e-Reservation Room System (Case Study: Koja Hospital Jakarta and Pelabuhan Hospital Jakarta)

As one of Prerequisite to Acquire Bachelor of Computer Science
At the Faculty of Science and Technology Syarif Hidayatullah
State Islamic University Jakarta

MUHAMMAD HANIF
Student ID :109091100008

STUDY PROGRAM INFORMATIC ENGINEERING
FACULTY OF SCIENCE AND TECHNOLOGY
UIN SYARIF HIDAYATULLAH
JAKARTA
2014 M/ 1435 H
e-Reservation Room System (Case study: Koja hospital Jakarta and Pelabuhan hospital Jakarta)

Thesis

As one of Prerequisite to Acquire Bachelor of Computer Science
At the Faculty of Science and Technology Syarif Hidayatullah

State Islamic University Jakarta

By:

MUHAMMAD HANIF

Student ID : 109091100008

Approved by,

Supervisor I,
Anif Hanifa Setianingrum, M.Si
NIP. 19740623 200901 2 005

Supervisor II,
Dewi Khairani, M.Sc
NIP. 19820522 201101 2 009

Head of the Computer Science Program,

Nurbaya, Ph.D
NIP. 19690310 199903 2 002
VALIDATION TEST

Thesis writing which the title is “e-Reservation Room System (Case Study: Koja hospital Jakarta and Pelabuhan hospital Jakarta)”, it has been tested and passed in Thesis Defence (Munaqosah) Faculty of Science and Technology, State Islamic University Jakarta on October 2014. This thesis writing as a requirement for achieving a bachelor degree of computer science at informatic engineering major.

Examiner Team,

Feri Fahrianto, M.Sc
NIP. 19800829 2011 01 1002

Siti Ummi Masruroh, M.Sc
NIP. 198208232011012013

Anif Hanifa Setianingrum, M.Si
NIDN. 0780 20340 3889997

Dewi Khatrani, M.Sc
NIP. 198205222011012009

Supervising team,

Dean of
Faculty of Science and Technology

Head of
Informatics Engineering Major

Dr. Agus Salim, M.Si
NIP. 19720816199903 1 003

Nurhayati, Ph.D
NIP. 196903161999032002
DECLARATION

I HEREBY DECLARE THAT THE THESIS TITLE E-RESERVATION ROOM SYSTEM (CASE STUDY: KOJA HOSPITAL JAKARTA AND PELABUHAN HOSPITAL JAKARTA) IS MY OWN WORK AND NEVER BEEN MADE IN OTHER UNIVERSITY. SOURCE OF INFORMATION DERICE FROM THIS TITLE IS WRITEN IN REFERENCES PAGE IN THE END OF THIS THESIS.

Jakarta, October 2014

MUHAMMAD HANIF
Student ID :10909110008
Muhammad Hanif. e-Reservation Room System (Case Study: Koja hospital and Pelabuhan hospital). Under the guidance of: Anif Hanifa Setianingrum, M.Si and Dewi Khairani, M.Sc

ABSTRACT

Jakarta Koja hospital and Pelabuhan hospital are some of hospital in Tanjung Priok, continuously growing to strive for improving service and quality. Pelabuhan hospital does not have any medical facility for patient, so Pelabuhan hospital have to send the referred patient to Koja hospital, because the facilities are available for those patient. According to the mechanism for sending referred patient from Pelabuhan hospital to Koja hospital occurs several problems, such as: searching for room information using guest book, make referral letter manually, there is no safety for referred letter, and the use of telephone to get and send information. Based on above explanation, the author interested to develop reservation systems for referred patient from Pelabuhan hospital to Koja hospital Jakarta using a web-based application hopefully it could help the reservation activity for both hospital. System development methodology used in development of system is RAD (Rapid Application Development) method, there are three stages in RAD, requirements planning, workshop design, and implementation. The design of the system conducted with the notation unified modeling language (UML). The author uses PHP for data processing and MYSQL for data storage. Final result achieved in this research is the building of online reservation room system which has advantages such as: can update empty treatment room in Koja hospital, making referred letter in digital, and safety referred letter.

Key words: Reservation room system, web-based application, RAD (Rapid Application Development)
ACKNOWLEDGMENT

Assalamu'alaikum wr. wb.

Alhamdulillahirabbil’alamin, all worship and praise be to Allah who has given the abundance of His grace and Hidayah so I can carry out and complete the thesis entitled “e-Reservation Room System (Case study: Koja hospital Jakarta and Pelabuhan hospital Jakarta)”. Blessings and greetings to Prophet Muhammad, his family, his friends, and his faithful followers until the end of time.

After the whole thesis is done, want to say a big thank you to all those who have helped whether it be motivation, guidance, moral and material, which is addressed to:

1. Mr. Agus Salim, as Dean of Faculty Science and Technology.
2. Mrs. Nurhayati,Ph.D as head Study Program of Informatics Engineering
3. Mrs. Anif Hanifa Setianingrum, M.Si as first supervisor which always gives guidance, encouragement and speen her time even though being busy.
4. Mrs. Dewi Khairani, M.Sc as second supervisor, which always provide feedback, motivation, and guidance during the preparation of this thesis.
5. Mr. Rahman, S.Kom, as head of IT in Koja hospital Jakarta that give permission and help so that this thesis process can be conducted so that this thesis can be resolved according the plan.

6. My Family, My friends that always give me support.
The author is aware that the preparation of this paper can still be developed again, therefore, the authors expect criticism and suggestions that can be delivered to the author via email muhammad.hanif91@yahoo.com. Finally, I hope this thesis in particular to the author's own and those who read it.

Wassalamu ‘alaikum wr. Wb

Jakarta, October 2014

Muhammad Hanif

Student ID : 109091100008
Table of Contents

TITLE PAGE.................................................................................................................. i
VALIDATION PAGE.......................................................................................................... ii
VALIDATION TEST PAGE.............................................................................................. iii
DECLARATION................................................................................................................ iv
ABSTRACT...................................................................................................................... v
ACKNOWLEDGMENT....................................................................................................... vi
TABLE OF CONTENT..................................................................................................... ix
LIST OF TABLE............................................................................................................... xii
LIST OF FIGURE............................................................................................................ xiv

CHAPTER I INTRODUCTION ......................................................................................... 1
  1.1 Background.............................................................................................................. 1
  1.2. Problem Statement ............................................................................................. 4
  1.3. Problem Limitation ............................................................................................ 5
  1.4. Purpose............................................................................................................... 6
  1.5. Thesis Benefit..................................................................................................... 7
    1. For the Author:.............................................................................................. 7
    2. For The Hospital: .......................................................................................... 8
    3. For University: .............................................................................................. 8
  1.6. Research Methodology....................................................................................... 8
    1.6.1. Data Collection Methodology ..................................................................... 8
    1.6.2. System Development Methodology ........................................................ 9
  1.7. Writing systematic: ............................................................................................ 9

CHAPTER II THEORY .................................................................................................. 11
  2.1. Information System........................................................................................... 11
    2.1.1. Information............................................................................................. 11
    2.1.2. System.................................................................................................... 12
2.2. Reservation ...................................................................................................... 15
  2.2.1. Reservation System .................................................................................. 15
  2.2.2. Reservation System Online ...................................................................... 16
2.3. RAD Model ...................................................................................................... 17
2.4. UML Diagrams ................................................................................................ 19
  2.4.1. Class Diagram .......................................................................................... 21
  2.4.2. Use Case Diagram .................................................................................... 22
  2.4.3. Activity Diagram ....................................................................................... 23
  2.4.4. Sequence Diagram ..................................................................................... 23
2.5. CodeIgniter ...................................................................................................... 24
2.6. Computer Web Language ................................................................................. 25
  2.6.1. HTML ....................................................................................................... 25
  2.6.2. CSS .......................................................................................................... 25
  2.6.3. JavaScript .................................................................................................. 26
  2.6.4. PHP .......................................................................................................... 26
2.7. Database System .............................................................................................. 29
  2.7.1. Database ................................................................................................... 29
  2.7.1. Database system ....................................................................................... 29
2.8. MYSQL ........................................................................................................... 29
2.9. Software of Web Design .................................................................................. 33
  2.9.1. Dreamweaver CS3 ..................................................................................... 33
  2.9.2. Adobe Photoshop CS5 .............................................................................. 34
2.10. Black Box Testing .......................................................................................... 35
2.11. Research Methodology ................................................................................... 36
  2.11.1. Data Collection Method .......................................................................... 36
Chapter III RESEARCH METHODOLOGY .............................................................. 40
  3.1 Data Collection Methodology ......................................................................... 40
    3.1.1 Literature study and related works ......................................................... 40
    3.1.2 Observation .............................................................................................. 44
    3.1.3 Interview .................................................................................................. 45
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.2 System Development Methodology</td>
<td>45</td>
</tr>
<tr>
<td>3.2.1 Requirement Planning</td>
<td>46</td>
</tr>
<tr>
<td>3.2.2 Workshop Design</td>
<td>47</td>
</tr>
<tr>
<td>3.2.3 Implementation</td>
<td>50</td>
</tr>
<tr>
<td>CHAPTER IV ANALYSIS, DESIGN, AND IMPLEMENTATION</td>
<td>53</td>
</tr>
<tr>
<td>4.1. Systems Development Methodology</td>
<td>53</td>
</tr>
<tr>
<td>4.1.1. Requirement Planning</td>
<td>53</td>
</tr>
<tr>
<td>4.2.2. RAD Design Workshop</td>
<td>63</td>
</tr>
<tr>
<td>4.2.2.1. Design System</td>
<td>63</td>
</tr>
<tr>
<td>4.2.2.2. Build System</td>
<td>163</td>
</tr>
<tr>
<td>4.3.1 Implementation</td>
<td>163</td>
</tr>
<tr>
<td>CHAPTER V CONCLUSION</td>
<td>174</td>
</tr>
<tr>
<td>5.1 SUMMARY</td>
<td>174</td>
</tr>
<tr>
<td>5.2 Recommendation</td>
<td>174</td>
</tr>
<tr>
<td>REFERENCE</td>
<td>176</td>
</tr>
</tbody>
</table>
List Of Table

Table 3.1 Related Work ................................................................. 41
Table 4.1 Identification Requirement ............................................ 61
Table 4.2 Actor Identification ....................................................... 66
Table 4.3 Usecase Definition .......................................................... 68
Table 4.4 Usecase Scenario Login .................................................. 76
Table 4.5 Usecase Scenario Logout ................................................. 76
Table 4.6 Usecase Scenario Master Rumah Sakit ......................... 78
Table 4.7 Usecase Scenario Master Perawatan .............................. 79
Table 4.8 Usecase Master Spesialis Data ....................................... 81
Table 4.9 Usecase Specification for Master Pasien ......................... 82
Table 4.10 Usecase Specification for Master Pengguna Data ............ 84
Table 4.11 Usecase Specification for Master Pasien Pelabuhan hospital Jakarta ................................................................. 86
Table 4.12 Usecase Specification for Master Pengguna Pelabuhan hospital staff ................................................................. 87
Table 4.13 Usecase Specification for Master Perawatan data Koja hospital staff ................................................................. 89
Table 4.14 Usecase Specification for Master pasien data Koja hospital Staff 90
Table 4.15 Usecase Specification for Master Pengguna Koja hospital .... 92
Table 4.16 Usecase Specification for Master Perawatan data Confirmation staff ................................................................. 93
Table 4.17 Usecase Spesification for Master Perawatan data Confirmation staff… 94
Table 4.18 Usecase Spesification Master Pengguna Confirmation staff........ 95
Table 4.19 Confirmation Patient................................................................. 139
Table 4.20 Sending Patient ....................................................................... 140
Table 4.21 Pasien ..................................................................................... 141
Table 4.22 Receive Patient ....................................................................... 144
Table 4.23 MasterSpesialis ..................................................................... 145
Table 4.24 MasterRs ............................................................................... 146
Table 4.25 Master Perawatan................................................................. 148
Table 4.26 User ....................................................................................... 149
Table 4.27 Role Administrator................................................................. 167
Table 4.28 Role Pelabuhan hospital staff............................................... 170
Table 4.29 Role Koja hospital staff......................................................... 172
Table 4.30 Role Confirmation Staff.......................................................... 174
List Of Figure

Figure 2.1 Workshop Design RAD (Kendal&Kendal 2010) ......................... 18
Figure 2.2 Diagram UML (Rosa 2011) ....................................................... 20
Figure 3.1 Mind Map ..................................................................................... 53
Figure 4.1 Process of Current System for Sending Referral Patient ........... 58
Figure 4.3 Analysis Proposed System .......................................................... 64
Figure 4.4 Detail Use Case Administrator .................................................... 74
Figure 4.5 Detail use case Pelabuhan Hospital Staff ..................................... 75
Figure 4.6 Detail Use case Koja Hospital staff: ......................................... 75
Figure 4.7 Detail Use case Confirmation Staff ............................................. 76
Figure 4.8 Activity Diagram for Login Process ........................................... 97
Figure 4.9 Activity Diagram for Logout Process .......................................... 97
Figure 4.10 Activity Diagram Master Rumah Sakit For Admin .................. 98
Figure 4.11 Activity Diagram Perawatan for Admin .................................. 100
Figure 4.12 Activity Diagram Master Spesialis for Admin ......................... 102
Figure 4.13 Activity Diagram Master Pasien for Admin ............................. 104
Figure 4.14 Activity Diagram Master Pengguna for Admin......................... 106
Figure 4.15 Activity Diagram Master Pasien For Pelabuhan Hospital Staff .. 108
Figure 4.16 Activity Diagram Master Pengguna For Pelabuhan Hospital Staff 110
Figure 4.17 Activity Diagram Master Perawatan for Koja Hospital Staff ...... 111
Figure 4.18 Activity Diagram Master Perawatan for Koja Hospital Staff ...... 113
Figure 4.19 Activity Diagram Master Pasien for Koja Hospital Staff .......... 115
Figure 4.20 Activity Diagram Master Pengguna For Koja Hospital Staff ...... 116
Figure 4.21 Activity Diagram Master Perawatan for Confirmation Staff ...... 118
Figure 4.22 Activity Diagram Master Pasien for Confirmation Staff............ 119
Figure 4.23 Sequence Diagram Master Rumah Sakit for Admin ............... 121
Figure 4.24 Sequence Diagram Master Perawatan for Admin ............... 122
Figure 4.25 Sequence Diagram Master Spesialis for Admin............... 124
Figure 4.26 Sequence Diagram Master Pasien for Admin.................. 126
Figure 4.27 Sequence Diagram Master Pengguna for Admin ................ 127
Figure 4.28 Sequence Diagram Master Pasien for Pelabuhan Hospital Staff 129
Figure 4.29 Sequence Diagram Master Pengguna for Pelabuhan Hospital Staff 130
Figure 4.30 Sequence Diagram Master Perawatan for Koja hospital Staff ... 131
Figure 4.31 Sequence Diagram Master Pasien for Koja Kospital Staff........ 132
Figure 4.32 Sequence Diagram Master Pengguna for Koja Hospital Staff ... 133
Figure 4.33 Sequence Diagram Master Perawatan for Confirmation Staff ... 134
Figure 4.34 Sequence Diagram Master Pasien for Confirmation Staff ....... 135
Figure 4.35 Sequence Diagram Master Pengguna for Confirmation Staff ... 136
Figure 4.36 Design of Class Diagram ...................................................... 137
Figure 4.37 Database Design ................................................................. 139
Figure 4.38 Interface Login Design ...................................................... 150
Figure 4.39 Interface Homepage Administrator .................................... 152
Figure 4.40 Interface Master Rumah Sakit Administrator .................... 153
Figure 4.41 Interface Perawatan Administrator .................................... 153
Figure 4.42 Interface Spesialis Administrator ...................................... 154
Figure 4.43 Interface Pasien Administrator ......................................... 155
Figure 4.44  Interface Homepage Pelabuhan Hospital Staff ......................... 156
Figure 4.45 Interface Master Pasien Pelabuhan Hospital Staff ....................... 157
Figure 4.46 Interface Master Pengguna Pelabuhan Hospital .......................... 158
Figure 4.47 Interface Homepage Koja Hospital Staff ................................... 159
Figure 4.48 Interface Mater Perawatan Koja Hospital ................................. 160
Figure 4.49 Interface Master Pengguna Koja Hospital ................................... 160
Figure 4.50 Interface Master Pasien Koja Hospital ....................................... 161
Figure 4.51 Interface Homepage Confirmation Staff ..................................... 162
Figure 4.52 Interface Master Perawatan Confirmation Staff .......................... 163
Figure 4.53 Interface Master Pengguna Confirmation Staff ........................... 163
Figure 4.49 Interface Master Perawatan Confirmation Staff .......................... 164
CHAPTER I

