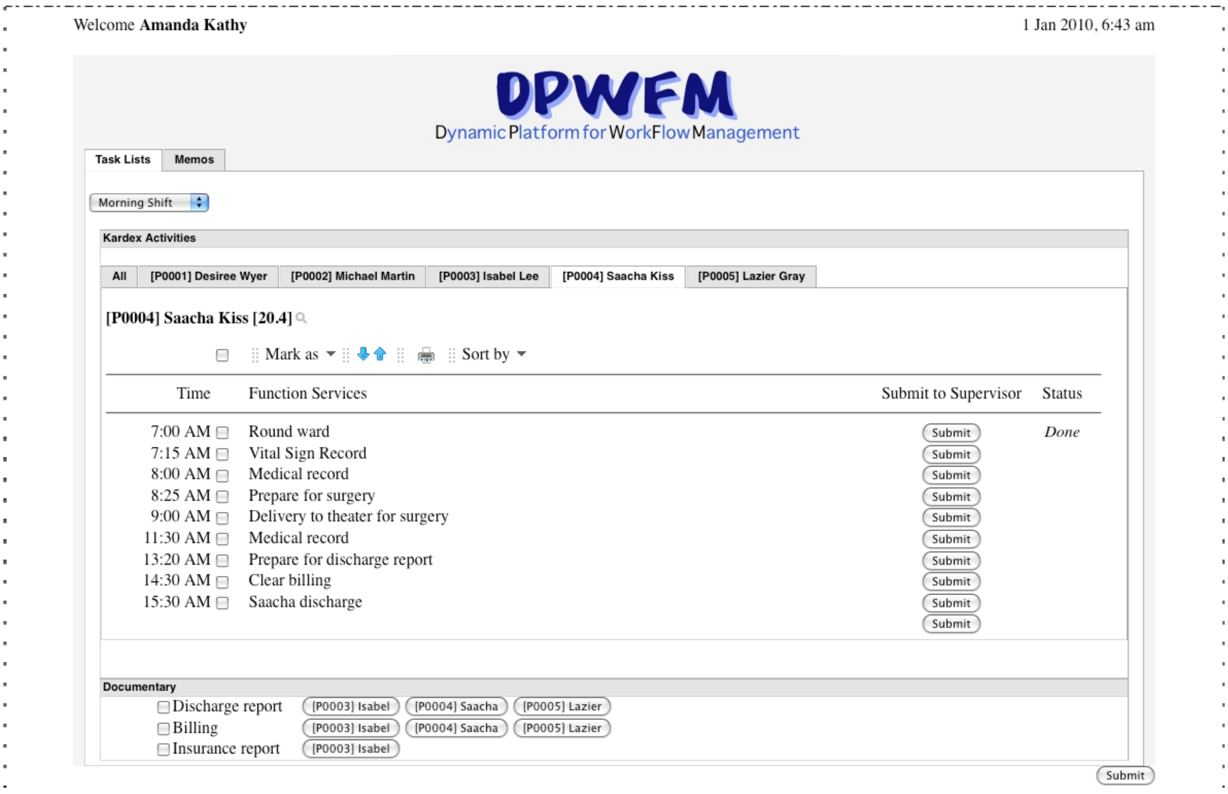


Figure 5.16 Kardex card of patient *Saacha* for nurse *Amanda*

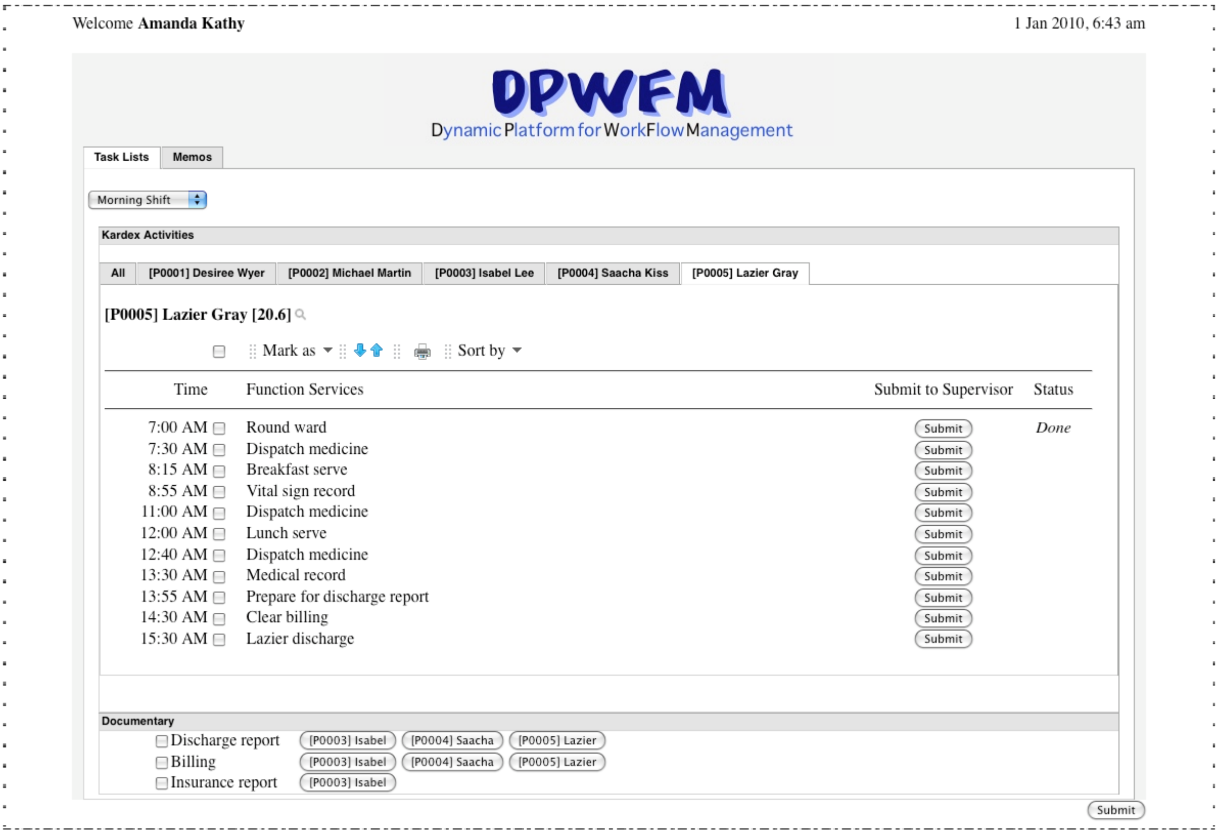



Figure 5.17 Kardex card of patient *Lazier* for nurse *Amanda*

*Amanda* is assigned new responsibility to handle the discharged documents for all patients. In order to do that *Amanda* selects the patients (*Isabel*, *Saacha* and *Lazier*) who are allowed to be discharged and submit information of discharge reports. The system then provides the report function services to the appropriate departments such as finance and insurance to process billing (see Figure 5.18). In order to do that, *Amanda* selects the *Patient Documentary* and click at specific patient. The discharged report window is shown in Figure 5.19.

Figure 5.18 Patient documentary

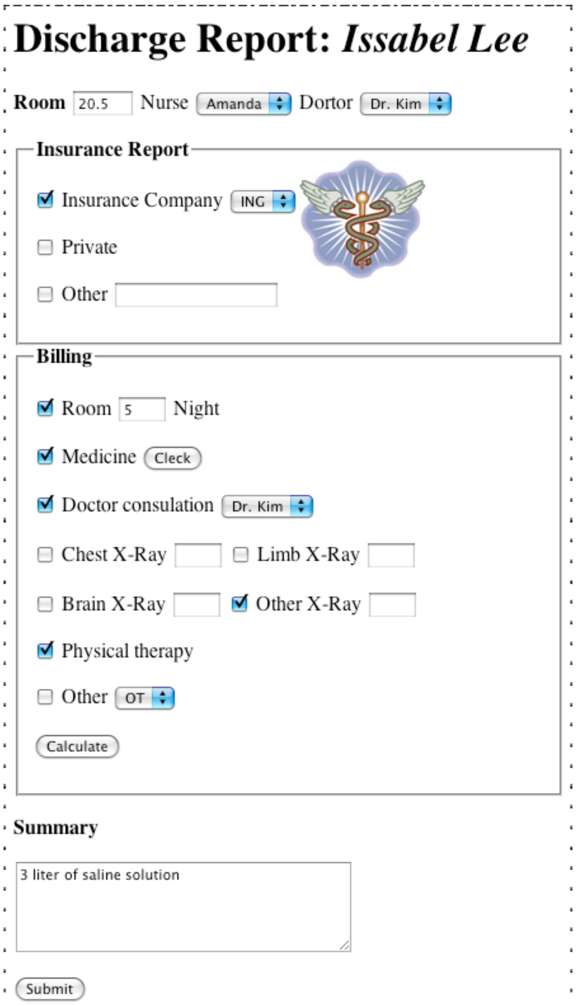



Figure 5.19 Discharge reports for *Issabel*

In the late morning, another two new patients are admitted. *Jenny* assigns these new patients (*Pim* and *Taara*) to *Amanda* via the DPWFM. Figure 5.20 shows *Jenny* has added two new patients and assigned them to *Amanda*. All information and tasks that are related to the new patients are retrieved and then recompiled into a sequence of tasks to *Amanda*'s Kardex service. Figure 5.21 shows the new Kardex activities of new patients (*Pim*) on *Amanda*'s Kardex panel.

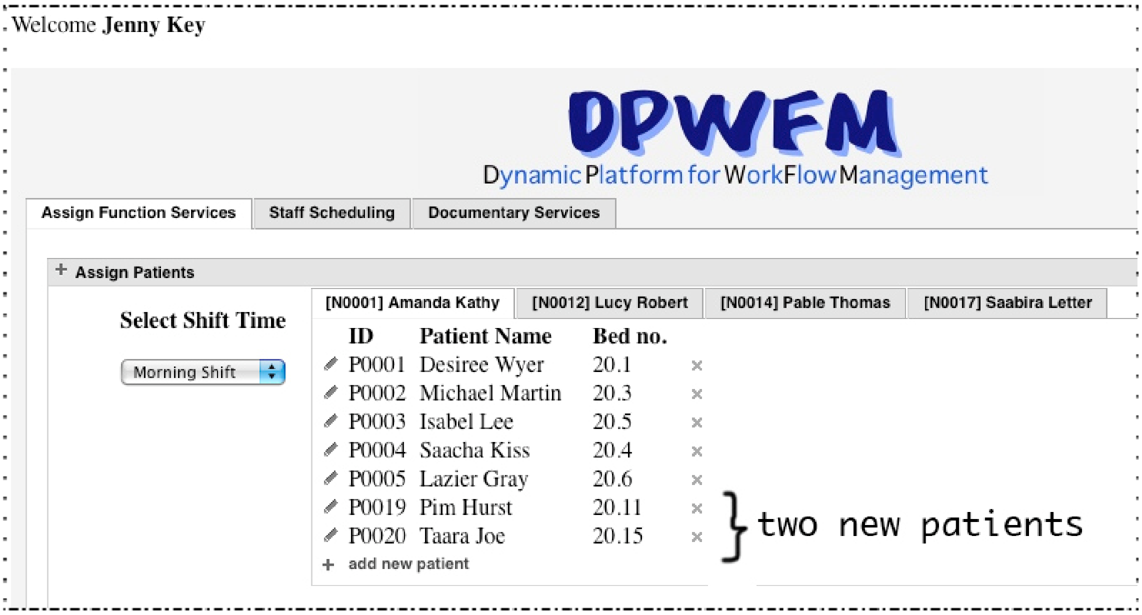



Figure 5.20 At *Jenny* web page: assigns two new patients for *Amanda*

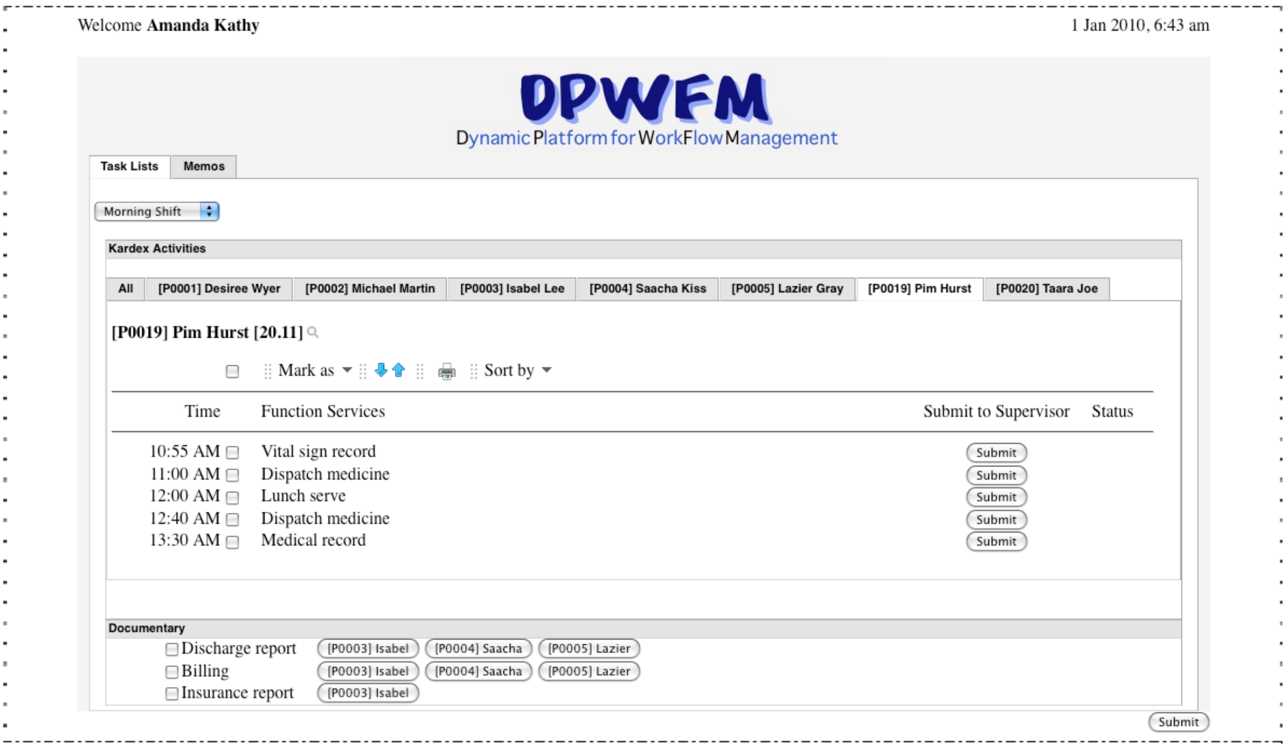



Figure 5.21 At *Amanda* panel: with new tasks of new patients

At eleven o'clock, *Jenny* assigns a new task via the DPWFM to *Amanda* (see Figure 5.22). The task is to help *Lucy* to dispatch medicine to *Lucy*'s patients. *Amanda* notices the new task inserted in her Kardex lists. However, she does not feel stressful because the DPWFM has recompiled the new tasks and her current tasks, and then lists the necessary information such as bed number, time, amount of drugs and patient information to support her tasks (see Figure 5.23). *Lucy* can also check reactions of her patients (after they have taken the medicine) via her own web application as soon as *Amanda* has entered the information using the DPWFM. During a shift, the DPWFM alerts *Amanda* to check and view lab results from the Haemoglobin Lab after the blood test result is returned. After finishing each task, *Amanda* records her nurse notes in the DPWFM system.

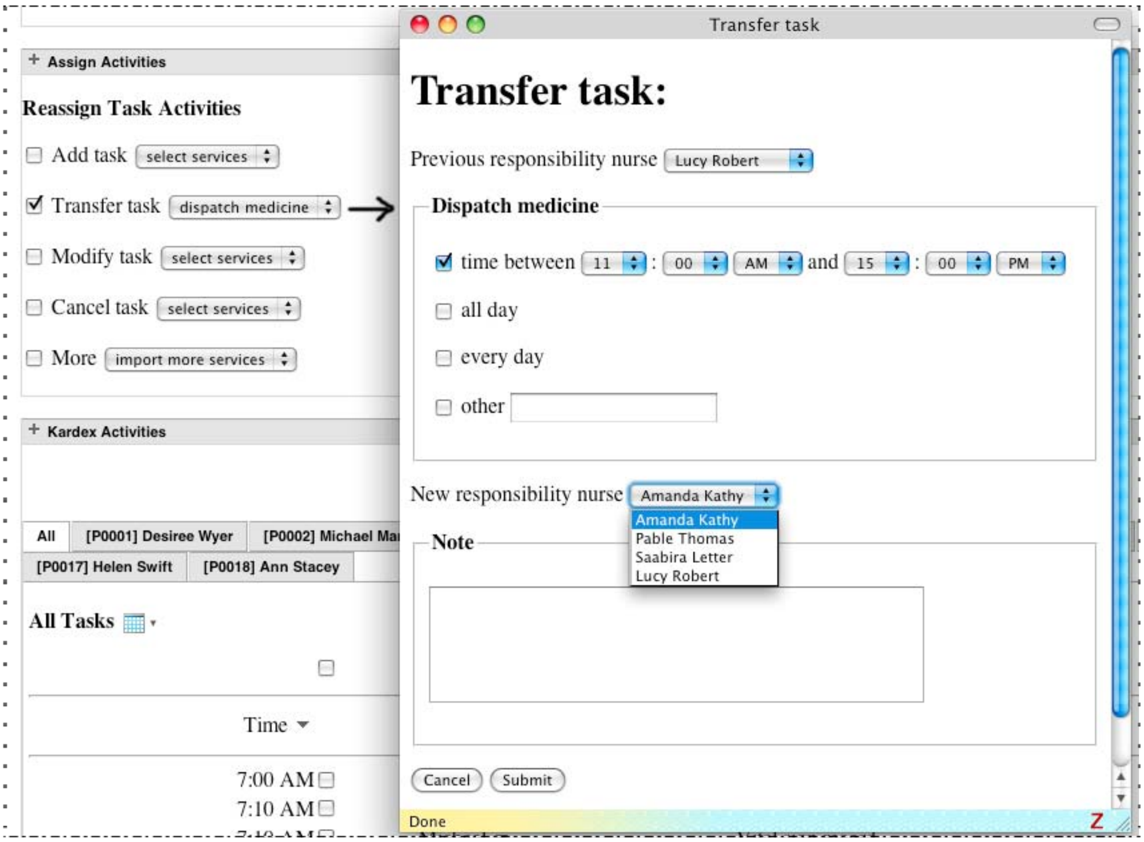



Figure 5.22 *Jenny* allocates a new task for *Amanda*

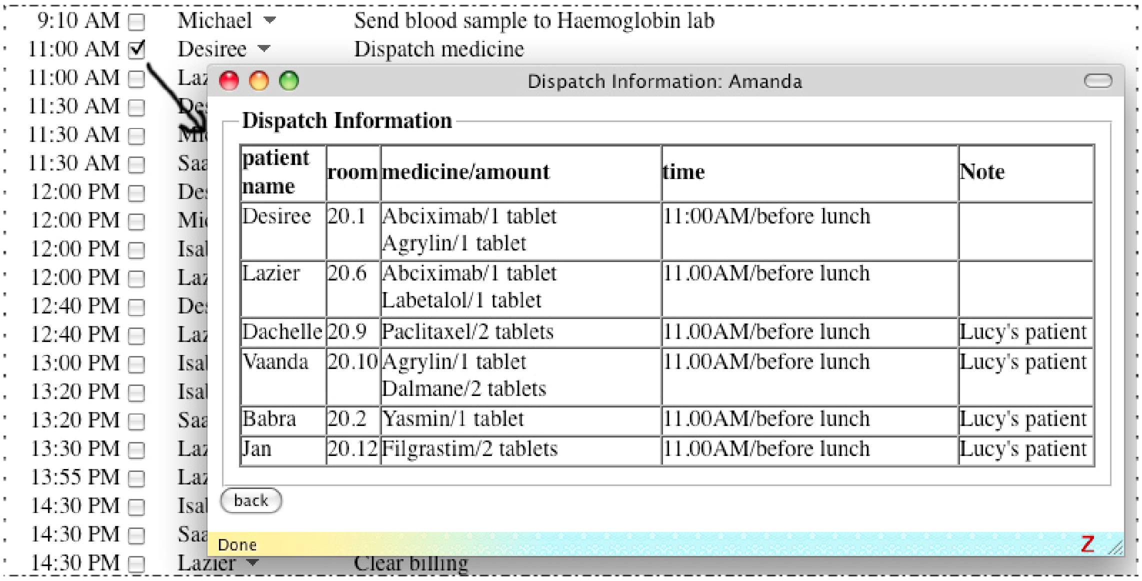



Figure 5.23 List of medicines that *Amanda* should deliver at 11AM

This scenario shows that using the DPWFM system *Amanda* is available to perform better patient care and reduce stress levels because *Amanda* only needs to focus on nursing care while the DPWFM helps her to organise the tasks. Fifteen minutes before the end of the shift, *Amanda* has time to review and fulfil details of all activities before submitting these tasks to *Jenny* (supervisor). The first example of fulfilment of nursing activities, *Amanda* writes notes of finished activities in the Kardex function services, then before click the submit button when she has available time during a break, she returns to fulfil details in the Kardex tasks. Another example is during the shift *Jenny* can monitor her subordinates' performances by accessing the DPWFM on her account as a supervisor. She can receive and retrieve the updated information immediately after her subordinate has submitted information of their tasks. *Jenny* can view the shift schedule of her subordinates too. This scenario demonstrates that the proposed recompilation feature of DPWFM system allows nurses to improve their job performance and to have a ward management that is much easier to organise.