INTRODUCTION

1.1 Background

In the current condition, Hospital is a vital requirement for the growth of the community and society which are increasing every year. With the increasing needs of the population, it also increases the need for health services. Along with the increasing population, therefore needs of the hospital will also increase the implementation of information technology in order to adjust the growth population. Technology is a body of knowledge used to create tools, develop skills, and extract or collect materials (National Institute Of Health, 2013). Technology and information system has a very close relationship especially about computer, computer today is a human need in a variety of activities, coupled with the growing role of information technology in the world of work. With that technology implementation, then the job will be faster, easier, and more accurate.

One of the tool is the internet technology, there is a global online network that provides millions of limitless types of information system, in line with the development of modern technology, there are so many applications in information technology for reservation processed system via internet. The most popular system is used to reserve a certain room in hotel.
Jakarta Koja hospital is one of the public hospital in Tanjung Priok, which is growing and continuing to strive for improving service and quality standards. According to the Indonesian Republic Ministerial decision number 983.MENKES/SK/1992 public hospital is a hospital which gives a health and also basic in general and health education and training.

Pelabuhan hospital is a private hospital located in the Tanjung Priok Jakarta, which is managed by “Badan Usaha Pelabuhan (BPP) Tanjung Priok”, According to the Indonesian Republic Ministerial decision number 983.MENKES/SK/1992, private hospital is a hospital provide general medical services, specialist and subspecialist.

On the reservation room between one hospital to another, Using information technology becomes a problem during this time because information technology is the term that encompasses all forms of technology used to create, store, exchange, and use information. To solve that problem we need some applications which have highly accurate and precise process. Web application which has highly accurate and precise to help reservation between one hospital to another.

Because of those problems that have been described above, the service of Koja Hospital Jakarta and Pelabuhan Hospital Jakarta becomes unreliable currently and this is the main problem. The existed system in Koja Hospital and Pelabuhan Hospital uses old manual system which is needed to be solved by new system. Based
on this matter, the problems which happen in Koja Hospital Jakarta and Pelabuhan Hospital Jakarta as follows:

1. To perform reservation room activity, the patients take a long time, because the reservation staff should open reservation book then search the available room, this problem can be disturbed if the referrals patient needs emergency health service from this hospital.

2. The information is not accurate, this problem happens because the staff receptionist can forget to write in the guest book so when you want to book a room in the hospital, it is possible that the room where you already book has been booked by another patient and cause clashes room or book in the same room and in the same time with two different patient.

3. Data security is not guaranteed, because it is possible that the book of reservation disappears or damages, this problem can cause harm to Pelabuhan Hospital who did the reservation room because the required time becomes longer and makes management of the hospital can not determine the amount of income per month or per year.

To solve the problem, the author tries to develop a reservation application to referrals patients in Pelabuhan hospital and Koja hospital Jakarta using web based to improve health services for the patient. Using the website as a media information and also to do reservations room for referrals patient that will give facility and can save
the time. Hospitals do not have to spend a lot of time by calling another hospital to get information and do reservation for referrals patient.

Thus, to make a solution for the problem, the author tries to offer an object from the study titled “e-Reservation Room System (case study: Koja Hospital and Pelabuhan Hospital in Jakarta)” to solve the problems which happen in Koja Hospital jakarta and Pelabuhan hospital jakarta.

1.2. Problem Statement

According to the background in which author already explained, the author finds some problem statements that will arise according to e-Reservation Room System.

1. How to develop a system that can help the referral patients from Pelabuhan hospital to get available treatment room and can display the details empty room in the Jakarta Koja hospital?

2. How the system can guarantee the privacy of patient data that will refer patients from Pelabuhan hospital to Koja hospital?
1.3. Problem Limitation

Based on the background that has been described above, then the problem limitation can be formulated as follows:

1. This system is developed for Jakarta Koja hospital.
2. e-Reservation Room System is only for reservation activity between Koja hospital and Pelabuhan hospital.
3. Users of this system are part of the IT Jakarta Koja hospital, receptionist Koja hospital, receptionist Pelabuhan hospital, and confirmation staff.
4. IT staff of Koja hospital serves as admin on the system.
5. Koja hospital receptionist functions as a user in the system to receive hospitals for referral patients (Koja Jakarta hospital).
6. Receptionist Pelabuhan hospital serves as a user in the system for hospitals sender for referral patients (Pelabuhan Jakarta hospital).
7. Confirmation staff serves as the person to make sure referral patients get the treatment room.
8. For developing the system, the author uses UML ver 2.3 and RAD (Rapid Application Methodology) as a methodology.
9. Developing and design the system is web-based using PHP programming language ver 4.0 and MySQL 6.0 Apache Web Server ver 2.5 as a web server. Mozilla Firefox v.3.6 as the browser.
1.4. Purpose

This system is aimed to solve some troubles of the reservation room for referrals patient. The patient’s referral is used for the patient that needs emergency health service from one hospital to another hospital. Also, this system is purposed to reduce the time of using of patient’s referrals data flow. So, the system would be faster, easier, and more accurate for room reservation of referrals patient.

The objective domain of this thesis is to develop e-Reservation Room System to support activity of relationship for Koja Hospital and Pelabuhan Hospital. This thesis has some purposes:

1. This system will help referral patient from Pelabuhan hospital to get a room in Koja hospital and help reservation activity between Koja hospital and Pelabuhan hospital, because in this system there will be some informations that can makes simpler to book the room reservation activity between Koja hospital and Pelabuhan hospital.

2. This system there will be a very accurate information. Koja hospital receptionist staff can input the empty room data in the system, and the receptionist staff of the Pelabuhan hospital will be able to see the information's data room empty in Koja hospital.
3. In this system, the patient data will be safe, the system will store patient data into the database in this system, there is a unique number in the system patient registration for each referral patient to be referred. Referral patients should not be afraid to keep their data privacy when it comes to Jakarta Koja hospital.

1.5. Thesis Benefit

The benefits of Thesis are:

1. **For the Author:**

   The author can understand the mechanism of e-Reservation Room System with Web Programming and Database system, the author could work with PHP language and JavaScript to implement the knowledge which are gotten form the college, the author knows the real condition which happens in the real work and understand how to integrate and apply the existing theories with real event, get experience the different between theory in college and the existing problem and to comply one of the graduate prerequisite in UIN Syarif Hidayatullah Jakarta.
2. **For The Hospital:**

   Participating in the personnel program, author helps the hospital in providing efficient and effective system of room reservation that can build the web technology in the Hospital service, participating in the personnel program, and help the hospital that serves public health.

3. **For University:**

   This thesis may be a reference for further research and this thesis also can be the supplementary material science computer engineering at the university.

### 1.6. Research Methodology

The research methods and techniques of data collection in this research include two methods. There are data collection method and application development methodology.

#### 1.6.1. Data Collection Methodology

The method that author uses in collecting data, comprises of three ways:

- Observation
- Interview
1.6.2. System Development Methodology

Systems development method used is Rapid Application Development (RAD). RAD is an object oriented approach to the development of a system that includes the development of methods and software tools. RAD has some step, which is: Requirement planning, Workshop design and Implementation (Kendall&Kendall, 2010:238).

1.7. Writing systematic:

This thesis is divided into some chapters. Those chapters are:

CHAPTER I: INTRODUCTION

This part will explain briefly about the background, scope, objective, advantage, time, place, methodology, and the writing outline.

CHAPTER II: LITERATURE REVIEW

This chapter consists of definitions and theoretical components to support the report of the thesis. Within this chapter, explanation about the tools that were used in developing this application is also elaborated.
CHAPTER III: METHODOLOGY

This chapter will explain about methodology used in application. Including data collection and method of application development.

CHAPTER IV: IMPLEMENTATION

Chapter four contains the development process based on the methodology, including features for admin and users.

CHAPTER V: CONCLUSION

The summary of the research and possibilities of future works in developing this application is given in the last chapter.
CHAPTER II

THEORY

2.1. Information System

2.1.1. Information

Information is data that has been classified or processed to be used for decision-making process. Information system processing will process data from useless to useful information for those who receive it. (Sutabri, 2005)

According to (Hartono, 2005), information defined as data that is processed into more useful and more meaningful for those who receive it. Information is very important thing in a system. System with less information will need longer time to process.

The data is processed through an information model. The recipient will receive that information from the data that has been processed, it will be used to make decisions and take actions that will generate other data. The new data will be captured as inputs; it will pass through the model and so on. This process called as the cycle information.
Information must meet the requirements as required by the manager in order to make decision. Based on the requirement, the information must be accurate, on time, relevant and valuable.

2.1.2. System

According to (Sutabri, 2005) there are two groups of approaches in defining the system first is group that stresses to procedure and second one is the group which emphasizes the element or component. An approach that emphasizes the procedure defines the system as a network of procedures that are interconnected, gathered together to perform an activity or to complete a specific goal. Meanwhile the second approach more emphasis on defining the elements or components of the system as a collection of elements that interact to achieve a certain goal.

Systems approach that emphasizes the elements or components is more widely accepted, because in fact the system is more of a parts or subsystems (Jogiyanto, 2005).

A system has the characteristics or certain properties. The characteristics are: (Sutabri, 2005)
1. **System components**

   A system consists of a number of interacting components, which work together to form a union. The components of the system can form a subsystem. Each subsystem has the properties of a system that execute a specific function and affect the overall process.

   A system that has larger systems called as supra system.

2. **System Boundaries**

   The scope of the system is area that limits the system with external environment. These boundaries enable the system to be seen as a unity that cannot be separated.

3. **Environment Outside System**

   The outside environment is boundary that effects the system operation, can be detrimental or beneficial.

4. **System Interface**

   System interface is a medium that links the system with other subsystems. Output of a subsystem can be input for other subsystems through a connector.

5. **System Input**
Input is everything that needs to be put into the system as the material that will be further processed to produce useful output. Input can be either data transaction or instructions.

6. System output
Output is a component of the system in the form that has been processed from input.

7. System Processing
System processing is the component of the system that processes the input in order to produces output that is useful for the user.

8. System objectives and goal
A system objective is the result of any particular stage which supports achievement of goals.

9. Control
Control in the main component in the system is to keep the process in the system running normally takes accordance to previous limitations

10. Feedback
Feedback is required as a system control to check occurrence of irregularities in the system and restore the system to normal conditions.
2.2. Reservation

According to (silalahi, 2009) reservation is a reservation request to obtain something, which is do at previous time by someone using a variety ways to make sure that the reservation service is available and can be ordered.

If we talk about the reservation within reservation hospitality field can be defined as: Reservation staff is the part that has a very vital function, it is because of high to low occupancy rate is determined by the reservation staff for handling room reservations at the hotel.

2.2.1. Reservation System

According to (silalahi, 2009) reservation room system in the hotel is reservation room by the customer to the hotel staff, where the data of ordering can be processed immediately generated for customers.

Based on explanation of the system and reservation above, it can be concluded the meaning of reservation system is a set of elements that are interconnected and work together to support an booking activity or request certain business services that can be produced output quickly and corresponding.
2.2.2. Reservation System Online

According to (silalahi, 2009) reservation concept is built into two types, that is reservation system manual (offline) and the automatic reservation system (online).

2.2.2.1 Offline Reservation

Is the concept of conventional reservation where incoming customer requests received and processed by a customer service that is in the lobby area. Furthermore, the customer data stored in customer books.

2.2.2.2 Online Reservation

Is the concept of reservation that growing by utilizing information technology's role in it. Room reservation data obtained from customers, can be processed so that the required information can be directly received by the customer.

Based on the above explanation, it can be concluded that the system is an online reservation is a reservation system that can process data and reservation information quickly and efficiently, and connected in a larger network.
2.3. RAD Model

Rapid Application Development is an object-oriented approach to the development of a system that includes the development of methods and software tools (Kendall & Kendall, 2010).

1. RAD Phase

In the Rapid Application Development (RAD) there are three phases that involve the author and the user in the phase of assessment, design, and implementation.

a. Requirement Planning

In the requirements planning phase, users and analysts meet to identify objectives of the application or system and to identify information requirements arising from those objectives. This phase requires intense involvement from users at different
levels of the organization. The orientation in this phase is to solve business problems.

b. RAD Design Workshop

The RAD design workshop phase is a design and refine phase that can best be characterized as a workshop. Participation is intense and typically hands on. RAD design workshops can take place over a series of days, but extended blocks of time are useful.

During the RAD design workshop, users respond to actual working prototypes and analysts refine designed modules based on user responses. The workshop format is exciting and stimulating and there is no question that this creative endeavor can propel development forward at an accelerated rate.

c. Implementation

During the workshop, analysts are working with users intensely to design the business or nontechnical aspects of the system. As soon as these aspects are agreed upon and the systems are built and refined, the new systems or part of systems are tested and then introduced to the organization.
By this time, the RAD design workshop will have generated excitement, user ownership, and acceptance of the new application. Typically, change brought about in this manner is far less wrenching than when a system is delivered with little or no user participation.

2.4. UML Diagrams

The main building block is a UML diagram (Prabowo, 2011: 7). In the UML 2.3 consists of 13 kinds of diagrams are grouped in 3 categories. Distribution of the various categories and diagram can be seen in the figure below (Rosa, 2011: 120).
Here is a brief description of the division category:

a. The structure of diagram used to describe the static structure of the system being modeled.

b. Behavior diagram is a collection of diagrams that is used to describe the behavior of a system or series of changes occur in a system.

c. Interaction diagram is the diagrams are used to describe the interaction of the system with other and the interactions between subsystem on a system.
2.4.1. Class Diagram

Class diagram describes the structure of the system in terms of defining the classes that will be created to build the system (Rosa, 2011). Class is called attributes and methods or operations.

a. Attributes are variables owned by the class.

b. Operating or methods are functions that are owned by a class

The Classes that exist in the structure of the system must be able to perform these functions in accordance with the needs of the system. The composition of good Class structure in the Class diagram should have the following class types:

a. Class play
   It has an initial function to execute when the system starts.

b. The Class that handle the display system
   Those Classes define and set the display to the user.

c. Classes are taken from defining use case
   That Class handles the functions that must be taken from the definition of the use case.

d. Classes are taken from the data defining
   This Class is used to hold or wrap the data even become an entity that it will be taken and saved to the database.
The types of classes above can be combined with one another in accordance to good considered as long as the functions that exist on the class structure remains. The composition of the class can be added to the utility classes such as database connections, read the text file, and conformity with requirements. In defining the existing methods in the classroom need to pay attention what is called the cohesion and coupling.

Cohesion is a measure of how close the connection instructions in a method related to one another while the coupling is a measure of how close the relationship between the method of instruction with Other Critical Method of a class. Generally, the rule of a method that is made must have strong levels of cohesion and weak coupling levels.

2.4.2. Use Case Diagram

Use Case diagram is for modeling behavior of information systems that creates. Use case describes an interaction between one or more actors with the information system that will be created. Roughly cites, the use case uses to know what are the functions that exist within an information system and anyone who has the right to use it functions (Rosa, 2011).

The term of naming the names defines a possibility use case and easy to understand. There are two main things on use case definition they are; actor and use case.
a. Actors are peoples, processes, or other systems that interacts with information systems that will be made beyond information systems to be created its own, so even if the symbol is a picture of the actor, but the actor is not necessarily a person.

b. Use case functionality is provided by the system as exchanging messages unit between units or actor.

2.4.3. Activity Diagram

Activity diagram describes the workflow or the activity of a system or business process (Rosa, 2011:134). To note in here is the activity diagram describes the activity of the system is not what the actors, so the activities can be carried out by the system.

Activity diagrams are also widely used to define the following:

a. The design of the business process in which every sequence of activities described is system-defined business processes.

b. The order or grouping display of system / user interface where each activity is considered to have a display interface design.

c. The design of the activity testing would require a test case of the test that needs to be defined.

2.4.4. Sequence Diagram

Sequence diagrams illustrate the behavior of case object which describes a life with objects and messages that sends and receives among
objects. Therefore to draw a diagram of the sequence must be known the objects that involved in a use case along with the methods that the class possessed the object instantiated.

The number of sequence diagrams that should be drawn is as much a defining use case and it has its own process or all important that use case has defined the course of message interactions are covered in the diagrams so that more use case sequence that defined the sequence diagram should be made is also growing.

2.5. CodeIgniter

CodeIgniter is an open source web application framework for the PHP language. CodeIgniter has many features that make it stand out from the crowd. Unlike some other PHP frameworks you may have come across, the documentation is very thorough and complete—covering every aspect of the framework. CodeIgniter will also run in shared hosting environments as it has a very low footprint, yet it still has exceptional performance (Adam grifith, 2010).
2.6. Computer Web Language

2.6.1. HTML

HTML is short for Hyper Text Markup Language. Facilities hypertext linking is the method (link) one document to another through a text document. HTML is located in the website or webpage. Thus, a site consisting of several HTML pages or webpage. Indeed, HTML is a collection of code that is displayed in the format Tag for show in web pages. Usually there is a hyperlink on a web page that when clicked by the user's browser will display the linked document with the link (Lia Kuswayatno, 2006).

2.6.2. CSS

CSS is a stand-alone document and can be included in the HTML code or just a reference to the HTML element in defining the style to one HTML can also be used to create a new style which is called class (Sulistyawan, Rubianto, Rahmad Saleh, 2008).

CSS can change the text size, change the background color on a page, or it can also change the border color of the table, and compassion many more things that can be done by CSS. In short, CSS is used to set the order of appearance in page HTML.
2.6.3. JavaScript

JavaScript is a popular scripting language on the internet and can also work in most popular browsers such as Internet Explorer (IE), Mozilla Firefox, Netscape, and Opera. JavaScript code can be inserted into a web page using the SCRIPT tag (Andi Sunyoto, 2007).

There consist about JavaScript:

a. JavaScript was designed to add a web interaction.
b. JavaScript is a scripting language.
c. The scripting language is a lightweight programming language.
d. JavaScript contains a line of code that is run on the computer (web browser).
e. JavaScript is usually inserted (embedded) in an HTML page.
f. JavaScript is a language interpreter (which means scripts are executed without compilation).
g. Everyone can use JavaScript without paying a license.

2.6.4. PHP

(Ellie, 2007) explain that PHP is simple, fast, portable scripting language well suited for development of databases-enabled web sites. PHP was developed in 1995 and it is currently powering millions of web sites
worldwide. By May 2000, PHP 4 was released. The core of PHP 4 was entirely rewritten to improve the performance of complex web applications and improve modularity of the platform. Zeev Suraski and Andi Gutmans, the author of PHP 3 introduced the new parsing engine, called Zend engine. Zend engine is the scripting language that powers PHP today at 2010.

The latest improvement of PHP was released in July 2004, PHP 5 added a whole new object oriented model to language. The new model is based on Zend engine 2 and greatly improves PHP performance and capabilities (Ellie, 2007).

PHP is a programming script to a web server-side script, the script that makes the document HTML (HyperText Mark-Up Language) is an on-the-fly HTML document generated from an application rather than an HTML document is created using a text editor or HTML editor (Sidik, 2006).

Belong to the PHP programming language based servers (server side scripting) that mean all PHP scripts are placed on the server and interpreted by the web server first, then the translation sent to the browser client. This is different from JavaScript. JavaScript program code must be downloaded on the client computer first, and then translated by Internet browsers. Therefore, the JavaScript program code is always visible on a web page in question, if done the storage of web files. Technologically, the PHP programming
language has similarities with the language of ASP (Active Server Pages), Cold Fusion, JSP (Java Server Pages) or Perl (Suprianto, 2008).

PHP is an open source software which you can download it for free. This software can also run on a web server such as PWS (Personal Web Server), Apache, IIS, AOLserver, fhtpd, phtpd and so on. PHP is also a programming language that can develop our own as to add new functions of. PHP support services such as communication with the IMAP protocol, SNMP, NNTP, POP3, and even HTTP. PHP can be installed as a part or module of the apache web server or as a standalone CGI script. Many advantages can be obtained when using PHP as a module of apache, among which are: high levels of security, execution time is faster than other web programming languages are oriented to server-side scripting, database access to a more flexible system such as MySQL. The concept of PHP working on a similar principle with HTML code, only when the requested PHP file obtained by the web server, it immediately passed to the PHP engine and the engine is what processed and give the result (in the form of HTML code) to the web server. Furthermore, the web server delivered to the client (Abdul Kadir, 2008).
2.7. Database System

2.7.1. Database

Database is collection of data that stored or recorded in to a file or more than one file that will be mange so that can be used again for certain needed. Expectation of database is there is no duplication or redundancy of data in database (Silberschatz, 2005).

According to (Marlinda, 2004), “database is a collection of operational data from organization that manage and stored using certain method by using computer so that this circumstance can provide useful information for user”.

2.7.1. Database system

Database system is systems which arrange and manage recoded data. By using computer database system stored and record, maintenance process also doing in computer so that can provide complete and useful information for user to make decision making (Marlinda, 2004 P1).

2.8. MYSQL

Today many organizations face the double threat of increasing volumes of data and transactions coinciding with a need to reduce spending. Many such organizations are migrating to open source database management system to keep
costs down and minimize change to their existing system. The world’s most popular of these open source database system (it’s free to download, use, and modify) is MySQL. It is distributed and supported by MySQL AB, a Swedish commercial company founded by the original developers, David Axmark and Michael “Monty” Widenius, who wrote MySQL in 1995. MySQL has its roots in Msq or mini SQL, a lightweight database developed at Bond University in Australia, to provide fast access to stored data with low memory requirements. Its symbol is a dolphin called “Sakila” representing speed, power, precision, and good nature of the MySQL database and community.

As a database server which has a modern database, MySQL has many advantages:

1. **Portability**

MySQL can run in any platform such as Windows, Linux, Mac OS X Server, etc.

2. **Open Source**

We can get MySQL for free without any cost.

3. **Multiuser**
MySQL can be used for many users at the same time without any problem or any conflict.

4. Performance Tuning

MySQL has high speed in managing simple query.

5. Column Types

MySQL has a complex column types such signed / unsigned integer, float, double, char, varchar, text, blob, date time, datetime, timestamp, year, set and enum.

6. Command and Functions

MySQL has operator and function which support SELECT instruction and WHERE instruction in query.

7. Security

MySQL has some layers of security such subnet mask, host name, detail of user access permission (ask password).
8. **Scalability and Limits**

MySQL can process a database in high scales with the total records are more than 50 million and 60 thousands table with 5 trillion line. The limit of index in each table is 32 indexes.

9. **Connectivity**

MySQL can connect to the client is using TCP / IP protocol, Socket Unix or Named Pipes (NT).

10. **Localisation**

MySQL can detect error message at client using more than 20 languages.

11. **Interface**

MySQL has interface in every application and language program in which used for database administration.
12. **Client and Tools**

MySQL completed with many tools that its can used for database administration, and each tool was completed an online instruction.

13. **Table Structure**

Compare with other database, table Structure in MySQL is more flexible in processing ALTER TABLE.

2.9. **Software of Web Design**

2.9.1. **Dreamweaver CS3**

Adobe Dreamweaver CS3 is a Web development tool that lets you create dynamic, interactive Web pages containing text, image, hyperlinks, animation, sounds, videos, and others elements. You can use Dreamweaver to create individual Web pages or complex Web sites consisting of many Web pages. A Web site is a group of related Web pages that are linked to gather and share a common interface and design. You can use Dreamweaver to create design elements such as text, tables, and interactive button, or you can import elements from other software programs. You can save Dreamweaver
files in many different file format including XHTML, HTML, JavaScript, CSS, or XML, to the name a few (Sherry Bishop, 2007).

XHTML is the acronym for eXtendible Hyper Text Markup Language, the current standard language used to create Web pages. You can still use HTML (Hyper Text Markup Language) in Dream weaver, you can easily convert existing HTML code to XHTML-complaint code. You use a browser to view your Web pages on internet. A browser is a program, such as Microsoft Internet Explorer or Mozilla Firefox, that lets you display HTML-developed Web pages (Sherry Bishop, 2007).

2.9.2. Adobe Photoshop CS5

Photoshop CS5 is a popular image editing software produced by Adobe Systems Incorporate. Image editing software refers to computer programs that allow you to create and modify digital images, or pictures in electronic form. One type of digital image is a digital photograph or photo, which is picture taken with a camera and stored as a digitalized file. The photo then is converted into a print, a slide, or used in another file. Other types of digital images include scanned images or electronic form of original artwork created from scratch. Digital images are used in graphic applications, advertising, print publishing, and on the web. Personal uses include private photos, online photo sharing, scrapbooking, blogging, and social networking,
among others. Image editing software, such as Photoshop, can be used for basic adjustments such as rotating, cropping, or resizing, as well as for more advanced manipulations, such as airbrushing, retouching, photo repair, changing the contrast of an image, balancing, or combining elements of different images. Because Photoshop allows you to save multilayered, composite images and then return later to extract parts of those images, it works well for repurposing a wide variety of graphic-related files (Gary B. Shelly, Joy L. Starks, 2011).

Photoshop CS5 is part of the Adobe Creative Suite 5 and comes package with most of the suite versions. It also is sold and used independently as a stand-alone application. Photoshop CS5 is available for both the PC and Machintosh computer platforms. Photoshop CS5 and some new features for working with 3D imagery, motion-based content, and advanced images analysis (Gary B. Shelly, Joy L. Starks, 2011).

2.10. Black Box Testing

Black-box testing is testing software in terms of functionality without test design specifications and program code. Test is intended to determine whether the functions, inputs, and outputs of the software according to the required specifications (Rosa, 2011: 213). Black box testing is done by creating test cases
that tries all the software functions use whether according to the required specifications. Test cases are created to perform black box testing that should be made to the case of right and wrong cases, for example in the case of the login process the test cases is created are:

a. If the user enter a user name and password correctly.

b. If the user enter a user name and password incorrectly, for example, if the user name correctly but the password is wrong, or vice versa, or both are wrong.

2.11. Research Methodology

2.11.1. Data Collection Method

The method of data collection is systematic and standardized procedures to obtain the necessary data (Nazir, 2005). Data collection is a very important step in the scientific method because in general the data collected is used to test the hypotheses that have been formulated. Several techniques are available to collect data. Data collection techniques can be as follows:
• Library Research

Library research is a literature study that contains descriptions of the theory, findings and other research materials obtained from reference materials used as a basis for research activities. Descriptions in the literature are directed to develop a clear framework of solving the problem that has been described previously in the formulation of the problem. By conducting studies to existing literature, researchers can learn more systematically more about ways to write papers, how to express the ideas that will make the research more critical and analytical in doing his own research (Nazir, 2005).

• Field Study
  o Observation

Observation (observation) is a technique or approach to obtain primary data by directly observing the data object (Jogiyanto, 2008). Further (Nazir, 2005), defines the collection of data by direct observation is the way of data collection using the eyes without any help other standard tools for this purpose. The researchers are trying to do exploratory field by identifying all elements of the social, physical and natural conditions to assess the situation,
circumstances, background and context, more specifically the way eyes observed.