## 5. 4 Chapter Summary

This chapter illustrates the functionalities of the proposed prototype that the supervisor and nurses will view compare to the paper-based workflow. Simple user interfaces with easy to use components such as checkboxes or drop-down lists allow nurses to use the DPWFM for their works. The examples show in this chapter how the supervisor can use the DPWFM to assign and reallocate tasks to the nurses. In addition, the nurses can use the DPWFM system to complete their tasks.

# CHAPTER 6: EVALUATION OF THE PROTOTYPE

This chapter reports the evaluation of the DPWFM prototype. The evaluation was conducted in Bangkok, Thailand by nurses working in eight hospital wards during the month of August 2010. This chapter is organised as follows. Section 1 discusses the evaluation process, section 2 discusses the evaluations results and section 3 concludes the chapter.

## 6. 1 Evaluation Process

The purpose of evaluation is to investigate perceived effectiveness of the DPWFM application in assisting nurses to improve the workflow system in a hospital ward. The DPWFM prototype has been developed with PHP and ASP.NET, the database server using MySQL, and the web server using Apache. The system can work on different platforms such as Windows or Max OS X. The DPWFM prototype server is located at URL *dpwfm.smethaisoft.com*. Source codes are included in Appendix A. The prototype database scripts are included in Appendix B.

The DPWFM prototype was demonstrated to six head nurses and three medical nurses of the In-Patient Department (IPD) wards in five hospitals in Bangkok, Thailand during the month of August 2010. The evaluation process is divided into three parts. Firstly, the main features of the DPWFM prototype were demonstrated to the nurses. We explained how the function services work and different components of the systems interact at different user levels. We have also explained to the nurses how the DPWFM prototype can be included in their current function tasks. Figure 6.1 shows the use of Kardex card in the traditional current system and the use of the Kardex activities in the

DPWFM prototype.

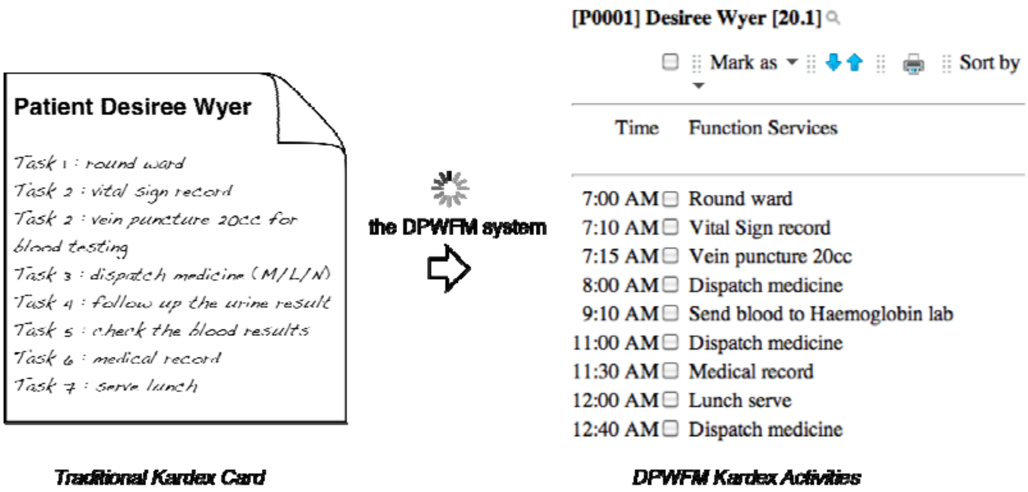



Figure 6.1 Kardex activities in the DPWFM sytem

After the introduction section, we conducted a demonstration of the DPWFM prototype to the nurses and the nurses are asked to enter data to the prototype system using individual data in the ward.

When the nurses access the DPWFM system, they need to verify their identification by user name, password and another verification such as fingerprint. For demonstrate use, the initial authorisation of head nurse as user:*jen* and password:*jen*, the authorisations of medical nurse are user:*ama* and password:*ama*. Figure 6.2 shows the home page of the DPWFM server.

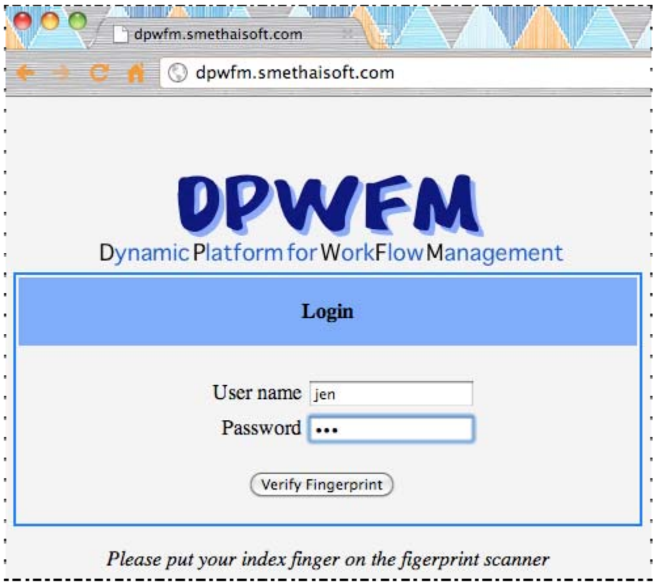
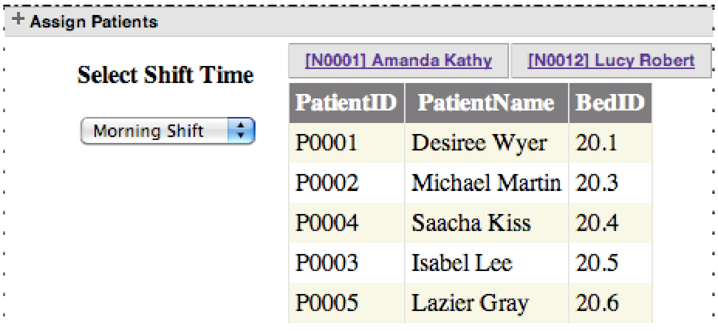



Figure 6.2 The DPWFM home page

In the demonstration, the tasks of the IPD ward had been entered based on in the Kardex activities. The allocation of new patients and the assignment of tasks to individual nurses are demonstrated to the participant nurses. A sample screen shot that shows the new allocated patients assigned by the head nurse (*Jenny*) is shown in Figure 6.3. Figure 6.4 shows the new assigned tasks of patient (*Desiree*) by the head nurse to the nurse staff.




| PatientID | PatientName | BedID |
|---|---|---|
| P0001 | Desiree Wyer | 20.1 |
| P0002 | Michael Martin | 20.3 |
| P0004 | Saacha Kiss | 20.4 |
| P0003 | Isabel Lee | 20.5 |
| P0005 | Lazier Gray | 20.6 |

Figure 6.3 The new assigned-patient service

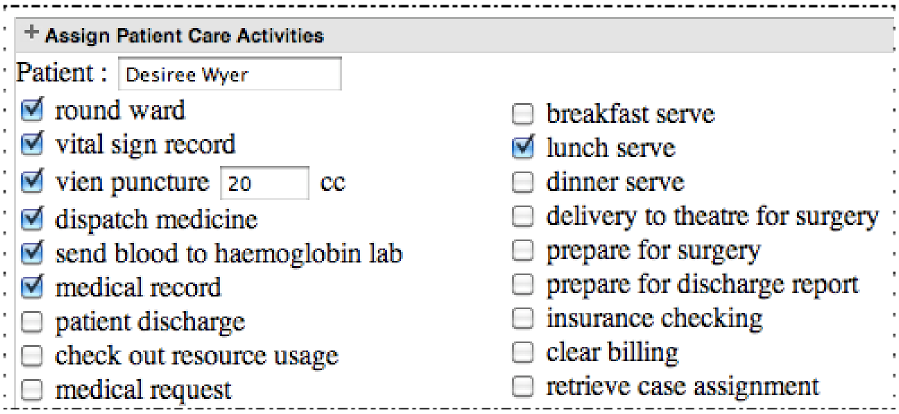



Figure 6.4 The new assigned-task service

After the head nurse has checked the assigned-task services of the patient (this case *Desiree*), a list of Kardex activities will be provide to the nurse (*Amanda*) to execute. Figure 6.5 shows the screen shot of the arranged Kardex activities of the assigned nurse.

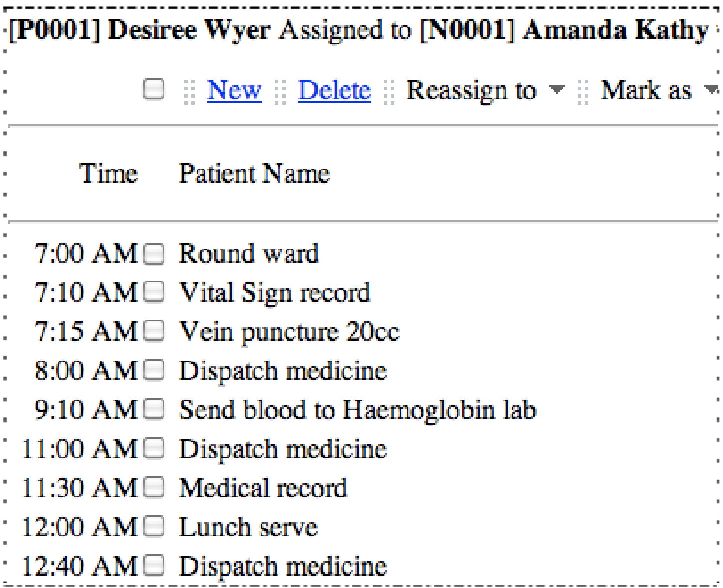



Figure 6.5 The new Kardex activities

The nurse can submit the finished task to the head nurse by clicking the submit button. The status of that task in the nurse staff page and the head nurse page will change to "*Done*". Figure 6.6 shows the screen shot of the submit button in the nurse screen and the status column of the Kardex activities.