Observation is a technique wherein the systems analyst either participates in or watches a person perform activities to learn about the system. Besides, observation technique has advantages and disadvantages. So that, the writer tries to summarize both of them below:

- **Advantages:** Data gathered can be very reliable, can see exactly what is being done in complex tasks, relatively inexpensive compared with other techniques, and can do work measurements.

- **Disadvantage:** People may perform differently when being observed, work observed may not be representative of normal conditions, timing can be inconvenient, interruptions, some tasks not always performed the same way, and may observe wrong way of doing things.

- **Interviews**

  Interview is a two-way communication to get the data from respondents (Jogiyanto, 2008). In this case the questioner or the interviewer asks the questions to the informants with direct face to face. The key aspects of the interview technique of recording the
interview data, because if not done properly then the interviewer can interview data loss and business interviews will be in vain.

Interview is a technique whereby the systems analysts collect information from individuals through face-to-face interaction. Interview also is a technique that is used for find facts, verify facts, clarify facts, generate enthusiasm, get the end-user involved, solicit ideas and opinions, and identify requirements. So that, the writer try to sumarize both of them below:

- **Advantages:** Give writer opportunity to motivate interviewee to respond freely and openly, allow analyst to probe for more feedback, permit analyst to adapt or reword questions for each individual, and can observe nonverbal communication.

- **Disadvantages:** Time-consuming, success highly dependent on writer's human relations skills, and may be impractical due to location of interviewees.
Chapter III

RESEARCH METHODOLOGY

3.1 Data Collection Methodology

In the second chapter has described the literature of the elements that will be applied in this thesis, therefore the authors can conclude that in making thesis requires scientific methods that meet the scientific requirements, so that hypotheses which have been formulated can be tested in a proper way. System development methodology using RAD (Rapid Application Development) methodology to develop e-Reservation Room System at RSUD (Rumah Sakit Umum Daerah) Koja Jakarta Utara.

There are have 3 data collection technique:

3.1.1 Literature study and related works

A literature study methods by collecting data and information that serve as the floating reference for develop this application using web-based. Authors collected references from the books contained in the library and research publications, articles, Internet sites, and related works other resources related to this thesis including, systems analysis and design, system
development methods, the PHP programming language and MySQL database as a tool to develop the application.

**Table 3.1** This is explanation 3 related works:

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1)</td>
<td>Title: Perancangan dan pembuatan aplikasi reservasi hotel transit FM3 Tangerang</td>
</tr>
<tr>
<td></td>
<td>Case Study: Hotel Transit FM3 Tangerang</td>
</tr>
<tr>
<td></td>
<td>Writer: Tri Eki Budi Wijaksono (Student from Sekolah Tinggi Manajemen Informatika Dan Komputer AMIKOM Yogyakarta) 2012.</td>
</tr>
<tr>
<td></td>
<td>Description: Writers on this title case studies of research conducted on hotel reservation system on Hotel Transit FM3 Tangerang start from less effective and efficient check in and check out, reservation process and reporting process is still using manual.</td>
</tr>
<tr>
<td></td>
<td>Contribution: System development methods in use is RAD (Rapid Application Development) Tools of making an application using HTML, PHP, CSS and SQL.</td>
</tr>
<tr>
<td></td>
<td>Weakness: The system is only limited for room</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>2)</td>
<td><strong>Title</strong>: Sistem Informasi Reservasi Sewa Kamar Hotel Berbasis Web</td>
</tr>
<tr>
<td></td>
<td><strong>Case Study</strong>: Hotel Karlita Tegal</td>
</tr>
<tr>
<td></td>
<td><strong>Writer</strong>: Pepi Siswhar Ismail (Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Diponogoro, Semarang) 2011</td>
</tr>
</tbody>
</table>
|   | **Description**: The function of Reservation information system is to make it easy for the receptionist and for customer in process to reservation room in the hotel. During this time, the existing system of rents rooms in Hotel Karlita still do it by the writing in reservation book so that the system needs to be improved. Some systems are assessed needs to be improved is the system for reservation room, which is a system check-in, check-out system. Limitation Problem on this reservation system is only for manual reservation system to digital. In the title of this study, the authors use the waterfall
| Contribution | System development methods in use is Waterfall methodology and tools of making an application using HTML, PHP, CSS and SQL. |
| Weakness | In this system only information for reservation room check in and check out and in this doesn’t have capabilities for printing the letter of reservation room. |

3) **Title**: Analisis Dan Perancangan Sistem Informasi Reservasi Kamar Berbasis Web Pada Hotel Sri Wedari Yogyakarta

**Case Study**: Hotel Sri Wedari Yogyakarta

**Writer**: Agnes Irmayanti J (Sekolah Tinggi Manajemen Informatika Dan Komputer “AMIKOM” Yogyakarta) 2010

**Description**: The author analyzes the system in this title which has been running on hotels Sri Wedari Yogyakarta and the authors build a web site that can be run dynamically according to the situation so that the information can be easily accepted by society. Limitation Problem in this
paper is the information system reservation online, information about booking rooms, facilities available at Hotel Sri Wedari Yogyakarta. The author uses the waterfall method on this thesis.

| Contribution | System development methods in use is Waterfall methodology and tools of making an application using HTML, PHP, CSS and SQL. |
| Weakness | In this system only information for reservation room check in and check out and in this doesn’t have capabilities for print the letter of reservation room. |

### 3.1.2 Observation

To gather information about the needs of the system, the authors perform data collection by observation at the Koja hospital and Pelabuhan hospital. Author visited Koja hospital Jakarta and Pelabuhan hospital Jakarta to know the booking process for treatment room to referall patients from Pelabuhan hospitals. This needs to be done so that the author can do useful analysis of the the current system and to determine the design of the new system to be built in order to keep in syncronous with the existing system.
3.1.3 Interview

In this part the author did interview to Rahman S.Kom he is the head of Information Technology in Koja Hospital, he has responsibilities for system that is running in the Koja hospital. The author asked about the problem in the process to referral patient, the purpose of this step is to collect the complete information of the process to referral patient from Pelabuhan Hospital to Koja Hospital. In the several of interview with both of hospital the author asked some question which is the history of Koja hospital and history of Pelabuhan hospital, then the process of when the Pelabuhan hospital send the referral patient to the Koja hospital and how to make referral latter for make the Koja hospital believe if the referral patient is from Pelabuhan hospital.

3.2 System Development Methodology

In system development methodology writer used RAD (Rapid Application Development) methodology, according (Kendall & Kendall, 2010) RAD consists of three main part which is requirements planning, workshop design, and implementation.
3.2.1 Requirement Planning

According (Kendall & Kendall, 2010), in this phase the analysis and users meet to identify the purpose of the application or system to be develop. In this phase need intense involvement that is the writer and the both of hospital staff as a user in the system to be built. In this phase researcher do some interviews and observation with both of hospital who has relation with the system that will be build, after that has some identification which is:

1. Analysis of current system on the Koja hospital staff as a receiver referral patient and Pelabuhan hospitals as a sender of referral patient.

2. Identify the problem. In this phase, the researcher found several problems that occurred in the both of hospital staff.

3. Analyze requirements system. In this phase, the researcher can analyze the needs of the system based on the identification problem. The needs of the system to be build which is:

   a. The system can help or facilitate Pelabuhan hospital staff to send referral patient and Koja hospital to receive referral patients.

   b. The system can make easier to input the available room in the koja hospital and make sure referral patients get the treatment room.

   c. Can make referral patient digitally.

4. Determine the purpose of develop the system. The purpose of the system that obtained in the systems development is to be the controller
for send and receive referral patient, which aims to facilitate both hospital staff in the process of sending and receiving referral patients more systematic, so that the service and process to give information can be implemented quickly and precisely.

### 3.2.2 Workshop Design

As the declaration by (Kendall & Kendall, 2010) the meaning of workshop design phase in the RAD methodology is the phase of designing and fixing application or system, Kendall & Kendall characterize this phase as a workshop. Why this phase as a workshop, because there is like a workshop that participants who participated strongly in the group and there is no a passive activity. Participants are analogy as small groups that set up to help the user in approving the design. During the RAD design workshop, the user responds working prototype that already exist, analysis and fix the modules that designed using the software based on user responses. Therefore, at this stage is divided into two parts which is:

#### 3.2.2.1 Design System

In the design of the system will do the designing processes that will occur in the system to be represented by the UML diagram to allow users and developers to better understand the initial steps to
build a physical system. The design of applications that researchers do is:

1. Usecase

In the design of the use case, researchers divided into three parts which is:

a) Actor Identification

Identify the actors do to be able to know who the user can use the system. In this case, researchers determined four user who will directly interact in the system, the user who will interact in the system is administrator (Head IT Koja hospital), Pelabuhan hospital staff, Koja hospital staff, and confirmation staff.

b) Usecase Diagram Design

Use case diagrams describe the interaction between actors in the information system are made.

c) Usecase Scenario

Use case scenario is a table that describes the use case that already exist.
2. Activity Diagram

Activity diagram design a workflows for a sequence of activities in a process. This diagram is very similar like flowchart because we can model the process logic, business processes and workflow.

3. Sequence Diagram

Sequence diagrams describe interactions between objects in and around the system (including the user, displays, and so on) describe like a message that is described with a time.

4. Database Design

In the database system, researcher divide into two type, which is:

a. Class Diagram

Class diagram describe about structure system from class definition that will be build.

b. Translation of data classes in the class diagram into the database. Translating the data classes in the class diagram in the form of tables and their fields and their relation to the database.
c. Interface design

The interface design give a description of the display system to be used as e-Reservation Room System.

3.2.2.2. Build System

In this step the output of the system design is development of applications that has been designed previously using PHP and CSS, for application server using XAMPP Package for Windows versi 1.7.4 and MySQL 6.0 for the database in the system.

3.2.3 Implementation

As described by (Kendall & Kendall, 2010) implementation phase is implemented based on the description the previous phase. During the design workshop, the analyzer and users work together intensively to design business process or aspects of non-technical applications. After every aspect are agreed and the systems that has been built and repaired, the new application was tested and shown to the organization to get a response.

In this step have some phase, which is:
1. Installation

Explain about the process of installation the system which aims to see how the application is work. In this step do some phase, which is:

a. The specification of hardware that should be used

The specification of hardware that should be used means is the minimum specification should be have (Personal Computer) which will implement this application in both of hospital staff.

b. Software support

Explaining the workings of the software to support e-Reservation Room System

2. Testing the system

Explain about the process of testing the system which aims to see how the application is work. In this step do some phase, which is:

a. The testing is used black box testing, means of the testing is use laptop. For this testing researcher using laptop/notebook Asus A83S.

b. User response for the application, in this step means reaction of the user who use the system. The user in this system is receptionist staff both hospital and IT staff in Koja hospital.
Figure 3.1 Mind Map

- Research
  - Data Collection Methodology
  - Observation
  - Interview
- System Development Methodology
  - Rapid Application Development Methodology (Kendall & Kendall, 2010)
- Requirement Planning
  - Analysis of Current System
  - Identify The Problem
  - Analysis Requirement System
  - Determine Purpose Of Develop System
- RAD Design Workshop
  - Use case
  - Activity Diagram
  - Sequence Diagram
  - Interface Design
- Build System
- Implementation
  - Installation
  - Testing (Blackbox)
CHAPTER IV
ANALYSIS, DESIGN AND IMPLEMENTATION

4.1. Systems Development Methodology

In the RAD (Rapid Application Development) methodology has several steps, which is:

4.1.1. Requirement Planning

4.1.1.1. Analysis of Current System

Based on observations and interviews with both hospital staff, the author found some problems while sending refer patients still using the manual method, here is a referral system that is running between the Koja hospital and the Port of Pelabuhan hospital:

- Patient came to Pelabuhan Hospital and do the registration process and then the patient will receive the patient card.

- After doing registration, patient have to report to nurse that responsible in for each doctor.

- After getting treatment from the doctor concerned, the doctor decides that the patient requires hospitalization where necessary facilities cannot be met by Pelabuhan hospitals, then the doctor
will give advice to immediately refer to Koja hospital and doctors
give suggestion letter to the patient.
- After the patient gets a referral later from a doctor, the next
  procedure is the patient must come to the receptionist staff who
  will be the liaison between the Pelabuhan hospital (hospital
  partners) and Koja hospital.
- Then, reception staff in Pelabuhan Hospital responsible for
  contacting the staff of Koja hospital to check the availability of
  rooms at Koja hospital and to confirm to the staff of the port
  hospital (hospital partners).
- After Pelabuhan hospital staff get confirmation from Koja hospital
  about the availability of rooms, then, if the room is available, the
  staff will give you a referral letter for the patient to be taken to
  Koja hospital.
- Then, refer the patient came to the Koja hospital to be verified in
  registration department, for further processing by a nurse in order
  to get a room.

As an illustration groove that runs the system can be described in Figure 4.1:
Figure 4.1 Process of Current System for Sending Referral Patient

4.1.1.2. Identify Problem

Based on analysis of the current system above, can be concluded that the activities conducted by Pelabuhan hospital and Koja hospital still using manual activities to referred patient, so there is a possibility of the problems as follows:
1. Reservation time that it takes quite a long time, because the receptionist staff of Koja hospital should open a book and look for available rooms.

2. Information is not accurate, because there is the possibility of receptionist staff Koja hospital may forget to record the information on the guest book treatment room empty or not, so if there are any patient who will be referred to Koja hospital but the information not accurate because of still manually input the data using the guest book.

3. Pelabuhan hospital staff still write by hand when made a referral letter.

4. When the referred patient has come to Koja hospital, patient are not directed to the room that already booked and the patient are not necessarily getting a room because of process in Koja hospital does not have guaranty if the referred patient getting the available room.

5. The security of the data is not safe:

   - The referral letter that referred patient brought there is not have patient number (a unique number), it possible that the letter could have been manipulated or abused by people who are not responsible in this process.
- Guest book in Koja hospital maybe damaged or missing. This can be the problem if the important information listed in the guest book so hard to find until Koja hospital staff does not have important information about inpatient room in Koja hospital, if the data is in the guest book are still using this way can cause problems by people who are not responsible.

In accordance with the problem identification, authors propose a system that will be able to make bookings for systematic referring patient becomes easier not take a long time and can be accessed as needed.

4.1.1.3. Analyze Requirements System

Based on the problem identification, Koja hospital and Pelabuhan hospital need an application that can provide easy to reservation room and made a referral letter for referral patient. Applications that are built can make easy for both receptionist staff, manage the patient data and also for booking the available room in Koja hospital for referral patient. Where an application must be able to update the empty rooms in Koja hospital and the application can make referral letters quickly and have the security of data, so in practice the system is able to improve the performance of the staff receptionist both hospital in order to more effectively and efficiently.
The strategy to determine the solutions of the above problems by means of:

1. Table identification requirements system

**Table 4.1** Identification requirement system

<table>
<thead>
<tr>
<th>NO</th>
<th>Title</th>
<th>Benchmark</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Search treatment rooms for referral patient.</td>
<td>Search rooms to referral patient still using manual or using guestbook.</td>
<td>Simplify for finding an empty room.</td>
</tr>
<tr>
<td>2</td>
<td>Input the patient data is still manual.</td>
<td>The possibility the information is inaccurate and unreadable.</td>
<td>Easy to input the patient data.</td>
</tr>
<tr>
<td>3</td>
<td>Made a referral letter is manual.</td>
<td>Made a referral letter is manual from Pelabuhan hospital.</td>
<td>Easy to make a referral letter.</td>
</tr>
<tr>
<td>4</td>
<td>Identification referral letter.</td>
<td>The letter comes just released from the Pelabuhan hospital receptionist staff.</td>
<td>There is any unique code in the referral letter.</td>
</tr>
</tbody>
</table>
4.1.1.4. Determine Purpose of Develop System

Because referral patient are patient who really need extra health facilities, the construction of the system has a purpose to help and facilitate the receptionist staff both hospital for reservation room process to be fast, precise and accurate.

Several functions contained in the system:

1. Make it easy for Pelabuhan hospital staff to see the update available rooms at Koja hospital.
2. Make it easy for Pelabuhan hospital staff to make the referral letters.
3. Make it easy Koja hospital staff to update the available room.
4. There is the security of data in each hospital reservation room process.
5. Ensure referral patient get the room treatment that has been booking in the system.

4.1.1.5 Analysis of Proposed System

The process of proposed a new system is more focused on how the receptionist staff both hospitals that can access the new system, for staff receptionist Pelabuhan hospital should know how to inputting
patient data digitally using computer and website that made by , for staff receptionist Koja hospital should know about input the data available room and know about how to see the patient data that already send by Pelabuhan hospital, after that the staff receptionist Koja hospital has responsibilities to confirmation or not referral patient in the application. Beginning of the process that must be done is to input the user who can access the application. Remember bookings treatment room is a very important part for the hospital because it concerned for the health of the patient who enter the information and who has access to the data should be clear and accurate, Because the data that has been entered can be responsible for both hospital staff.
Below is a proposed system that will run between Pelabuhan hospital and Koja hospital:

- Patient came to Pelabuhan hospital to registration in order to get patient card.
- Then, patient have to report to nurse that responsible for each doctor.

- After get treatment from doctor, the doctor decided that patient need hospitalization where necessary facilities cannot be met by Pelabuhan hospitals, then the doctor will give advice to immediately refer to Koja hospital and doctor gave a suggestion letter to the patient.

- Then refer patients come to the Pelabuhan hospital reception and provide a letter from the doctor's suggestion. Then, the receptionist saw an empty hospital room through e-Reservation Room System if there is an empty room directly input the data front and reprinting letters refer patients through the system.

- Refer Patients come to the Koja hospital and register in the receptionist with a letter of reference. After giving the letter refer to staff receptionist, then check the refer letters to be verified in the system, after getting approval from the receptionist refer patients refer to a letter of confirmation to be sure staff get a hospital room.
4.2.2. RAD Design Workshop

In the design RAD workshop (Rapid Application Development) there are several steps, which are:

4.2.2.1. Design System

1. Use case

In making use case, authors divided into 4 parts in its design, namely:

a) Actor Identification

Here is a description of the defining actors in the application e-Reservation Room System:

<table>
<thead>
<tr>
<th>No</th>
<th>Actor</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Administrator</td>
<td>People who set up users who can use the system, enter the name of the hospital, enter the type of treatment room, and maintain data security.</td>
</tr>
<tr>
<td>2</td>
<td>Pelabuhan Hospital Staff</td>
<td>People who use the facility of the system to input the data such as registration information.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>3</td>
<td>Koja Hospital Staff</td>
<td>People who update treatment rooms that are empty or not, people who see the referral patient data, and people who confirm referral patient that already arrived to the Koja hospital.</td>
</tr>
<tr>
<td>4</td>
<td>Confirmation Staff</td>
<td>People who make sure referral patient get treatment room in Koja hospital (nurse, and etc).</td>
</tr>
</tbody>
</table>

b) Usecase Definition

Here is a description of the defining usecase in the application e-Reservation Room System:
<table>
<thead>
<tr>
<th>NO</th>
<th>Use Case</th>
<th>Actor</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Login</td>
<td>All actor</td>
<td>It is the process of checking permissions, login mandatory for all users that connected with this system, and login are also required for every functions that related to change access to the database,</td>
</tr>
<tr>
<td>2</td>
<td>Logout</td>
<td>All actor</td>
<td>It is the process of logout for all of the user that connected with this system, and also logout to close all access that connected with the changing the data to the database.</td>
</tr>
<tr>
<td>3</td>
<td>Master Rumah Sakit</td>
<td>Administrator</td>
<td>Is the process of arranging the names of hospitals that contribute in this system. In the Master rumah sakit there are three master management process which is the process of input rumah sakit’s data, edit rumah sakit’s data and delete rumah sakit’s data that has contribute in this</td>
</tr>
<tr>
<td>No</td>
<td>Master</td>
<td>Administrator</td>
<td>Description</td>
</tr>
<tr>
<td>----</td>
<td>-------------------------</td>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>4</td>
<td>Master Perawatan</td>
<td>Administrator</td>
<td>Is a process to set the treatment room in koja hospital. In the master perawatan management there are three processes which is input perawatan’s data, edit perawatan’s data, and delete perawatan’s data.</td>
</tr>
<tr>
<td>5</td>
<td>Master Spesialis</td>
<td>Administrator</td>
<td>Is a process to set the type of specialist facility that koja hospital have. In the master spesialis management there are three processes which is input specialist, edit specialist, and delete specialist data.</td>
</tr>
<tr>
<td>6</td>
<td>Master Pasien</td>
<td>Administrator</td>
<td>Is a process to see the patient data that already send by Pelabuhan hospital. In the master pasien management there are three processes which is check pasien’s data and confirmation arrival patient, print referral letter and delete patient data.</td>
</tr>
<tr>
<td>7</td>
<td>Master Pengguna</td>
<td>Administrator</td>
<td>Is a process to set the user in the system. In the master pengguna management</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>67</td>
<td></td>
<td>there are three processes which is input pengguna’s data, edit pengguna’s data, and delete pengguna’s data of this system.</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Master Pasien</td>
<td>Pelabuhan Hospital Staff</td>
<td>Is a process to manage the data of referral patient. In the master pasien management that have some function which is input pasien’s data, check pasien’s data, print referral letter and delete pasien’s data.</td>
</tr>
<tr>
<td>9</td>
<td>Master Pengguna</td>
<td>Pelabuhan Hospital Staff</td>
<td>Is a process to set the user in the system. In the master pengguna management there are only one processes which is edit pengguna’s data.</td>
</tr>
<tr>
<td>10</td>
<td>Master Perawatan</td>
<td>Koja Hospital Staff</td>
<td>Is a process to set the treatment room. In the master perawatan management there are three processes which is input perawatan’s data, edit perawatan’s data, and delete perawatan’s data.</td>
</tr>
<tr>
<td>11</td>
<td>Master Pasien</td>
<td>Koja Hospital Staff</td>
<td>Is a process to see the patient data that already send by Pelabuhan hospital. In</td>
</tr>
<tr>
<td>No</td>
<td>Module</td>
<td>Role</td>
<td>Description</td>
</tr>
<tr>
<td>----</td>
<td>----------------</td>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>12</td>
<td>Master Pengguna</td>
<td>Koja Hospital Staff</td>
<td>Is a process to set the user in the system. In the master pengguna management there are only one processes which is edit pengguna’s data.</td>
</tr>
<tr>
<td>13</td>
<td>Master Perawatan</td>
<td>Confirmation Staff</td>
<td>Is a process to set the treatment room. In the master perawatan management there are two processes which is edit perawatan’s data, and delete perawatan’s data.</td>
</tr>
<tr>
<td>14</td>
<td>Master Pasien</td>
<td>Confirmation Staff</td>
<td>Is a process to see the patient data that already send by Pelabuhan hospital. In the master pasien management there are three processes which is check arrival patient, confirmation arrival patient and delete pasien’s data.</td>
</tr>
<tr>
<td>15</td>
<td>Master Pengguna</td>
<td>Confirmation Staff</td>
<td>Is a process to set the user in the system. In the master pengguna management</td>
</tr>
</tbody>
</table>
there are only one processes which is edit pengguna’s data.

c) Use case Diagram

In use case diagrams, authors describe the use case diagram.

Figure 4.4 Is a Detail Use case Administrator in e-Reservation Room System:
Figure 4.4 Use Case Actor

Figure 4.5 to 4.7 is a detailed picture of each use case actor:

Figure 4.4 Is a Detail Use case Administrator:
Figure 4.4 describe about interaction between admin and the system, admin can manage Master Rumah Sakit, Master Spesialis, Master Pasien, Master Perawatan, and Master Pengguna. In each Master, admin can access or manage to check patient, input, edit, and delete data.
Figure 4.5 Use Case Details Pelabuhan hospital staff as follows:

**Figure 4.5** Detail use case Pelabuhan Hospital Staff

**Figure 4.5** describe about interaction between Pelabuhan hospital Staff and the system, Pelabuhan hospital Staff can manage Master Pasien and Master Pengguna. In each Master, Pelabuhan hospital staff can access or manage to check patient, print referral letter, input and delete data.
Figure 4.6 Use Case Details Koja hospital staff as follows:

**Figure 4.6 Detail Use case Koja Hospital staff**

**Figure 4.6** describe about interaction between Koja hospital staff and the system, Koja hospital staff can manage Master Perawatan, Master Pasien and Master Pengguna. In each Master, Koja hospital staff can access or manage to check arrival patient, confirmation referral letter, and delete data.
Figure 4.7 Use Case Details Confirmation staff as follows:

Figure 4.7 Detail Use case Confirmation Staff

Figure 4.7 describe about interaction between confirmation staff and the system, confirmation staff can manage Master Perawatan, Master Pasien and Master Pengguna. In each Master, confirmation staff can access or manage to check arrival patient, confirmation referral letter, and delete data.
d) Use case Scenario

Here is the scenario of each use case that has been previously identified:

**Table 4.4 Use Case Scenario Login**

<table>
<thead>
<tr>
<th>Use Case Name:</th>
<th>Login</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actor(s)</td>
<td>Administrator, Pelabuhan hospital staff, Koja hospital staff, confirmation staff.</td>
</tr>
<tr>
<td>Description:</td>
<td>Use Case describes an actor who will enter the system.</td>
</tr>
<tr>
<td>Precondition:</td>
<td>Actor should have a username and password that is registered in the system.</td>
</tr>
<tr>
<td>Trigger:</td>
<td>Actor wants to enter the system</td>
</tr>
<tr>
<td>Basic Flow:</td>
<td><strong>Actors Activity</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Step 1:</strong> Actor Insert a username and password then press the login button.</td>
</tr>
<tr>
<td></td>
<td><strong>Step 3:</strong> The system responds by directing the actor to the home page system.</td>
</tr>
</tbody>
</table>
Alternative Field: **Alt-Step 1:** If the username and password verification does not match then the system will show the display message that the login is not appropriate.

Postcondition: Actor has been on the home page

Business Rules: Actor has been on Logout on the system

---

**Table 4.6 Use Case Scenario Master Rumah Sakit**

<table>
<thead>
<tr>
<th>Use Case Name:</th>
<th>Master rumah sakit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actor (s)</td>
<td>Administrator</td>
</tr>
<tr>
<td>Description:</td>
<td>This use case describes the event of an actor is to input, edit, and delete rumah sakit’s data that will contribute to the system.</td>
</tr>
<tr>
<td>Precondition:</td>
<td>Enter to the system</td>
</tr>
<tr>
<td>Trigger:</td>
<td>Actor want to manage hospital Data</td>
</tr>
</tbody>
</table>

**Basic Flow:**

**Step1:** The user should log in as administrator, then select the master, and choose master rumah sakit

**Step2:** The System Show master rumah sakit page
<table>
<thead>
<tr>
<th>Step 3:</th>
<th>Admin choose input rumah sakit’s data.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 4:</td>
<td>Admin choose edit rumah sakit’s data.</td>
</tr>
<tr>
<td>Step 5:</td>
<td>The rumah sakit’s data edited.</td>
</tr>
<tr>
<td>Step 6:</td>
<td>Admin choose delete rumah sakit’s data.</td>
</tr>
<tr>
<td>Step 7:</td>
<td>The rumah sakit’s data deleted.</td>
</tr>
</tbody>
</table>

**Alternative Field:**

| Alternative Field: | Alt-Step1: if the admin want to control master rumah sakit, so choose menu master rumah sakit. |

**Post condition:**

| Post condition: | Actor has been on the master rumah sakit page. |

**Business Rules:**

| Business Rules: | Actor has a valid user name and password. |
Table 4.7 Use Case Scenario Master perawatan

<table>
<thead>
<tr>
<th>Use Case Name:</th>
<th>Master perawatan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actor (s)</td>
<td>Administrator</td>
</tr>
<tr>
<td>Description:</td>
<td>This use case describes the event of an actor is to input, edit, and delete perawatan’s data or set the treatment room data in Koja hospital Jakarta.</td>
</tr>
<tr>
<td>Precondition:</td>
<td>Enter to the system.</td>
</tr>
<tr>
<td>Trigger:</td>
<td>Actor want to manage master perawatan data.</td>
</tr>
<tr>
<td>Basic Flow:</td>
<td><strong>Actors Activity</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Step 1:</strong> The user should log in as administrator, then select the master, and choose master perawatan.</td>
</tr>
<tr>
<td></td>
<td><strong>Step 3:</strong> Admin choose input perawatan’s data.</td>
</tr>
<tr>
<td></td>
<td><strong>Step 4:</strong> Admin choose edit perawatan’s data.</td>
</tr>
</tbody>
</table>
Step 6: Admin choose delete perawatan’s data.

Step 7: The perawatan’s data deleted.

Alternative Field: Alt-Step1: If admins want to control the master perawatan, then select master perawatan menu.

Postkondition: Actor has been on the master perawatan page.

Business Rules: Actor has a valid user name and password.