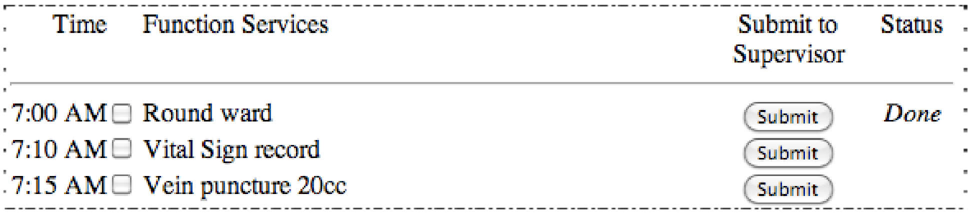



Figure 6.6 The status change to *Done* after the task is completed and submitted

After the prototype has been demonstrated, the following questions were given to the nurses to ask them to evaluate the prototype. The responses were collected from six head nurses and three nurse staffs. We wish to point out that the respondents provide verbal comments and feedbacks in Thai language and these comments were translated to English scribes. (IPD Staffs 2010).

Q1: *Would you be comfortable in using the DPWFM system in your work?*

Q2: *Would you be comfortable introducing the DPWFM system to your staff?*

Q3: *Do you think the DPWFM system can help in reducing instructional and interpretation errors in your work?*

Q4: *Do you think the DPWFM system could improve efficiency in carrying out your daily work?*

Q5: *Do you think the DPWFM system could increase or decrease levels of stress in your work?*

Q6: *What is your view in relation to the advantages and disadvantages of using the DPWFM application in a ward hospital setting?*

## 6. 2 Evaluation Results

Q1: *Would you be comfortable in using the DPWFM system in your work?*

All respondents have indicated that they feel comfortable in using the system. An example of comment from participant A is:

> "*I feel comfortable with the DPWFM system. From the demonstration, I think it is similar to my current nursing tools. I just only need to use computer instead of writing on a paper. It seems not too difficult for me to use.*" (Comment by participant A)

Q2: *Would you be comfortable introducing the DPWFM system to your staff?*

All respondents are comfortable to introduce the DPWFM system to other nurses. Five respondents think they will introduce the system to other staff and three respondents think the user interface of checkboxes and list boxes are useful to use the system. The following show three comments from the participants A, D and E:

> "*The DPWFM system looks all right and the DPWFM interface looks friendly to me. I am comfortable to introduce the DPWFM system to my staff. I think we are able to use it after having some training.*" (Comment by participant A)

> "*I think it is similar to the way we input information to the Kardex cards. But I am worry about how we key-in the information to the DPWFM system, because we are familiar with the writing style. The checkbox and list box selection look easier than fill-in data by typing. Therefore, I am comfortable to introduce the DPWFM system to my staff.*" (Response from participation D)

"*I am comfortable with using the DPWFM system and will introduce the system to the nurses. I think it will help me to monitor the nurses' tasks much more easily and reports from nurses can be sent to me quicker.*" (Response from participation E)

Q3: *Do you think the DPWFM system can help in reducing instructional and interpretation errors in your work?*

All participations respond positively to this question and they think the system can reduce errors. Six respondents think the provided guidelines of the system can help to reduce errors and two respondents think it is useful during the busy period. The following comments are received:

"*I think the DPWFM system should be able to reduce the instructional and interpretation errors, because the nurses will be able to know what the task is from the Kardex lists and which task will come first and which task will come next.*" (Comment by participant A)

"*A current problem is when I assign the tasks to my staff, sometime I assign through face-to-face communication, which may lead to interpretation errors when we are in the busy hours during the morning shift. The details of tasks can be missed and lead to incomplete reports. I think if the DPWFM system can input the assignment easily and nurses can easily retrieve and view their tasks, it will be able to reduce interpretation errors in my ward.*" (Comment by participant B)

"*I think it can help to reduce the errors. I am doubtful about the time it takes when we need to enter the tasks in the computer, because our ward has only a few* *computers. In addition, we can assign the tasks by face-to-face, it seems to be a duplicate task if we need to use the DPWFM system to assign tasks to nurse staffs. However, it will be helpful when we are able to check and monitor the assigned tasks for evaluating the performance of the nurse.*" (Comment by participant D)

Q4: *Do you think the DPWFM system could improve efficiency in carrying out your daily work?*

All participants respond positively to this question. Four respondents have indicated that the system is able to improve efficiency in their works and two respondents think the system is useful when working with complicated tasks. Examples of comments from participant D, F, G and I are as follow:

"*I think it can help to improve the work performance. The DPWFM system will work well with the new generation nurses, but for our generation it will be difficult for us to learn the computerised system. However, if the hospital installs the DPWFM system in the future and we have to use it, we need to have some training.*" (Response from participant D)

"*I know that the computer system is very helpful. It should be able to improve our efficiency in carrying our tasks. However, we are the middle-aged nurses who are not familiar with the computerised system. We need to have time for training to use the new system. As I tried to use the DPWFM prototype, I think I can use it and it will help us to communicate to another nurses.*" (Response from participation F)

"*I think it can help us to improve the work performance. The good idea is that the DPWFM system will compile the systematic sequence of tasks for us to follow as guideline instructions. That will be helpful when we have many patients and should cooperate with too complicate tasks.*" (Response from participation G)

"*I do not have the problem with using a computer. I think it can improve the work performances.*" (Response from participation I)

Q5: *Do you think the DPWFM system could increase or decrease levels of stress in your work?*

Five respondents think the system may increase the stress because their generation are not familiar with the computer system. Two respondents are familiar with the computers so it will not increase their stress levels. The following show three responses from the respondents:

"*I think the DPWFM system may increase the levels of stress due to the majority of nurses are in the middle-aged. However, it might be only in the first stage, after we have training and be familiar with the DPWFM system, it will be all right for us to continue using the system for our work.*" (Response from participation D)

"*I think the DPWFM system is able to increase the levels of stress because we did not use the computer system in our work before.*" (Response from participation H)

"*I think the levels of stress might be the same because nursing tasks are still the same and I am familiar with the computer system.*" (Response from participation I)

Q6: *What is your view in relation to the advantages and disadvantages of using the DPWFM application in a ward hospital setting?*

All participants have given the advantages and disadvantages of the DPWFM system. Seven respondents found that the system's advantages are easy to follow the task instructions and helpful during busy period. The system's disadvantage is because there is only a few computers each ward, so that deployment of the system can be difficult. Two respondents think that the system's advantage is easy to monitor task's process, they mention that the lack of computer's skill of nurses can be a problem. Examples of comments from participant C and D are:

> "*The current problem of my ward is not easy to manage the ward when there are the maximum admitted patients in the ward. Sometime I have to request the nurses from another ward to help to handle the tasks. The nurses need to know the specific instruction to do their tasks. The DPWFM system demonstrates that nurses are able to complete the tasks by following the Kardex activities that are arranged in sequence. It also benefits the new trainee nurses and casual nurses. A disadvantage is the DPWFM system is deployed on the computerised system, but the ward has only a few computers located in the nurse station. So nurses have to write information in their nurse notes and then key-in the same information into the DPWFM system that could take times and duplication.*" (Response from participation C)

> "*I think the advantage of the DPWFM system is able to check and monitor the nurses' activities. That will help us to have reports as evidences when we have to conduct quality control by the health department. The only one disadvantage is*

*the nurses do not frequently use the computer, so it will difficult for them when they have to use the DPWFM system on the Internet.*" (Response from participation D)

In summary, all nurses are comfortable in using the DPWFM system and feel comfortable to introduce the DPWFM to other nurses. The majorities of nurses agree that the DPWFM system will be able to reduce instruction and interpretation errors by knowing what the next task to be done. Three nurses mentioned that the system is helpful for their works and it is easy to input and access information. All nine respondents think the DPWFM system can improve their work performances. However, five participants are in the middle-aged nurses, who are not familiar with using the computer system, they think the system may increase their stressfulness.

### 6. 3 Conclusion

Based on the evaluation results, we have found that nurses in Bangkok feel comfortable to use the DPWFM system in their works (IPD Staffs 2010, Prinyapol et al. 2010). A majority of nurses think that the DPWFM system is useful and reduce interpretation errors. They also found that the system could improve efficiency in carrying out their daily works due to the reasons that the Kardex activities can provide helpful guideline, especially, during the busy periods.

The evaluation results show that nurses require simple and friendly user interface design to help them handle the constantly changing workloads and to improve patient documentation process. Due to the reason that current majority medical nurses are not skilful in using computers (IPD Staffs 2010). The user interface of the DPWFM system has been designed to use checkboxes and list boxes so that it can be used by the nurses.

In summary, nurses' survey responded the positively to the DPWFM system as it is easy to monitor the individual nurse's activities. The nurse supervisors think that it helps to manage their ward and can access to monitor the progress of tasks of each nurse. The supervisors and head nurses also agree that the DPWFM system can help to allocate assignments of tasks to the nurses.

# CHAPTER 7: CONCLUSION

This chapter concludes the presentation of this thesis. The chapter is organised as follows. Section 1 presents research findings, section 2 discusses research contribution, and section 3 proposes future research direction.

## 7. 1 Research Findings

As outlined in chapter 1, this research has the following three research objectives:

In this research, we have found that the nursing workflow is fairly complicated due to the complex and dynamic nature of nursing care processes. Each hospital has different patient care procedures in their workflow processes, however the majority of information flow and communication media are similar. In addition to nursing care, the nurses are required to be vigilant in record keeping and patient documentation. Currently, the record keeping processes are still performed manually and in paper-based form. Our observations of the hospitals in New South Wales and Thailand show that nurses working in the ward are mainly in the middle-aged group and they are often not familiar with using computerised system in their works. Therefore a system that has easy-to-use interface is desired.

We have found that WFM system integrated with web services technology can improve flexibility and dynamic interoperability in managing hospital ward workflow system. Research on practical applications of web services for hospital ward has not been thoroughly investigated. In this research, we have proposed a DPWFM framework based on web service technologies to manage workflow in a hospital ward workflow. The proposed DPWFM has two main features of dynamic recompilation and customisation to support dynamic nature of workflow in a hospital ward and to allow customisation to suit the individual needs of the nurses and wards. There are four components in the DPWFM model: work profile service (WPS), function service (FS), function allocation service (FAS) and recompilation service (RS). The WPS defines organisational role of nurses in the ward, the FS describes activities and function tasks to be performed and the FAS describes allocation and assignment of tasks. The agenda workflow is the outcome of the RS that results from the dynamic recompiling operation of the WPS, FS and FAS.

We have demonstrated using a hospital scenario to show how the DPWFM works and how the supervisor can use the DPWFM system to assign and reallocate tasks to the nurses as well as how the nurses can use the DPWFM system to complete their tasks. Three architecture layers of the DPWFM have been presented. They are presentation, business logic and data layers. The presentation layer designs the user-friendly interface to interact with the nurses. The business logic layer provides the mechanism of dynamic recompilation that operates services in the four DPWFM repositories of WPS, FS, FAS and RS. The data layer consists of the DBMS and we propose to use XML to communicate with existing database.

The evaluation results show that the nurses respond positively to the use of the DPWFM system because they believe that it can help them to improve their work and allow better control of workflow and to manage nursing documentation of their ward. The implementation of the DPWFM system is advantageous for the next generation of nurses than the current generation of nurses. That is because the majority of current nurses are in the middle-aged group who are not familiar with the computerise system, especially in Thailand. However, the simple user interface design of the DPWFM system makes it easier for them to use the system in their works.

## 7. 2 Contribution

The contribution of this research is the development of a workflow management system using web services technology for hospital ward management. We have proposed an innovative method of ward workflow management system that has the advantage of customisation and recompilation. The DPWFM platform is an easy-to-use and flexible platform to improve nursing workflows and enhance efficiency in patient documentations for individual nurse in the hospital ward.

## 7. 3 Further Research

Web service technologies are able to be used in any platforms and can be deployed over the Internet. Thus, the DPWFM platform can be operated using any available devices such as PDA, mobile phone, tablet computer and iPad. In future research, we will investigate the DPWFM prototype development using different devices such as iPad, mobile phone and tablet computer.

## REFERENCES


Alonso, G, Casati, F, Kuno, H & Machiraju, V 2004, *Web Services: Concepts, Architectures and Applications*, Data-Centric Systems and Applications, Springer, New York.

ANA 2010, accessed 04/01/2010, http://www.nursingworld.org/

ANMC 2010, accessed 06/01/2010, http://www.anmc.org.au/

Antai-Otong, D 2007, *Nurse-Client Communication: A Life Span Approach*, Jones and Bartlett Publishers, Sudbury, Massachusetts.

Audit Scotland 2007, *Planning Ward Nursing -Legacy or Design?: A Follow-Up Report*, accessed 02/02/2008, http://www.audit-scotland.gov.uk/docs/health/2006/nr_070125_ward_nursing_followup.pdf

Australian Bureau of Statistics 2008, *Business Use of Information Technology (8129.0 2007-08)*, accessed 17/11/2009, http://www.abs.gov.au

Baida, Z, Gordijn, J & Omelayenko, B 2004, 'A Shared Service Terminology for Online Service Provisioning', in *Proceedings of the 6th International Conference on Electronic Commerce (ICEC2004)*, Delft, The Netherlands, 25-27 October 2004.

Baida, Z, Gordijn, J, Sale, H, Akkermans, H & Morch, AZ 2005, 'An Ontological Approach for Eliciting and Understanding Needs in e-Services', *Lecture Notes in Computer Science*, vol.3520, pp400-414.

Barry, DK 2003, *Web Services and Service-Oriented Architectures: the Savvy Manager's Guide: Your Road Map to Emerging IT*, Elsevier Science, California.

Basu, A & Blanning, RW 2000, 'A Formal Approach to Workflow Analysis', *Information Systems Research*, vol.11, no.1, p17.

Basu, A & Blanning, RW 2003, 'Synthesis and Decomposition of Processes in

Organizations', *Information Systems Research*, vol.14, no.4, p337.

Bauer, SM 2009, 'The Use of Technology in Healthcare and Primary Care', *Pennsylvania Nurse*, vol.64, no.1, p11.

Bellinaso, M 2006, *ASP.NET 2.0 Website Programming: Problem - Design - Solution*, Wiley Technology Publishing, Hoboken, New Jersey.

Berardi, D, Calvanese, D, Giacomo, GD, Hull, R & Mecella, M 2005, 'Automatic Composition of Transition-based Semantic Web Services with Messaging', in *Proceedings of the 31st International Conference on Very Large Data Bases (VLDB 2005)*, Trondheim, Norway, 30 August -2 September 2005.

Bilignaut, PJ 1999, 'Software for Primary Healthcare in a Developing Country: Background and Problem Statement', *Computers in Nursing*, vol.17, no.6, pp291-296.

Blair, A, Jensen, M, Carder, R, Mitchell, C & McNalty, E 1999, 'Staff Stress Related to Computerised Patient Documentation in the Theater Suite', in *Proceedings of the 7th National Health Informatics Conference (HIC99)*, Hobart, Tasmania, 29-31 August 1999.

Blakely, B 2001, *Streamline Your Project Work with This Workflow Process*, accessed 02/05/2007, http://articles.techrepublic.com.com/5100-10878_11-5034036.html

Brambilla, M, Ceri, S, Fraternali, P & Manolescu, I 2006, 'Process Modeling in Web Applications', *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol.15, no.4, pp360-409.

Broom, M, Lynch, M & Preece, W 2009, 'Using Online Simulation in Child Health Nurse Education', *Paediatric Nursing*, vol.21, no.8, pp32-36.

Buchan, J 1998, *Ward Management in Practice*, Churchill Livingstone, New York.

Chang, JF 2006, *Business Process Management Systems: Strategy and Implementation*, Auerbach Publications, Boca Raton, Florida.

Channabasavaiah, K, Holley, K & Edward Tuggle, J 2003, *Migrating to A Service-Oriented Architecture, Part 1*, accessed 05/06/2009, http://www-128.ibm.com/developerworks/library/ws-migratesoa

Chatterjee, S 2004, *Developing Enterprise Web Services: An Architect's Guide*, Prentice Hall PTR, Upper Saddle River, New Jersey.

Chukmol, U 2008, 'A Framework for Web Service Discovery: Service's Reuse, Quality, Evolution and User's Data Handling', in *Proceedings of the 2nd SIGMOD PhD Workshop on Innovative Database Research*, Vancouver, Canada, 13 June 2008.

Cichocki, A, Rusinkiewicz, M & Woelk, D 1998, *Workflow and Process Automation: Concepts and Technology*, Kluwer Academic Publishers, Boston.

Cormack, DF 2000, *The Research Process in Nursing*, 4th edn, BlackwellScience, Malden, Massachusetts.

Craven, RF & Hirnle, CJ 2008, *Fundamentals of Nursing: Human Health and Function*, 6th edn, Lippincott, Philadelphia.

Curtis, B, Kellner, MI & Over, J 1992, 'Process Modeling', *Communications of the ACM*, vol.35, no.9, pp75-90.

Daily, FW 2004, *Choosing Your Record-Keeping System*, accessed 25/03/2010, http://www.unclefed.com/AuthorsRow/Daily/fwdrecordsystem.html

DEST 2001, *Enrolled Nurse Education*, accessed 30/05/2009, http://www.dest.gov.au/archive/highered/eippubs/eip01_17/4.htm#3

DiCaterino, A, Larsen, K, Tang, M-H & Wang, W-L 1997, *An Introduction to Workflow Management Systems*, http://www.ctg.albany.edu/publications/reports/workflow_mgmt

Fischer, L 2003, *Workflow Handbook 2003*, Future Strategies, Lighthouse Point, Florida.

Fischer, L 2007, *2007 BPM and Workflow Handbook*, Future Strategies, Lighthouse

Point, Florida.

Gedda, R 2008, 'SA Health Gets Web-Based Patient Records: Central Records Should Avoid MisDiagnosis', weblog post, *Computerworld*, IDG Business Network, 13 March, accessed 13/03/2008, http://www.computerworld.com.au/article/208805/sa_health_gets_web-based_patient_records/

Gelinas, UJ, Sutton, SG & Fedorowicz, J 2004, *Business Processes and Information Technology*, South-Western/Thomson Learning, Cincinnati, Ohio.

Gerrish, K & Lacey, A 2010, *The Research Process in Nursing*, 6th edn, Wiley-Blackwell, London.

Ghahramani, N, Lendel, I, Haque, R & Sawruk, K 2009, 'User Satisfaction with Computerized Order Entry System and Its Effect on Workplace Level of Stress', *Journal of Medical Systems*, vol.33, no.3, pp199-205.

Goossen, WTF 2002, 'Human Interaction', in *Textbook in Health Technology Informatics: a Nursing Perspective*, J. Mantas & A. Haman, Editors, 2nd edn, IOS Press, Amsterdam, The Netherlands, vol.65, pp266-291.

Gordijn, J, Yu, E & van der Raadt, B 2006, 'e-Service Design Using i* and e3value Modeling', *IEEE Software*, vol.23, no.3, pp26-33.

Grefen, P, Pernici, B & Sánchez, G 1999, *Database Support for Workflow Management: the WIDE Project*, Kluwer Academic Publishers, Boston.

Grimes, B & Jackson, J 2006, 'Google: Enterprise Software too Difficult to Use', weblog post, *Tech Blog*, Government Computer News, 17 May, accessed 02/03/2009, http://gcn.com/Blogs/Tech-Blog/2006/05/Google-Enterprise-software-too-difficult-to-use.aspx

Gutmans, A, Bakken, SS & Rethans, D 2005, *PHP 5 Power Programming*, Prentice Hall PTR, Indianapolis, Indiana.

Hannah, KJ, Ball, M & Edwards, MJA 2006, 'Introduction to Nursing Informatics', in

*Health Informatics*, 3rd edn, Springer, New York.

Harold, ER 2003, *Processing XML with Java: A Guide to SAX, DOM, JDOM, JAXP, and TrAX*, Addison-Wesley Professional, Boston.

Health WA 2009, *Nursing and Midwifery in Western Australia*, accessed 03/07/2009, http://www.nursing.health.wa.gov.au/become/enrolled.cfm

HIC 2009, accessed 12/11/2009, http://www.hisa.org.au/hic09

Holzner, S 2004, *Sams Teach Yourself XML in 21 Days*, 3rd end, Sams, Indianapolis, Indiana.

Huber, J 2001, 'Patient Acceptance of Computerized Progress Note Documentation', in *Information Technology for the Practicing Physician*, Health Informatics, Springer, New York, pp114-117.

IBM 2001, *History-Making Components*, accessed 04/07/2009, http://www-128.ibm.com/developerworks/webservices/library/co-tmline

Inverardi, P & Tivoli, M 2007, 'A Reuse-Based Approach to the Correct and Automatic Composition of Web-Services', in *Proceedings of International Workshop on Engineering of Software Services for Pervasive Environments: in Conjunction with the 6th ESEC/FSE joint meeting*, Dubrovnik, Croatia, 4 September 2007.

IPD Staffs 2010, Research Survey: Head Nurses, *Minutes from 16/08/2010*, In-Patient Department, Bangkok Hospital, thailand.

Jablonski, S & Goetz, M 2008, 'Perspective Oriented Business Process Visualization', in *Business Process Management Workshops*, Springer, Berlin, Heidelberg, vol.4928/2008, pp144-155.

Jipiaoneta 2008, *Airline Ticket Books Flow*, accessed 20/03/2009, http://en.jipiaoneta.cn/Travel-common-sense/200812/29-138.html

Just, WP 2008, 'Information Needs and Uses of Thai Nurses: A National Sample Survey', PhD Thesis, University of North Carolina, Chapel Hill, accessed

05/01/2010, CINAHL database.