Table 4.8 Use Case Specification for Master Spesialis data

<table>
<thead>
<tr>
<th>Use Case Name:</th>
<th>Master Spesialis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actor (s)</td>
<td>Administrator</td>
</tr>
<tr>
<td>Description:</td>
<td>This use case describes the event of an actor is to input, edit, and delete spesialis’s data or set the disease data.</td>
</tr>
<tr>
<td>Precondition:</td>
<td>Enter to the system.</td>
</tr>
<tr>
<td>Trigger:</td>
<td>Actor want to manage Master Spesialis data.</td>
</tr>
<tr>
<td>Basic Flow:</td>
<td><strong>Actors Activity</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Step1</strong>: The user should log in as administrator, then select the master spesialis’s page.</td>
</tr>
<tr>
<td>Alternative Field:</td>
<td>Alt-Step1: If admins want to control the master spesialis, then select master spesialis menu.</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Postkondition:</td>
<td>Actor has been on the master spesialis page.</td>
</tr>
<tr>
<td>Business Rules:</td>
<td>Actor has a valid user name and password.</td>
</tr>
</tbody>
</table>

**Step 1:**
Admin choose input spesialis.

**Step 3:**
Admin choose input spesialis’s data.

**Step 4:**
Admin choose edit spesialis’s data.

**Step 5:**
The spesialis’s data edited.

**Step 6:**
Admin choose delete spesialis’s data.

**Step 7:**
The spesialis’s data deleted.

**Step 4:**
System will save every update spesialis’s data to the database and show a list of spesialis’s data that has been update.

**Step 5:**
The spesialis’s data edited.
Table 4.9 Use Case Specification for Master pasien data

<table>
<thead>
<tr>
<th>Use Case Name:</th>
<th>Master pasien</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actor (s)</td>
<td>Administrator</td>
</tr>
<tr>
<td>Description:</td>
<td>This use case describes the event of an actor is to print referral letter, delete pasien’s data, and check pasien’s data and confirmation referral letter or manage patient data.</td>
</tr>
<tr>
<td>Precondition:</td>
<td>Enter to the system.</td>
</tr>
<tr>
<td>Trigger:</td>
<td>Actor want to manage master pasien data.</td>
</tr>
<tr>
<td>Basic Flow:</td>
<td></td>
</tr>
<tr>
<td><strong>Actors Activity</strong></td>
<td><strong>Response System</strong></td>
</tr>
<tr>
<td><strong>Step 1</strong>: The user should log in as administrator, then select the master, and choose master pasien.</td>
<td><strong>Step 2</strong>: The System Show master spesialis’s page.</td>
</tr>
<tr>
<td><strong>Step 3</strong>: Admin choose input pasien’s data.</td>
<td><strong>Step 4</strong>: System will save every update pasien’s data to the database and show a list of pasien’s data that has been update.</td>
</tr>
<tr>
<td><strong>Step 4</strong>: Admin choose edit pasien’s data.</td>
<td><strong>Step 5</strong>: The pasien’s data edited.</td>
</tr>
</tbody>
</table>
**Step 6:** Admin choose delete pasien’s data.

**Step 7:** The pasien’s data deleted.

**Alt-Step1:** If admins want to control the master pasien, then select master pasien menu.

**Postkondition:** Actor has been on the master pasien page.

**Business Rules:** Actor has a valid user name and password.

---

**Table 4.10** Use Case specification for master pengguna data

<table>
<thead>
<tr>
<th>Use Case Name:</th>
<th>Master Pengguna</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actor(s):</td>
<td>Administrator</td>
</tr>
<tr>
<td>Description:</td>
<td>This use case describes the event of an actor is to input, edit, and delete or manage users data.</td>
</tr>
<tr>
<td>Precondition:</td>
<td>Enter to the system.</td>
</tr>
<tr>
<td>Trigger:</td>
<td>Actor want to manage master pengguna data.</td>
</tr>
<tr>
<td>Basic Flow:</td>
<td>Actors Activity</td>
</tr>
<tr>
<td>Step 1: The user should log in as administrator, then select the master, and choose master pengguna.</td>
<td></td>
</tr>
<tr>
<td>---------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3:</strong> Admin choose input pengguna’s data.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4:</strong> Admin choose edit pengguna’s data.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6:</strong> Admin choose delete pengguna’s data.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2:</strong> The System Show master pengguna’s page.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4:</strong> System will save every update pengguna’s data to the database and show a list of pengguna’s data that has been update.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5:</strong> The pengguna’s data edited.</td>
<td></td>
</tr>
<tr>
<td><strong>Step 7:</strong> The pengguna’s data deleted.</td>
<td></td>
</tr>
</tbody>
</table>

| Alternative Field: | **Alt-Step 1:** If admins want to control the master pengguna, then select master pengguna menu. |

| Postkondition: | Actor has been on the master pengguna page. |

| Business Rules: | Actor has a valid user name and password. |
## Table 4.11 Use Case Specification for Master Pasien

<table>
<thead>
<tr>
<th>Use Case Name:</th>
<th>Master Pasien</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actor (s)</td>
<td>Pelabuhan hospital staff</td>
</tr>
<tr>
<td>Description:</td>
<td>This use case describes the event of an actor is to input pasien’s data, print referral letter, and delete pasien’s data or manage Pasien data.</td>
</tr>
<tr>
<td>Precondition:</td>
<td>Enter to the system.</td>
</tr>
<tr>
<td>Trigger:</td>
<td>Actor want to manage Master Spesialis data.</td>
</tr>
<tr>
<td>Basic Flow:</td>
<td><strong>Actors Activity</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Step 1:</strong> The user should log in as Pelabuhan hospital staff, then select the master, and choose master pasien.</td>
</tr>
<tr>
<td></td>
<td><strong>Step 3:</strong> Pelabuhan hospital staff choose input pasien’s data.</td>
</tr>
<tr>
<td>Alternative Field:</td>
<td>Alt-Step1: If Pelabuhan hospital staff want to control the master Pasien, then select master Pasien menu.</td>
</tr>
<tr>
<td>--------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Postkondition:</td>
<td>Actor has been on the Master Pasien page.</td>
</tr>
<tr>
<td>Business Rules:</td>
<td>Actor has a valid user name and password.</td>
</tr>
</tbody>
</table>

**Table 4.12 Use Case Specification for Master Pengguna Pelabuhan hospital staff**

<table>
<thead>
<tr>
<th>Use Case Name:</th>
<th>Master Pengguna</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actor (s)</td>
<td>Pelabuhan hospital staff</td>
</tr>
<tr>
<td>Description:</td>
<td>This use case describes the event of an actor is to manage Pengguna’s data.</td>
</tr>
<tr>
<td>Precondition:</td>
<td>Enter to the system.</td>
</tr>
</tbody>
</table>
### Trigger:
Actor want to manage Master Pengguna data.

### Basic Flow:
<table>
<thead>
<tr>
<th>Actors Activity</th>
<th>Response System</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1:</strong> The user should log in as an Pelabuhan hospital staff, then select the master, and choose Master Pengguna.</td>
<td><strong>Step 2:</strong> System Show Master Pengguna’s page.</td>
</tr>
<tr>
<td><strong>Step 3:</strong> Pelabuhan hospital staff choose edit pengguna’s data.</td>
<td><strong>Step 4:</strong> The pengguna’s data edited.</td>
</tr>
</tbody>
</table>

### Alternative Field:
**Alt-Step1:** If Pelabuhan hospital staff want to control the master Pengguna, then select master Pengguna’s menu.

### Postkondition:
Actor has been on the Master Pengguna page.

### Business Rules:
Actor has a valid user name and password.
**Table 4.13** Use Case Specification for Master Perawatan data Koja hospital staff

<table>
<thead>
<tr>
<th>Use Case Name:</th>
<th>Master perawatan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actor(s):</td>
<td>Koja hospital staff</td>
</tr>
<tr>
<td>Description:</td>
<td>This use case describes the event of an actor is to input, edit, and delete perawatan’s data or set the treatment room data.</td>
</tr>
<tr>
<td>Precondition:</td>
<td>Enter to the system.</td>
</tr>
<tr>
<td>Trigger:</td>
<td>Actor want to input Master Perawatan data.</td>
</tr>
</tbody>
</table>

**Basic Flow:**

<table>
<thead>
<tr>
<th>Actors Activity</th>
<th>Response System</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1:</strong> The user should log in as Koja hospital staff, then select the master, and choose master perawatan.</td>
<td><strong>Step 2:</strong> The System Show master perawatan’s page</td>
</tr>
<tr>
<td><strong>Step 3:</strong> Koja hospital staff choose input perawatan’s data.</td>
<td><strong>Step 4:</strong> System will save every update perawatan’s data to the database and show a list of perawatan’s data that has been update.</td>
</tr>
<tr>
<td><strong>Step 4:</strong> Koja hospital staff choose edit perawatan’s data.</td>
<td><strong>Step 5:</strong> The perawatan’s data edited.</td>
</tr>
</tbody>
</table>
**Step 6:** Koja hospital staff choose delete perawatan’s data.

**Step 7:** The perawatan’s data deleted.

**Alt-Step1:** If Koja hospital staff want to control the master perawatan, then select master perawatan menu.

**Postkondition:** Actor has been on the Master Perawatan page.

**Business Rules:** Actor has a valid user name and password.

### Table 4.14 Use Case Specification for Master pasien data Koja hospital staff

<table>
<thead>
<tr>
<th><strong>Use Case Name:</strong></th>
<th>Master pasien</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Actor (s):</strong></td>
<td>Koja hospital staff</td>
</tr>
<tr>
<td><strong>Description:</strong></td>
<td>This use case describes the event of an actor is to check arrival patient, confirmation arrival patient, delete pasien’s data or manage patient data.</td>
</tr>
<tr>
<td><strong>Precondition:</strong></td>
<td>Enter to the system.</td>
</tr>
<tr>
<td><strong>Trigger:</strong></td>
<td>Actor want to manage master pasien data.</td>
</tr>
<tr>
<td><strong>Basic Flow:</strong></td>
<td><strong>Actors Activity</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Step 1:</strong> The user should log in as Koja hospital staff, then</td>
</tr>
<tr>
<td>Alternative Field:</td>
<td>select the master, and choose master pasien.</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td><strong>Step 3:</strong> Koja hospital staff choose check arrival patient data.</td>
<td><strong>Step 4:</strong> System will show pasien’s data that already input by pelabuhan hospital staff.</td>
</tr>
<tr>
<td><strong>Step 4:</strong> Koja hospital staff choose confirmation arrival patient.</td>
<td><strong>Step 5:</strong> System will update the information of referral patient.</td>
</tr>
<tr>
<td><strong>Step 6:</strong> Koja hospital staff choose delete perawatan’s data.</td>
<td><strong>Step 7:</strong> The perawatan’s data deleted.</td>
</tr>
<tr>
<td><strong>Alt-Step1:</strong> If admins want to control the master pasien, then select master pasien menu.</td>
<td></td>
</tr>
<tr>
<td><strong>Postkondition:</strong></td>
<td>Actor has been on the master pasien page.</td>
</tr>
<tr>
<td><strong>Business Rules:</strong></td>
<td>Actor has a valid user name and password.</td>
</tr>
<tr>
<td>Use Case Name:</td>
<td>Master Pengguna</td>
</tr>
<tr>
<td>---------------</td>
<td>----------------</td>
</tr>
<tr>
<td>Actor (s)</td>
<td>Koja hospital staff</td>
</tr>
<tr>
<td>Description:</td>
<td>This use case describes the event of an actor is to manage Koja data.</td>
</tr>
<tr>
<td>Precondition:</td>
<td>Enter to the system.</td>
</tr>
<tr>
<td>Trigger:</td>
<td>Actor want to manage Master Pengguna data.</td>
</tr>
</tbody>
</table>

**Basic Flow:**

<table>
<thead>
<tr>
<th>Actors Activity</th>
<th>Response System</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1:</strong> The user should log in as a Koja hospital staff, then select the master, and choose Master Pengguna.</td>
<td><strong>Step 2:</strong> System Show Master Pengguna page.</td>
</tr>
<tr>
<td><strong>Step 3:</strong> Koja hospital staff choose edit pengguna’s data.</td>
<td><strong>Step 4:</strong> The pengguna’s data edited.</td>
</tr>
</tbody>
</table>

**Alternative Field:**

| Alt-Step1: | If Koja hospital staff want to control the master Pengguna, then select master Pengguna menu. |

**Postkondition:**

Actor has been on the Master Pengguna page.

**Business Rules:**

Actor has a valid user name and password.
**Table 4.16 Use Case Specification for Master Perawatan data Confirmation staff**

<table>
<thead>
<tr>
<th>Use Case Name:</th>
<th>Master perawatan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actor (s)</td>
<td>Confirmation staff</td>
</tr>
<tr>
<td>Description:</td>
<td>This use case describes the event of an actor is to edit perawatan’s data and delete perawatan’s data or set the treatment room data.</td>
</tr>
<tr>
<td>Precondition:</td>
<td>Enter to the system.</td>
</tr>
<tr>
<td>Trigger:</td>
<td>Actor want to input Master Perawatan data.</td>
</tr>
<tr>
<td>Basic Flow:</td>
<td><strong>Actors Activity</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Step 1:</strong> The user should log in as confirmation staff, then select the master, and choose Master Perawatan.</td>
</tr>
<tr>
<td></td>
<td><strong>Step 3:</strong> Confirmation staff choose edit perawatan’s data.</td>
</tr>
<tr>
<td></td>
<td><strong>Step 5:</strong> Confirmation staff choose delete perawatan’s data.</td>
</tr>
<tr>
<td>Alternative Field:</td>
<td><strong>Alt-Step1:</strong> If confirmation staff want to control the master</td>
</tr>
</tbody>
</table>
perawatan, then select master perawatan menu.

**Postkondition:** Actor has been on the Master Perawatan page.

**Business Rules:** Actor has a valid user name and password.

---

<table>
<thead>
<tr>
<th>Table 4.17 Use Case Specification for Master Perawatan data Confirmation staff</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Use Case Name:</strong></td>
</tr>
<tr>
<td><strong>Actor (s):</strong></td>
</tr>
<tr>
<td><strong>Description:</strong></td>
</tr>
<tr>
<td><strong>Precondition:</strong></td>
</tr>
<tr>
<td><strong>Trigger:</strong></td>
</tr>
<tr>
<td><strong>Basic Flow:</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Alternative Field:</td>
</tr>
<tr>
<td>-------------------</td>
</tr>
<tr>
<td>Postkondition:</td>
</tr>
<tr>
<td>Business Rules:</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Use Case Name:</th>
<th>Master pengguna</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actor (s)</td>
<td>Confirmation staff</td>
</tr>
<tr>
<td>Description:</td>
<td>This use case describes the event of an actor is to manage user data.</td>
</tr>
</tbody>
</table>

**Table 4.18 Use Case Specification for Master Pengguna Confirmation Staff**

**Step 5:** Confirmation staff choose confirmation arrival patient.

**Step 6:** System will update the information of referral patient.

**Step 7:** Confirmation staff choose delete perawatan’s data.

**Step 8:** The perawatan’s data deleted.
<table>
<thead>
<tr>
<th>Precondition:</th>
<th>Enter to the system.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trigger:</td>
<td>Actor want to manage master pengguna data.</td>
</tr>
<tr>
<td>Basic Flow:</td>
<td><strong>Actors Activity</strong></td>
</tr>
<tr>
<td>Step 1:</td>
<td>The user should log in as confirmation staff, then select the master, and choose master pengguna.</td>
</tr>
<tr>
<td>Step 3:</td>
<td>confirmation staff choose edit pengguna’s data.</td>
</tr>
<tr>
<td>Alternative Field:</td>
<td>Alt-Step1: If confirmation staff want to control the master pengguna, then select master pengguna menu.</td>
</tr>
<tr>
<td>Postkondition:</td>
<td>Actor has been on the master pengguna page.</td>
</tr>
<tr>
<td>Business Rules:</td>
<td>Actor has a valid user name and password.</td>
</tr>
</tbody>
</table>
2. Activity Diagram

a) Activity Diagram for Login Process

**Figure 4.8** Activity Diagram for Login Process

**Figure 4.8** describe about actor should enter a username and password first when will enter the system. If the username and password is entered incorrectly, the system returns the user to login menu. If the username and password are entered correctly, the system will show the main menu system.
Figure 4.10 Activity Diagram Master Rumah Sakit for Admin

Figure 4.10 describe about for manage master rumah sakit actor should login to the system first. After success for login actor enter to the system, then select menu master master, after that actor choose master rumah sakit. In the homepage master rumah sakit actor can manage input, edit, and delete hospital data.
For input hospital data actor should choose form tambah rumah sakit and input new data, after input new data actor click button save, then new hospital data is save to the database. Actor can also change the hospital data by choosing the edit button, the system will show the edit form hospital, actor will change hospital data after changing the data actor choose save button and the data automatically save to the database. If the actor wants to delete hospital data, actor can choose delete button, system will give a confirmation message if the hospital data will be deleted, after admin approve the confirmation, after admin approve the confirmation, the system will delete the data from the database.
c) Activity Diagram Perawatan for Admin

Figure 4.11 Activity Diagram Perawatan for Admin

Figure 4.11 describe about for manage master perawatan actor should login to the system first. After success for login, actor enter to the system, then select menu master perawatan, after that actor choose master perawatan. In the homepage master perawatan actor can manage input, edit, and delete treatment room data.
For input treatment room data actor should choose form tambah perawatan and input new data, after input new data actor click button save, then new treatment room data is save to the database. Actor can also change the treatment room data by choosing the edit button, the system will show the edit form treatment room, actor will change treatment room data after changing the data actor choose save button and the data automatically save to the database. If the actor wants to delete treatment room data, actor can choose delete button, system will give a confirmation message if the treatment room data will be deleted, after actor approve the confirmation, the system will delete the data from the database.
d) Acitivity Diagram Master Spesialis for Admin

Figure 4.12 Acitivity Diagram Master Spesialis for Admin

Figure 4.12 describe about for manage master spesials actor should login to the system first. After success for login, actor enter to the system, then select menu master spesialis, after that actor choose master spesialis. In the homepage master spesialis actor can manage input, edit, and delete the disease data.
For input the disease data actor should choose form tambah spesialis and input new data, after input new data actor click button save, then new disease data is save to the database. Actor can also change the disease data by choosing the edit button, the system will show the edit disease data form, actor will change disease data after changing the data actor choose save button and the data automatically save to the database. If the actor wants to delete disease data, actor can choose delete button, system will give a confirmation message if the disease data will be deleted, after actor approve the confirmation, the system will delete the data from the database.
e) Activity Diagram Master Pasien for Admin

Figure 4.13 Activity Diagram Master Pasien for Admin

Figure 4.13 describe about for manage master pasien actor should login to the system first. After success for login, actor enter to the system, then select menu master pasien, after that actor choose master pasien. In the homepage master pasien actor can manage check patient data, confirmation arrival patient, print referral patient and delete patient data.
To check patient the actor should click the check icon and the system will show patient data that already inputted from Pelabuhan hospital staff in the system. To confirm that the patient has arrived actor should choose confirmation button after that the system will show a notification for confirmation or not, if yes the system will automatically change the information that the patient has arrived. To print the referral letter to the actor should choose print button, after that the system will show referral letter in digital form and can be printed directly. If the actor wants to delete patient data, actor can choose delete button, system will give a confirmation message if the patient data will be deleted, after actor approve the confirmation, the system will delete the data from the database.
Figure 4.14 Activity Diagram Master Pengguna for Admin

Figure 4.14 describe about for manage master pengguna actor should login to the system first. After success for login, actor enter to the system, then select menu master pengguna, after that actor choose
master pengguna. In the homepage master pengguna actor can manage input, edit, and delete the user data.

For input the disease data actor should choose form tambah pengguna and input new data, after input new data actor click button save, then new user data is save to the database. Actor can also change user data by choosing the edit button, the system will show the edit user data form, actor will change user data, after changing the data actor choose save button and the data automatically save to the database. If the actor wants to delete user data, actor can choose delete button, system will give a confirmation message if the user data will be deleted, after actor approve the confirmation, the system will delete the data from the database.
g) Activity Diagram Master Pasien for Pelabuhan Hospital Staff

Figure 4.15 Activity Diagram Master Pasien for Pelabuhan Hospital Staff

Figure 4.15 describe about for manage master pasien actor should login to the system first. After success for login, actor enter to the system, then select menu master pasien, after that actor choose master pasien. In the homepage master pasien actor can manage check patient data, confirmation arrival patient, print referral patient and delete patient data.
To check patient the actor should click the check icon and the system will show patient data that already inputted from Pelabuhan hospital staff in the system. To confirm that the patient has arrived actor should choose confirmation button after that the system will show a notification for confirmation or not, if yes the system will automatically change the information that the patient has arrived. To print the referral letter to the actor should choose print button, after that the system will show referral letter in digital form and can be printed directly. If the actor wants to delete patient data, actor can choose delete button, system will give a confirmation message if the patient data will be deleted, after actor approve the confirmation, the system will delete the data from the database.
h) Activity Diagram Master Pengguna for Pelabuhan Hospital

Figure 4.16 Activity Diagram Master Pengguna for Pelabuhan Hospital

Figure 4.16 describe about for manage master pengguna actor should login to the system first. After success for login, actor enter to the system, then select menu master pengguna, after that actor choose
master pengguna. In the homepage master pengguna actor only can manage edit user data.

Actor is only can change user data by choosing the edit button, the system will show the edit user data form, actor will change user data, after changing the data actor choose save button and the data automatically save to the database.

i) Activity Diagram Master Perawatan for Koja Hospital Staff

Figure 4.17 Activity Diagram Master Perawatan for Koja Hospital Staff
Figure 4.17 describes about managing master perawatan, where actors should log in to the system first. After successful login, actors enter the system, then select the master perawatan menu, after which they choose master perawatan. In the homepage for master perawatan, actors can manage input, edit, and delete treatment room data.

For inputting treatment room data, actors should choose the form tambah perawatan and input new data. After inputting new data, actors click the save button, and the new treatment room data is saved to the database. Actors can also change the treatment room data by choosing the edit button. The system will show the edit form for the treatment room, and actors can change the treatment room data. After changing the data, actors choose the save button, and the data is automatically saved to the database. If actors want to delete treatment room data, they can choose the delete button. The system will provide a confirmation message if the treatment room data will be deleted. After the actor approves the confirmation, the system will delete the data from the database.
j) Activity Diagram Master Perawatan For Koja Hospital Staff

Figure 4.18 Activity Diagram Master Perawatan for Koja Hospital Staff

Figure 4.18 describe about for manage master perawatan actor should login to the system first. After success for login, actor enter to the system, then select menu masterperawatan, after that actor choose master perawatan. In the homepage master perawatan actor can manage input, edit, and delete treatment room data.
For input treatment room data actor should choose form tambah perawatan and input new data, after input new data actor click button save, then new treatment room data is save to the database. Actor can also change the treatment room data by choosing the edit button, the system will show the edit form treatment room, actor will change treatment room data after changing the data actor choose save button and the data automatically save to the database. If the actor wants to delete treatment room data, actor can choose delete button, system will give a confirmation message if the treatment room data will be deleted, after actor approve the confirmation, the system will delete the data from the database.
k) Activity Diagram Master Pasien For Koja Hospital Staff

Figure 4.19 Activity Diagram Master Pasien for Koja Hospital Staff

Figure 4.19 describe about for manage master pasien actor should login to the system first. After success for login, actor enter to the system, then select menu master pasien, after that actor choose master pasien. In the homepage master pasien actor can manage check patient data, confirmation arrival patient, and delete patient data.

To check patient the actor should click the check icon and the system will show patient data that already inputted from Pelabuhan
hospital staff in the system. To confirm that the patient has arrived actor should choose confirmation button after that the system will show a notification for confirmation or not, if yes the system will automatically change the information that the patient has arrived. If the actor wants to delete patient data, actor can choose delete button, system will give a confirmation message if the patient data will be deleted, after actor approve the confirmation, the system will delete the data from the database.
1) Activity Diagram Master Pengguna For Koja Hospital Staff

Figure 4.20 Activity Diagram Master Pengguna For Koja Hospital Staff

Figure 4.20 describe about for manage master pengguna actor should login to the system first. After success for login, actor enter to the system, then select menu master pengguna, after that actor choose master pengguna. In the homepage master pengguna actor only can manage edit user data.

Actor is only can change user data by choosing the edit button, the system will show the edit user data form, actor will change user
data, after changing the data actor choose save button and the data automatically save to the database.

**Figure 4.21** Activity Diagram Master Perawatan for Confirmation Staff

**Figure 4.21** describe about for manage master perawatan actor should login to the system first. After success for login, actor enter to
the system, then select menu master perawatan, after that actor choose master perawatan. In the homepage master perawatan actor only can manage edit treatment room data.

Actor only can change the treatment room data by choosing the edit button, the system will show the edit form treatment room, actor will change treatment room data after changing the data actor choose save button and the data automatically save to the database.

n) Activity Diagram Master Pasien for Confirmation Staff

![Activity Diagram Master Pasien for Confirmation Staff](image)

**Figure 4.22** Activity Diagram Master Pasien for Confirmation Staff
**Figure 4.22** describe about for manage master pasien actor should login to the system first. After success for login, actor enter to the system, then select menu master pasien, after that actor choose master pasien. In the homepage master pasien actor can manage check patient data, confirmation patient, and delete patient data.

To check patient the actor should click the check icon and the system will show patient data that already inputted from Pelabuhan hospital staff in the system. To confirm that the patient has arrived actor should choose confirmation button after that the system will show a notification for confirmation or not, if yes the system will automatically change the information that the patient has arrived. If the actor wants to delete patient data, actor can choose delete button, system will give a confirmation message if the patient data will be deleted, after actor approve the confirmation, the system will delete the data from the database.
3. Sequence Diagram

a) Sequence Diagram Master Rumah Sakit for Admin

![Sequence Diagram Master Rumah Sakit for Admin]

**Figure 4.23** Sequence Diagram Master Rumah Sakit for Admin

**Figure 4.243** describe about for manage master rumah sakit data, actor should enter to the master rumah sakit form, in the form input hospital there is a function for input hospital data. If the category data has been entered, the data will be checked first by the system (validation). If the data entered is complete, so hospital data is saved in the database, then the system will show new data on the list of master rumah sakit.
If the actor wants to change the hospital data, the actor can choose the edit menu. After choosing the edit menu, the system will show the edit form. When the name of the hospital has changed, the name of the hospital will be saved to the database and will show new data in the system that has been changed. And if the actor wants to delete the hospital data, the actor can choose the menu delete. After choosing the menu delete, the system will delete the data from the database, then the system will show all new hospital data.

b) Sequence Diagram Master Perawatan for Admin

![Sequence Diagram Master Perawatan for Admin](image)

Figure 4.24 Sequence Diagram Master Perawatan for Admin
Figure 4.24 describe about for manage master perawatan data, actor should enter to master perawatan form, in the form input treatment room there is a function for input treatment room data. If the category data has been entered, the data will be checked first by the system (validation). If the data entered is complete, so treatment room data is saved in the database, then the system will show new data on the list of master perawatan.

If the actor want to change the treatment room data. Actor can choose the edit menu. After choosing edit menu, the system will show the edit form, when the name of the treatment room has changed the name of the treatment room will be save to the database and will show new data in system that has been changed. And if actor want to delete the treatment room data, actor can choose menu delete, after actor choosing menu delete, the system will delete the data from the database, then the system will shows all new treatment room data.
c) Sequence Diagram Master Spesialis for Admin

Figure 4.25 Sequence Diagram Master Spesialis for Admin

Figure 4.25 describe about for manage master spesialis data, actor should enter to master spesialis form, in the form input type of disease there is a function for input type of disease data. If the category data has been entered, the data will be checked first by the system (validation). If the data entered is complete, so the type of disease data is saved in the database, then the system will show new data on the list of master spesialis.
If the actor want to change the treatment room data. Actor can choose the edit menu. After choosing edit menu, the system will show the edit form, when the name of the treatment room has changed the name of the treatment room will be save to the database and will show new data in system that has been changed. And if actor want to delete the treatment room data, actor can choose menu delete, after actor choosing menu delete, the system will delete the data from the database, then the system will shows all new treatment room data.
d) Sequence Diagram Master Pasien for Admin

Figure 4.26 Sequence Diagram Master Pasien for Admin

Figure 4.26 describe about for manage master pasien data, actor should enter to the master patient form, there is a function in the form of patient for confirmation patient. If actor want to confirm the patient has arrived. Actor can choose the confirmation button. After being selected, the system will show form data that the patient has come, and patient data will be changed and saved to the database.
If the actor wants to delete the patient data, the actor can choose button delete, after choosing button delete, the system will delete the data from the database category, then the system will show the patient data. And if actor want to print referral letter, admin can choose the print menu, after choosing print menu, the system will show patient data, then the referral patient can be printed.

e) Sequence Diagram Master Pengguna for Admin

![Sequence Diagram Master Pengguna for Admin]

Figure 4.27 Sequence Diagram Master Pengguna for Admin
Figure 4.27 describe about for manage master pengguna data, actor should enter to master pengguna form, in the form input user there is a function for input user data. If the category data has been entered, the data will be checked first by the system (validation). If the data entered is complete, so the user data is saved in the database, then the system will show new data on the list of master pengguna.