Kardex 2008, *A Dictionary of Nursing*, accessed 11/01/2010, http://www.encyclopedia.com/doc/1O62-Kardex.html

Kim, JY & Lewandrowski, K 2009, 'Point-of-Care Testing Informatics', *Clinics in Laboratory Medicine*, vol.29, no.3, pp449-461.

King, KA, Vidourek, R & Schwiebert, M 2009, 'Disordered Eating and Job Stress Among Nurses', *Journal of Nursing Management*, vol.17, no.7, pp861-869.

Kulliyyah 2009, *Process Flow to get Registered at Kulliyyah*, accessed 10/10/2009, http://www.iiu.edu.my/anr/news_files/10april09%20ProcessFlow.pdf

Kwan, MM & Balasubramanian, PR 1997, 'Dynamic Workflow Management: A Framework for Modeling Workflows', in *Proceedings of the 30th Annual Hawaii International Conference on System Sciences (HICSS-30), Vol.4*, Maui, Hawaii, 7-10 January 1997.

Liaskos, J & Mantas, J 2002, 'Nursing Information System', in *Textbook in Health Technology Informatics: A Nursing Perspective*, J. Mantas & A. Haman, Editors, 2nd edn, IOS Press, Amsterdam, the Netherlands, vol.65, p504.

Livingston, D 2002, *Essential XML for Web Professionals*, Prentice Hall PTR, Upper Saddle River, New Jersey.

Lyng, KM, Hildebrandt, T & Mukkamala, RR 2009, 'From Paper Based Clinical Practice Guidelines to Declarative Workflow Management', in *Business Process Management Workshops*, pp336-347.

Ma, S 2005, 'Workflow Analysis to Identify the Opportunities for Improving Information Management and Nurses' Work Efficiency in Palliative Care', Master Thesis, School of Information Technology and Computer Science, Faculty of Informatic, University of Wollongong.

Mahler, C, Ammenwerth, E, Wagner, A, Tautz, A, Happek, T, Hoppe, B & Eichstädter, R 2007, 'Effects of a Computer- Based Nursing Documentation System on the Quality of Nursing Documentation', *Journal of Medical Systems*, vol.31, no.4, pp274-282.

Mancini, E, Villano, U & Rak, M 2006, 'Autonomic Web Service Development with MAWes', in *Proceedings of the 20th International Conference on Advanced Information Networking and Applications (AINA'06)*, Viena, Austria, 18-20 April 2006.

Matnieseir, M 2010, *Nurse Unit Manager of the Second Floor Rehabilitation Ward*, Personal Communication, 23 March 2010.

Mbabazi, P & Cassimjee, R 2006, 'The Quality of Nursing Documentation in a Hospital in Rwanda', *Africa Journal of Nursing & Midwifery*, vol.8, no.1, pp31-42.

Middleton, M, Bradford, C, Frantzis, J, Ambler, A, Sisson, T, Montgomerie, D & Martin, J 2009, 'Paperless and Paper-based Processes in the Modern Radiotherapy Department', *Radiography*, vol.15, no.4, pp300-305.

Mol, M 2008, *Nursing Information System*, accessed 14/01/2009, http://www.zwo.nhl.nl/hbov/telemark/

Morgan, C 2007, *XML for PHP Developers Part 1: The 15-minute PHP-with-XML Starter*, accessed 20/07/2007, http://www.ibm.com/developerworks/library/x-xmlphp1.html

Motion Computing 2007, *Improving Quality of Care and Nursing Workflow: A Clinician Usability Pilot of the Motion C5 Mobile Clinical Assistant*, accessed 10/04/2009, http://www.motioncomputing.com/about/news/case_study_C5_UCSF.pdf

Murphy, J 2004, '"I Prefer Contact This Close": Perceptions of AAC by People with Motor Neurone Disease and Their Communication Partners', *AAC: Augmentative & Alternative Communication*, vol.20, no.4, pp259-271.

Newcomer, E & Lomow, G 2005, *Understanding SOA with Web Services*, Addison-Wesley Professional, USA.

Newman, C, Buckley, T, Dunn, S & Cashin, A 2009, 'Preferences for Continuing Education Through Existing Electronic Access for Australian Nurse Practitioners and Its Implication in Prescribing Potential', *Collegian*, vol.16, no.2, pp79-83.

NHS 2009, *Nurse and Midwifery*, accessed 02/06/2009, http://www.nhs.uk/

NHS Careers 2009, *Pay for Nurses and Midwives*, accessed 04/06/2009, http://www.nhscareers.nhs.uk/details/Default.aspx?Id=4

NHS Connecting for Health 2004, *The Use of Computers in Health Care Can Reduce Errors, Improve Patient Safety, and Enhance the Quality of Service - There Is Evidence*, accessed 20/09/2009, http://www.connectingforhealth.nhs.uk/newsroom/worldview/protti2

NIA 2004, *A Framework for Nursing Informatics in Australia: A Strategic Paper*, accessed 11/11/2008, http://www.hisa.org.au/nursing

Panchenko, O & Zeier, A 2008, 'Searching and Finding Concepts in Service-Oriented Enterprise Software', in *Proceedings of IEEE Symposium on Advanced Management of Information for Globalized Enterprises*, Tianjin, China, 28-29 September 2008.

Papazoglou, M 2008, *Web Services: Principles and Technology*, Prentice Hall PTR, Harlow, England.

Parker, CD & Baldwin, K 2008, 'Mobile Device Improves Documentation Workflow and Nurse Satisfaction', *Caring Newsleter*, vol.23, no.2, pp14-18.

Plimmer, B & Mason, P 2004, 'Designing an Environment for Annotating and Grading Student Assignments', in *Proceedings of OZCHI*, Wollongong, 21-24 November 2004.

Prinyapol, N, Lau, SK & Fan, J 2010, 'A Dynamic Nursing Workflow Management System: A Thailand Hospital Scenario', in *Intelligent Automation and Computer Engineering*, Lecture Notes in Electrical Engineering, Springer, vol.52, pp489-

501.

Rajeev, D, Staes, CJ, Evans, RS, Mottice, S, Rolfs, R, Samore, MH, Whitney, J, Kurzban, R & Huff, SM 2010, 'Development of an Electronic Public Health Case Report Using HL7 v2.5 to Meet Public Health Needs', *Journal of the American Medical Informatics Association*, vol.17, no.1, pp34-41.

Ram, MB, Carpenter, I & Williams, J 2009, 'Reducing Risk and Improving Quality of Patient Care in Hospital: the Contribution of Standardized Medical Records', *Clinical Rish*, vol.15, no.5, pp183-187.

Raptis, DA, Fernandes, C, Chua, W & Boulos, PB 2009, 'Electronic Software Significantly Improves Quality of Handover in a London Teaching Hospital', *Health Informatics Journal*, vol.15, no.3, pp191-198.

RecordSystem.net 2009, *Medical Panels*, JPEG, accessed 12/10/2009, http://www.recordsystems.net/medicalpanels.jpg

Reijers, HA, Limam, S & van der Aalst, WMP 2003, 'Product-Based Workflow Design', *Journal of Management Information Systems*, vol.20, no.1, pp229-262.

Riehl, M & Sterin, I 2003, *XML and Perl*, New Riders, Indianapolis, Indiana.

Schäl, T 1998, 'Workflow Management Systems for Process Organisations', in *Lecture Notes in Computer Science*, G. Goos, J. Hartmanis et al., Editors, 2nd edn, Springer-Verlag, New York, p229.

Shefter, SM 2006, 'Workflow Technology: the New Frontier: How to Overcome the Barriers and Join the Future', *Lippincott's Case Management*, vol.11, no.1, pp25-36.

Specht, DM 2008, 'The Effect of Supportive Strategies on Computer Anxiety and Computer Self-Efficacy in Nursing Students', PhD Thesis, School of Nursing, Widener University.

St Joseph's Healthcare System 2009, *Nursing Information Systems*, accessed 20/05/2009, http://www.stjosephshealth.org/index.php/why-nurses-choose-us

Street, LJ 2002, *A Guide to Patient Care Technology: A Review of Medical Equipment*, Parthenon, Boca Raton.

Studentdoc 2009, *Nursing Job Descriptions*, accessed 25/05/2009, http://www.studentdoc.com/nursing-job-description.html

Swanson, EB & Dans, E 2000, 'System Life Expectancy and the Maintenance Effort: Exploring their Equilibration', *MIS Quarterly*, vol.24, no.2, p277.

Swenson, K 1995, *The Difference Between Workflow and BPR*, accessed 20/10/2008, http://www.e-workflow.org/downloads/workflow-bpr_kswenson.pdf

Thompson, IE, Melia, KM & Boyd, KM 2001, *Nursing Ethics*, 4th edn, Churchill Livingstone, London.

TIBCO 2007, *Business Process Discovery*, accessed 05/09/2009, http://www.tibco.com/multimedia/business-process-discovery_tcm8-2401.pdf

Tucker, A 2009, 'Improved Record-keeping with Reading Handovers', *Nursing Management - UK*, vol.16, no.8, pp30-34.

Tynan, K 2010, *Nurse Unit Manager of Ground Floor Rehabilitation Ward*, Personal Communication, 24 March 2010.

van der Aalst, WMP, Hofstede, AHMt & Weske, M 2003, 'Business Process Management: A Survey', *Lecture Notes in Computer Science*, vol.2678, pp1-12.

van der Aalst, WMP & van Hee, KM 2004, *Workflow Management: Models, Methods, and Systems*, MIT Press, London

Vaughan, J 2009, *SearchSOA.com*, accessed 10/12/2009, http://searchsoa.techtarget.com/

Vaughan-Nichols, SJ 2002, 'Web Services: Beyond the Hype', *Computer*, vol.35, no.2, pp18-21.

Vila, C, Estruch, A, Siller, HR, Abellán, JV & Romero, F 2007, 'Workflow Methodology for Collaborative Design and Manufacturing', in *Cooperative Design, Visualization, and Engineering*, Springer, Berlin, Heidelberg, vol.4674/2007, pp42-49.

W3C 2000, *Simple Object Access Protocal (SOAP) 1.1*, accessed 10/05/2008, http://www.w3.org/TR/2000/NOTE-SOAP-20000508/

W3C 2001, *Web Services Description Language (WSDL) 1.1*, accessed 20/03/2007, http://www.w3.org/TR/wsdl

W3C 2004, *Web Services Architecture*, accessed 12/11/2008, http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/

W3C 2006, *Web Services Activity: History*, accessed 12/11/2008, http://www.w3.org/2002/ws/history.html

Walker, DW, Rana, OF, Huang, Y & Huang, L 2005, 'Workflow Optimisation for e-Science Applications', in *Proceedings of the 27th International Conference on Information Technology Interfaces*, Cavtat, Croatia, 20-23 June 2005.

WfMC 1995, *Workflow Management*, accessed 21/10/2009, http://www.wfmc.org

Zimmermann, O 2003, *Perspectives on Web Services: Applying SOAP, WSDL and UDDI to Real-World Projects*, Springer, London.

# APPENDIX A - PROGRAM CODES FOR PROTOTYPE

## Default.aspx

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head><title>

</title></head>
<body bgcolor="#F0F0F0">
    <form name="form1" method="post" action="Default.aspx"
onsubmit="javascript:return WebForm_OnSubmit();" id="form1">
<div>
<input type="hidden" name="__EVENTTARGET" id="__EVENTTARGET" value="" />
<input type="hidden" name="__EVENTARGUMENT" id="__EVENTARGUMENT" value="" />
<input type="hidden" name="__VIEWSTATE" id="__VIEWSTATE"
value="/wEPDwUKMjA5Mjk4NDU1NQ9kFgICAw9kFgICAQ8PFgIeBFRleHQFEzIwMTAtMDktMDIgMTY
6NTYgUE1kZGQjN+JqbyBtwbQMJGfdncKmvspnPg==" />
</div>

<script type="text/javascript">
//<![CDATA[
var theForm = document.forms['form1'];
if (!theForm) {
    theForm = document.form1;
}
function __doPostBack(eventTarget, eventArgument) {
    if (!theForm.onsubmit || (theForm.onsubmit() != false)) {
        theForm.__EVENTTARGET.value = eventTarget;
        theForm.__EVENTARGUMENT.value = eventArgument;
        theForm.submit();
    }
}
//]]>
</script>


<script
src="/WebResource.axd?d=4D1sWT17uSqHR3fGS6sV9A2&amp;t=634171790735000000"
type="text/javascript"></script>

<script src="/WebResource.axd?d=RnvJ-L9oRR0XsRaqudbwidkjhXfA2D6xkru-
h3PQLJ81&amp;t=634171790735000000" type="text/javascript"></script>
<script type="text/javascript">
//<![CDATA[
function WebForm_OnSubmit() {
if (typeof(ValidatorOnSubmit) == "function" && ValidatorOnSubmit() == false)
return false;
return true;
}
//]]>
</script>

<div>  <input type="hidden" name="__EVENTVALIDATION" id="__EVENTVALIDATION"
value="/wEWBAK954e/DgKl1bK4CQK1qbSRCwKC3IeGDJHVfowq1b3pkMDHGNet4rNv4MJ2" />
</div>
    <div>
    <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<script src="Scripts/AC_RunActiveContent.js" type="text/javascript"></script>
</head>
```

```
<body bgcolor="#F0F0F0">
<table width="800" border="0" align="center" cellpadding="0" cellspacing="0">
  <tr>
    <td><div align="right">
        <span id="lbCurrentDate">2010-09-02 16:56 PM</span>
        </div></td>
  </tr>
  <tr>
    <td> </td>
  </tr>
  <tr>
   <td align="center" bgcolor="#F0F0F0"><img src="Images/DPWFM.jpg"
width="439" height="90" /></td>
  </tr>
  <tr>
    <td align="center" bgcolor="#F0F0F0">
      <table width="60%" border="1" cellpadding="0" cellspacing="0"
bordercolor="#0066FF">
        <tr>
          <td><table width="100%" border="0" align="center"
bordercolor="#3366FF">
              <tr>
                <td colspan="2" align="right" bgcolor="#6699FF"><div
align="center">
                    <p><strong>Login</strong></p>
                </div></td>
              </tr>
              <tr>
                <td align="right"> </td>
                <td align="left"> </td>
              </tr>
              <tr>
                <td width="46%" align="right">User name</td>
                <td width="54%" align="left">
                    <input name="txtUsername" type="text" id="txtUsername" />
                    <span id="txtUsernameReq"
style="color:Red;visibility:hidden;">*</span>
                  </td>
              </tr>
              <tr>
                <td align="right">Password</td>
                <td style="text-align: left">
                    <input name="txtPassword" type="password" id="txtPassword"
/>
                    <span id="txtPasswordReq"
style="color:Red;visibility:hidden;">*</span>
                  </td>
              </tr>
              <tr>
                <td colspan="2"><label>
                  <div align="center">
                      <p>
                          <input type="submit" name="btnLogin" value="Verify
Fingerprint" onclick="javascript:WebForm_DoPostBackWithOptions(new
WebForm_PostBackOptions(&quot;btnLogin&quot;, &quot;&quot;, true,
&quot;&quot;, &quot;&quot;, false, false))" id="btnLogin" />
 </p>
                  </div>
                  </label></td>
              </tr>
          </table></td>
        </tr>
      </table>
      <p align="center">
          <span id="lbError" style="font-style: italic">Please put your index
```

```
finger on the figerprint scanner</span>
        </p></td>
  </tr>
</table>
    </div>
<script type="text/javascript">
//<![CDATA[
var Page_Validators =  new Array(document.getElementById("txtUsernameReq"),
document.getElementById("txtPasswordReq"));
//]]>
</script>

<script type="text/javascript">
//<![CDATA[
var txtUsernameReq = document.all ? document.all["txtUsernameReq"] :
document.getElementById("txtUsernameReq");
txtUsernameReq.controltovalidate = "txtUsername";
txtUsernameReq.focusOnError = "t";
txtUsernameReq.errormessage = "*";
txtUsernameReq.evaluationfunction = "RequiredFieldValidatorEvaluateIsValid";
txtUsernameReq.initialvalue = "";
var txtPasswordReq = document.all ? document.all["txtPasswordReq"] :
document.getElementById("txtPasswordReq");
txtPasswordReq.controltovalidate = "txtPassword";
txtPasswordReq.focusOnError = "t";
txtPasswordReq.errormessage = "*";
txtPasswordReq.evaluationfunction = "RequiredFieldValidatorEvaluateIsValid";
txtPasswordReq.initialvalue = "";
//]]>
</script>

<script type="text/javascript">
//<![CDATA[

var Page_ValidationActive = false;
if (typeof(ValidatorOnLoad) == "function") {
    ValidatorOnLoad();
}

function ValidatorOnSubmit() {
    if (Page_ValidationActive) {
        return ValidatorCommonOnSubmit();
    }
    else {
        return true;
    }
}
        //]]>
</script>
</form>
</body>
</html>
```

## Default.aspx.vb

```
Imports MySql.Data.MySqlClient
Imports UserBLL
Partial Class _Default
    Inherits System.Web.UI.Page

    Public Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Me.Load
        lbCurrentDate.Text = Date.Now.ToString("yyyy-MM-dd HH:mm tt")
    End Sub

    Public Sub btnLogin_Click(ByVal sender As Object, ByVal e As
System.EventArgs) Handles btnLogin.Click
        Dim userBLL As New UserBLL

        If userBLL.IsValidUser(txtUsername.Text.Trim(),
txtPassword.Text.Trim()) = False Then
            lbError.Text = "Invalid User Account"
            lbError.ForeColor = Drawing.Color.Red
        Else
            ' Session Username
            Session("Fullname") = userBLL.GetFullname(txtUsername.Text.Trim(),
txtPassword.Text.Trim())
            Session("Level") = userBLL.GetLevel(txtUsername.Text.Trim(),
txtPassword.Text.Trim())

            ' Redirect to Main Page
            If Session("Fullname").ToString() <> "" And
Session("Level").ToString() <> "" Then
                Response.Redirect("AssignFunctionServices.aspx")
            End If
        End If
    End Sub
End Class
```

## UserBLL.vb

```
Imports Microsoft.VisualBasic
Imports System.Data
Imports MySql.Data

<ComponentModel.DataObject()> _
Public Class UserBLL
    Public obj As New UserDAL
    Public dt As New DataTable
    Public dr As DataRow

    Public Function IsValidUser(ByVal Username As String, ByVal Password As
String) As Boolean
        Try
            Return obj.IsValidUser(Username, Password)
        Catch ex As Exception
            Throw
        End Try
    End Function

    Public Function GetFullname(ByVal usr As String, ByVal pwd As String) As
String
        Try
            Return obj.GetFullName(usr, pwd)
        Catch ex As Exception
            Throw
        End Try
    End Function

    Public Function GetLevel(ByVal usr As String, ByVal pwd As String) As
String
        Try
            Return obj.GetLevel(usr, pwd)
        Catch ex As Exception
            Throw
        End Try
    End Function
End Class
```

## UserDAL.vb

```
Imports System
Imports System.Data
Imports System.Configuration
Imports System.Web
Imports System.Web.Security
Imports System.Web.UI
Imports System.Web.UI.WebControls
Imports System.Web.UI.WebControls.WebParts
Imports System.Web.UI.HtmlControls
Imports MySql.Data.MySqlClient

Public Class UserDAL
    Public conStr As String =
ConfigurationManager.ConnectionStrings("conect_host").ToString()

    Public Function IsValidUser(ByVal username As String, ByVal password As
String) As Boolean
        Dim myConnection As MySqlConnection
        Dim myDataAdapter As MySqlDataAdapter
        Dim myDataSet As DataSet

        Dim strSQL As String = ""

        myConnection = New MySqlConnection(conStr)
        myConnection = New MySqlConnection(conStr)
        strSQL = "SELECT   *  FROM Login WHERE   (UserName ='" & username &
"') AND (Pass ='" & password & "')"

        myDataAdapter = New MySqlDataAdapter(strSQL, myConnection)
        myDataSet = New DataSet()
        myDataAdapter.Fill(myDataSet, "mytable")

        Dim dt As New DataTable
        dt = myDataSet.Tables("mytable")
        If dt.Rows.Count > 0 Then
            Return True
        Else
            Return False
        End If
    End Function

    Public Function GetFullName(ByVal username As String, ByVal password As
String) As String
        Dim Fullname As String = ""
        Dim myConnection As MySqlConnection
        Dim da As MySqlDataAdapter
        Dim ds As DataSet
        Dim strSQL As String = ""

        myConnection = New MySqlConnection(conStr)
        myConnection = New MySqlConnection(conStr)
        strSQL = "SELECT   Description  FROM Login WHERE   (UserName ='" &
username & "') AND (Pass ='" & password & "')"

        da = New MySqlDataAdapter(strSQL, myConnection)
        ds = New DataSet()
        da.Fill(ds, "mytable")

        Dim dt As New DataTable
        dt = ds.Tables("mytable")
        If dt.Rows.Count > 0 Then
            Dim dr As DataRow = dt.Rows(0)
            Fullname = dr.Item("Description").ToString()
        End If
```

```
        Return Fullname
    End Function


    Public Function GetLevel(ByVal username As String, ByVal password As
String) As String
        Dim Level As String = ""
        Dim myConnection As MySqlConnection
        Dim da As MySqlDataAdapter
        Dim ds As DataSet
        Dim strSQL As String = ""

        myConnection = New MySqlConnection(conStr)
        myConnection = New MySqlConnection(conStr)
        strSQL = "SELECT   *  FROM Login WHERE   (UserName ='" & username &
"') AND (Pass ='" & password & "')"

        da = New MySqlDataAdapter(strSQL, myConnection)
        ds = New DataSet()
        da.Fill(ds, "mytable")

        Dim dt As New DataTable
        dt = ds.Tables("mytable")
        If dt.Rows.Count > 0 Then
            Dim dr As DataRow = dt.Rows(0)
            Level = Left(dr.Item("WorkerID").ToString(), 1).ToUpper()
        End If
        Return Level
    End Function
End Class
```

# APPENDIX B – MYSQL SCRIPTS

```
-- MySQL Administrator dump 1.4
--
-- ------------------------------------------------------
-- Server version   5.1.48


/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;

/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;

--
-- Create schema ward
--
CREATE DATABASE IF NOT EXISTS ward;
USE ward;
--
-- Definition of table `ward`.`Assign`
--

DROP TABLE IF EXISTS `ward`.`Assign`;
CREATE TABLE  `ward`.`Assign` (
  `WorkerID` varchar(6) DEFAULT NULL,
  `TaskID1` varchar(6) DEFAULT NULL,
  `TaskID2` varchar(6) DEFAULT NULL,
  `TaskID3` varchar(6) DEFAULT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

DROP TABLE IF EXISTS `ward`.`BedInfo`;
CREATE TABLE  `ward`.`BedInfo` (
  `BedID` varchar(5) CHARACTER SET latin1 NOT NULL,
  `WardID` varchar(5) DEFAULT NULL,
  `Building` varchar(50) DEFAULT NULL,
  `Status` varchar(15) DEFAULT NULL,
  PRIMARY KEY (`BedID`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

DROP TABLE IF EXISTS `ward`.`DynamicTask`;
CREATE TABLE  `ward`.`DynamicTask` (
  `TaskID` varchar(6) NOT NULL,
  `PatientID` varchar(5) DEFAULT NULL,
  `SuperID` varchar(5) DEFAULT NULL,
  `NurseID` varchar(5) DEFAULT NULL,
  `Remark` varchar(255) DEFAULT NULL,
  `Status` varchar(15) DEFAULT NULL,
  PRIMARY KEY (`TaskID`) USING BTREE
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

DROP TABLE IF EXISTS `ward`.`FunctionDescription`;
CREATE TABLE  `ward`.`FunctionDescription` (
  `FunctionID` varchar(8) NOT NULL COMMENT 'Function Services ID',
  `FunctionName` varchar(55) DEFAULT NULL COMMENT 'Function Service Name',
  `Description` varchar(255) DEFAULT NULL,
  `Predec` varchar(6) DEFAULT NULL,
  `Logical` varchar(6) DEFAULT NULL,
  `TaskLevel` varchar(2) DEFAULT NULL,
```

```
  `Note` varchar(255) DEFAULT NULL,
  `PrioritySeq` varchar(6) NOT NULL,
  PRIMARY KEY (`FunctionID`) USING BTREE
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

DROP TABLE IF EXISTS `ward`.`Login`;
CREATE TABLE  `ward`.`Login` (
  `WorkerID` varchar(6) NOT NULL DEFAULT '',
  `UserName` varchar(20) DEFAULT NULL,
  `Pass` varchar(20) DEFAULT NULL,
  `Description` varchar(255) DEFAULT NULL,
  `SercurityLevel` varchar(6) DEFAULT NULL,
  `PermissionLevel` varchar(6) DEFAULT NULL,
  `usertype` varchar(2) DEFAULT NULL,
  PRIMARY KEY (`WorkerID`),
  UNIQUE KEY `id` (`WorkerID`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

DROP TABLE IF EXISTS `ward`.`Mentor`;
CREATE TABLE  `ward`.`Mentor` (
  `NurseID` varchar(5) DEFAULT NULL,
  `MentorID` varchar(5) DEFAULT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

DROP TABLE IF EXISTS `ward`.`NurseInfo`;
CREATE TABLE  `ward`.`NurseInfo` (
  `WorkerID` varchar(5) NOT NULL DEFAULT '',
  `Name` varchar(20) DEFAULT NULL,
  `Lastname` varchar(20) DEFAULT NULL,
  `StartWorkDate` varchar(20) DEFAULT NULL,
  `Speciality` varchar(50) DEFAULT NULL,
  `Position` varchar(50) DEFAULT NULL,
  `SecurityLevel` varchar(3) DEFAULT NULL,
  `Status` varchar(15) DEFAULT NULL,
  PRIMARY KEY (`WorkerID`),
  UNIQUE KEY `id` (`WorkerID`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

DROP TABLE IF EXISTS `ward`.`NurseShiftSchd`;
CREATE TABLE  `ward`.`NurseShiftSchd` (
  `WorkerID` varchar(5) DEFAULT NULL,
  `ScheduleID` varchar(5) NOT NULL,
  `theDate` varchar(10) DEFAULT NULL,
  `Weekday` varchar(10) DEFAULT NULL,
  `WhichShift` varchar(2) DEFAULT NULL,
  `StartTime` varchar(5) DEFAULT NULL,
  `EndTime` varchar(5) DEFAULT NULL,
  `WhatRole` varchar(5) DEFAULT NULL,
  `WardID` varchar(5) DEFAULT NULL,
  `TAssignRecID` varchar(3) DEFAULT NULL,
  `Comments` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`ScheduleID`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

DROP TABLE IF EXISTS `ward`.`PatientProfile`;
CREATE TABLE  `ward`.`PatientProfile` (
  `PatientID` varchar(5) CHARACTER SET latin1 COLLATE latin1_bin NOT NULL,
  `Name` varchar(20) DEFAULT NULL,
  `Lastname` varchar(20) DEFAULT NULL,
  `CaseDesc` varchar(255) DEFAULT NULL,
  `EnrollDate` varchar(10) DEFAULT NULL,
  PRIMARY KEY (`PatientID`) USING BTREE
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

DROP TABLE IF EXISTS `ward`.`PrioritySequence`;
CREATE TABLE  `ward`.`PrioritySequence` (
```

```
  `TaskID` varchar(6) NOT NULL,
  `PrioritySeq` varchar(6) NOT NULL,
  `ServiceType` varchar(50) DEFAULT NULL,
  `ServiceSource` varchar(50) DEFAULT NULL,
  `ServiceDesc` varchar(50) DEFAULT NULL,
  `PriorityTime` varchar(25) DEFAULT NULL,
  `PriorityTask` varchar(25) DEFAULT NULL,
  `PriorityNurse` varchar(25) DEFAULT NULL,
  `PriorityConstrain` varchar(50) DEFAULT NULL,
  `Comment` varchar(255) DEFAULT NULL,
  PRIMARY KEY (`TaskID`,`PrioritySeq`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

DROP TABLE IF EXISTS `ward`.`RoleDescription`;
CREATE TABLE  `ward`.`RoleDescription` (
  `RoleID` varchar(5) NOT NULL DEFAULT '',
  `Description` varchar(255) DEFAULT NULL,
  PRIMARY KEY (`RoleID`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

DROP TABLE IF EXISTS `ward`.`RoleResponsibility`;
CREATE TABLE  `ward`.`RoleResponsibility` (
  `WorkerID` varchar(5) NOT NULL DEFAULT '',
  `RoleID1` varchar(5) DEFAULT NULL,
  `RoleID2` varchar(5) DEFAULT NULL,
  `RoleID3` varchar(5) DEFAULT NULL,
  `RoleID4` varchar(5) DEFAULT NULL,
  `RoleID5` varchar(5) DEFAULT NULL,
  PRIMARY KEY (`WorkerID`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

DROP TABLE IF EXISTS `ward`.`SubDescription`;
CREATE TABLE  `ward`.`SubDescription` (
  `SubID` varchar(6) NOT NULL,
  `SubID1` varchar(6) DEFAULT NULL,
  `SubID2` varchar(6) DEFAULT NULL,
  `SubID3` varchar(6) DEFAULT NULL,
  `SubID4` varchar(6) DEFAULT NULL,
  `SubID5` varchar(6) DEFAULT NULL,
  `SubID6` varchar(6) DEFAULT NULL,
  `SubID7` varchar(6) DEFAULT NULL,
  PRIMARY KEY (`SubID`) USING BTREE
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

DROP TABLE IF EXISTS `ward`.`SubTask`;
CREATE TABLE  `ward`.`SubTask` (
  `TaskID` varchar(6) NOT NULL DEFAULT '',
  `Time` varchar(5) DEFAULT NULL,
  `SubID` varchar(6) NOT NULL,
  `FunctionID` varchar(8) NOT NULL,
  `Description` varchar(255) DEFAULT NULL,
  `LevelID` varchar(2) DEFAULT NULL,
  `Status` varchar(15) DEFAULT NULL,
  PRIMARY KEY (`SubID`) USING BTREE
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

DROP TABLE IF EXISTS `ward`.`TaskAssignRecord`;
CREATE TABLE  `ward`.`TaskAssignRecord` (
  `TAssignRecID` varchar(3) NOT NULL,
  `PatientID` varchar(5) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT
NULL,
  `BedID` varchar(5) DEFAULT NULL,
  `AdmitDate` varchar(10) DEFAULT NULL,
  `DischargeDate` varchar(10) DEFAULT NULL,
  `CaseAssignment` varchar(5) NOT NULL COMMENT 'Nurse owner',
  PRIMARY KEY (`TAssignRecID`)
```

```
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

DROP TABLE IF EXISTS `ward`.`TaskLevel`;
CREATE TABLE  `ward`.`TaskLevel` (
  `LevelID` varchar(2) NOT NULL,
  `WorkerName` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`LevelID`) USING BTREE
) ENGINE=MyISAM DEFAULT CHARSET=utf8;

DROP TABLE IF EXISTS `ward`.`WardDescription`;
CREATE TABLE  `ward`.`WardDescription` (
  `WardID` varchar(5) NOT NULL,
  `WardName` varchar(50) DEFAULT NULL,
  `Buidling` varchar(50) DEFAULT NULL,
  `Floor` varchar(25) DEFAULT NULL,
  `Rome` varchar(25) DEFAULT NULL,
  `Note` varchar(255) DEFAULT NULL,
  PRIMARY KEY (`WardID`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

DROP TABLE IF EXISTS `ward`.`WorkerTimeTable`;
CREATE TABLE  `ward`.`WorkerTimeTable` (
  `WorkerID` varchar(5) DEFAULT NULL,
  `Description` varchar(255) DEFAULT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
```