If the actor want to change the user data. Actor can choose the edit menu. After choosing edit menu, the system will show the edit form, when the name of the user has changed the name of the user will be save to the database and will show new data in system that has been changed. And if actor want to delete user data, actor can choose menu delete, after actor choosing menu delete, the system will delete the data from the database, then the system will shows all new user data.
f) Sequence Diagram Master Pasien for Pelabuhan Hospital Staff

Figure 4.28 Sequence Diagram Master Pasien for Pelabuhan Hospital Staff

Figure 4.28 describe about for manage master pengguna data, actor should enter to master pasien form, in the form input patient there is a function for input patient data. If the category data has been entered, the data will be checked first by the system (validation). If the data entered is complete, so the patient data is saved in the database, then the system will show new data on the list of master pasien.

If actor want to delete user data, actor can choose menu delete, after actor choosing menu delete, the system will delete the data from the database, then the system will shows all patient data that already input. And if actor want to print referral letter, admin can choose the
print menu, after choosing print menu, the system will show patient data, then the referral patient letter can be printed.

**g) Sequence Diagram Master Pengguna for Pelabuhan Hospital Staff**

**Figure 4.29** Sequence Diagram Master Pengguna for Pelabuhan Hospital Staff

**Figure 4.30** describe about actor want to change the user data. Actor can choose the edit menu. After choosing edit menu, the system will show the edit form, when the name of the user has changed the name of the user will be save to the database and will show new data in system that has been changed.
h) Sequence Diagram Master Perawatan for Koja Hospital Staff

Figure 4.30 describe about for manage master perawatan data, actor should enter to master perawatan form, in the form input treatment room there is a function for input treatment room data. If the category data has been entered, the data will be checked first by the system (validation). If the data entered is complete, so treatment room data is saved in the database, then the system will show new data on the list of master perawatan.
If the actor want to change the treatment room data. Actor can choose the edit menu. After choosing edit menu, the system will show the edit form, when the name of the treatment room has changed the name of the treatment room will be save to the database and will show new data in system that has been changed. And if actor want to delete the treatment room data, actor can choose menu delete, after actor choosing menu delete, the system will delete the data from the database, then the system will shows all new treatment room data.

i) Sequence Diagram Master Pasien for Koja Kospital Staff

Figure 4.31 Sequence Diagram Master Pasien for Koja Kospital Staff
Figure 4.31 describe about for manage master pasien data, actor should enter to the master patient form, there is a function in the form of patient for confirmation patient. If actor want to confirm the patient has arrived. Actor can choose the confirmation button. After being selected, the system will show form data that the patient has come, and patient data will be changed and saved to the database.

j) Sequence Diagram Master Pengguna for Koja Hospital Staff

Figure 4.32 Sequence Diagram Master Pengguna for Koja Hospital Staff

Figure 4.32 describe about actor want to change the user data. Actor can choose the edit menu. After choosing edit menu, the system will show the edit form, when the name of the user has changed the name of the user will be save to the database and will show new data in system that has been changed.
k) Sequence diagram master perawatan for confirmation staff

Figure 4.33 describe about if the actor want to change the treatment room data. Actor can choose the edit menu. After choosing edit menu, the system will show the edit form, when the name of the treatment room has changed the name of the treatment room will be save to the database and will show new data in system that has been changed.
1) Sequence diagram master pasien for confirmation staff

![Sequence Diagram Master Pasien for Confirmation Staff]

**Figure 4.34** Sequence Diagram Master Pasien for Confirmation Staff

**Figure 4.34** describe about for manage master pasien data, actor should enter to the master patient form, there is a function in the form of patient for confirmation patient. If actor want to confirm the patient has arrived, Actor can choose the confirmation button. After being selected, the system will show form data that the patient has come, and patient data will be changed and saved to the database.
m) Sequence Diagram Master Pengguna for Confirmation Staff

Figure 4.35 describe about actor want to change the user data. Actor can choose the edit menu. After choosing edit menu, the system will show the edit form, when the name of the user has changed the name of the user will be save to the database and will show new data in system that has been changed.
4. Database Design

a) Design of Class Diagram

\[\text{Figure 4.36 Design of Class Diagram}\]
Base on Figure 4.35 in the e-Reservation Room System class diagram start from main class, main class have some function that organize interface which is for login, manage master pasien, manage master spesialis, manage master pengguna, manage master perawatan, and manage master rumah sakit. On the figure 4.35 shown that the relations multiplicity in the main class for every class only have one (1) relation. This shows that every management has one relation with every single view of each operation. To manage every function in the system, user in this system should login to the system, the data of every user in this system is save to the database. This is has a meaning if to manage every single function in this system, user should login first as user who manage every single function in the system. Except login activity, every single function has a relation with database, every management has one relation or more than one for every activity that carried out during the management in the system. The last is every management has relation with every class data in the database that is using for storage the data, and those data classes has relation with one class to another class.
b) Database Design

Base on Figure 4.36

1. konfirmasi_pasien (Confirmation Tabel)
   
   Tabel Name : konfirmasi_tabel

   Primary Key : id

   Foreign Key : -

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data Type</th>
<th>Size</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Id</td>
<td>Integer</td>
<td>10</td>
<td>Id for</td>
</tr>
</tbody>
</table>
### Table 4.20 MengirimPasien

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data type</th>
<th>Size</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Id</td>
<td>Integer</td>
<td>10</td>
<td>Id for send the referral patient</td>
</tr>
<tr>
<td>Tanggal</td>
<td>Date</td>
<td>-</td>
<td>Date for send the referral patient</td>
</tr>
</tbody>
</table>

2. mengirim_pasien (Send patient tabel)

- **Table Name**: mengirim_pasien
- **Primary Key**: id
- **Foreign Key**: id_pasient, id_rumah_sakit, id_master_dokter, id_master_spesialis
### Pasien (Patient Table)

**Table Name**: pasien  
**Primary Key**: id  
**Foreign key**: -

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data type</th>
<th>Size</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Id</td>
<td>Integer</td>
<td>10</td>
<td>Id for referral patient</td>
</tr>
<tr>
<td>No_daftar</td>
<td>Varchar</td>
<td>100</td>
<td>Registration</td>
</tr>
</tbody>
</table>

---

1. **Id_pasient**  
   **Data type**: Integer  
   **Size**: 10  
   **Explanation**: Id patient for referral patient.

2. **Id_rumah_sakit**  
   **Data type**: Integer  
   **Size**: 10  
   **Explanation**: Id hospital for referral patient.

3. **Id_master_perawatan**  
   **Data type**: Integer  
   **Size**: 10  
   **Explanation**: Id perawatan/room for referral patient.

4. **Id_master_spesialis**  
   **Data type**: Integer  
   **Size**: 10  
   **Explanation**: Id for what kind of disease that patient have.
<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Length</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nama</td>
<td>Varchar</td>
<td>100</td>
<td>The referral patient name</td>
</tr>
<tr>
<td>Alamat</td>
<td>Text</td>
<td>-</td>
<td>The address of referral patient</td>
</tr>
<tr>
<td>Jk</td>
<td>Varchar</td>
<td>10</td>
<td>Gender of referral patient</td>
</tr>
<tr>
<td>Tlp</td>
<td>Varchar</td>
<td>100</td>
<td>Phone number referral patient</td>
</tr>
<tr>
<td>Jenis penyakit</td>
<td>Text</td>
<td>-</td>
<td>The illness of referral patient</td>
</tr>
<tr>
<td>Saran dokter</td>
<td>Text</td>
<td>-</td>
<td>Suggestion from doctor to referral patient</td>
</tr>
<tr>
<td>Tgl daftar</td>
<td>Varchar</td>
<td>100</td>
<td>Date of registration when the patient registration</td>
</tr>
<tr>
<td>Tgl rujuk</td>
<td>Varchar</td>
<td>100</td>
<td>Date when the referral patient</td>
</tr>
<tr>
<td>Field</td>
<td>Type</td>
<td>Length</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>----------</td>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Tgl_lahir</td>
<td>Varchar</td>
<td>100</td>
<td>Date of birth referral patient</td>
</tr>
<tr>
<td>Rs_dari</td>
<td>Varchar</td>
<td>100</td>
<td>From where hospital the referral patient is come</td>
</tr>
<tr>
<td>Rs</td>
<td>Varchar</td>
<td>100</td>
<td>The name of hospital</td>
</tr>
<tr>
<td>Spesialis</td>
<td>Integer</td>
<td>100</td>
<td>The name for what type of treatment</td>
</tr>
<tr>
<td>Ruangan</td>
<td>Integer</td>
<td>100</td>
<td>The room treatment for referral patient</td>
</tr>
<tr>
<td>Datang</td>
<td>Integer</td>
<td>11</td>
<td>The date when the referral patient is come to Koja hospital</td>
</tr>
<tr>
<td>Masuk_ruangan</td>
<td>Integer</td>
<td>11</td>
<td>The referral</td>
</tr>
</tbody>
</table>
4. Pasien_datang

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data type</th>
<th>Size</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Id</td>
<td>Integer</td>
<td>10</td>
<td>Id for patient that already come</td>
</tr>
</tbody>
</table>

**Table 4.22 PasienDatang**
<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data type</th>
<th>Size</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Id_mengirim_pasien</td>
<td>Integer</td>
<td>10</td>
<td>Id for patient who already come to Koja hospital</td>
</tr>
<tr>
<td>Tanggal_datang</td>
<td>Date</td>
<td>-</td>
<td>The date when the patient is come to Koja hospital</td>
</tr>
</tbody>
</table>

5. Master_spesialis (Disease Tabel)

Table 4.23 MasterSpesialis

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data type</th>
<th>Size</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Id</td>
<td>Integer</td>
<td>10</td>
<td>Id for what kind of disease</td>
</tr>
<tr>
<td>Nama</td>
<td>Varchar</td>
<td>100</td>
<td>The name for what kind of disease</td>
</tr>
</tbody>
</table>
6. Master_rs

Table name : master_rs

Primary key : id

Foreign key : -

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data type</th>
<th>Size</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Id</td>
<td>Integer</td>
<td>10</td>
<td>Id for hospital in this system</td>
</tr>
<tr>
<td>Nama</td>
<td>Varchar</td>
<td>200</td>
<td>The name hospital in this system</td>
</tr>
<tr>
<td>Alamat</td>
<td>Text</td>
<td>-</td>
<td>Address the hospital in this system</td>
</tr>
</tbody>
</table>
Spesialis | Text | - | The service disease treatment Koja hospital

Tlp | Varchar | 100 | Phone number for both hospital in this system

Fax | Varchar | 100 | Fax number for both hospital in this system

Email | Varchar | 100 | Email adress for both hospital in this system

Kab/kota | Varchar | 100 | Address for both hospital in this system
7. Master_perawatan

Table name : master_perawatan
Primary key : id
Foreign key : -

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data type</th>
<th>Size</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Id</td>
<td>Int</td>
<td>11</td>
<td>Id for room treatment in the Koja hospital</td>
</tr>
<tr>
<td>Jenis</td>
<td>Varchar</td>
<td>200</td>
<td>What kind of room treatment in the system</td>
</tr>
<tr>
<td>Rs</td>
<td>Varchar</td>
<td>100</td>
<td>The hospital that has room treatment in this system</td>
</tr>
<tr>
<td>Ruangan</td>
<td>Int</td>
<td>100</td>
<td>The treatment room</td>
</tr>
</tbody>
</table>
### Table 4.26 User

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Data type</th>
<th>Size</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Id</td>
<td>Integer</td>
<td>11</td>
<td>Id for user in this system</td>
</tr>
<tr>
<td>User</td>
<td>Varchar</td>
<td>50</td>
<td>User in this system</td>
</tr>
<tr>
<td>Sebagai</td>
<td>Varchar</td>
<td>100</td>
<td>What kind user in this system</td>
</tr>
<tr>
<td>Katasandi</td>
<td>Varchar</td>
<td>32</td>
<td>Password for user in this system</td>
</tr>
<tr>
<td>Namalengkap</td>
<td>Varchar</td>
<td>100</td>
<td>Full name user in this system</td>
</tr>
<tr>
<td>Rs</td>
<td>Integer</td>
<td>11</td>
<td>Every hospital in this system has a user in this system</td>
</tr>
</tbody>
</table>
5. Design Interface

a) Design interface login

Figure 4.38 Interface Login
b) Interface Design Homepage Administrator

Figure 4.39 Interface Homepage Administrator
c) Interface Design Master Rumah Sakit Administrator

Figure 4.40 Interface Master Rumah Sakit Administrator

d) Interface Design Perawatan Administrator

Figure 4.41 Interface Perawatan Administrator
e) **Interface Design Spesialis Administrator**

![Figure 4.42 Interface Spesialis Administrator](image-url)

<table>
<thead>
<tr>
<th>Nama Spesialis</th>
<th>Rumah Sakit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RS Koja Jakarta</td>
</tr>
<tr>
<td>Anak</td>
<td>RS Koja Jakarta</td>
</tr>
<tr>
<td>RGT</td>
<td>RS Koja Jakarta</td>
</tr>
<tr>
<td>Umum</td>
<td>RS Koja Jakarta</td>
</tr>
<tr>
<td>Peri-Peri</td>
<td>RS Koja Jakarta</td>
</tr>
<tr>
<td>Kantor</td>
<td>RS Koja Jakarta</td>
</tr>
<tr>
<td>Klinik</td>
<td>RS Koja Jakarta</td>
</tr>
</tbody>
</table>

**Log in anda sebagai Administrator**

---

**Figure 4.42 Interface Spesialis Administrator**
f) **Interface Design Pasien Administrator**

![Figure 4.43 Interface Pasien Administrator](image)

*Figure 4.43 Interface Pasien Administrator*
g) Interface Design Homepage Pelabuhan Hospital Staff

Figure 4.44 Interface Homepage Pelabuhan Hospital Staff
h) Interface Design Pelabuhan Hospital Staff

Figure 4.45 Interface Master Pasien Pelabuhan Hospital Staff
i) Interface Design Master Pengguna Pelabuhan Hospital

Figure 4.46 Interface Master Pengguna Pelabuhan Hospital
j) Interface Design Homepage Koja Hospital Staff

Figure 4.47 Interface Homepage Koja Hospital Staff
k) **Interface Design Mater Perawatan Koja Hospital**

![Figure 4.48 Interface Mater Perawatan Koja Hospital](image)

l) **Interface Design** Master Pengguna Koja Hospital

![Figure 4.49 Interface Master Pengguna Koja Hospital](image)
m) Interface Design Master Pasien Koja Hospital

**Figure 4.50** Interface Master Pasien Koja Hospital
Figure 4.51 Interface Homepage Confirmation Staff
o) Interface Design Master Perawatan Confirmation Staff

Figure 4.52 Interface Master Perawatan Confirmation Staff

p) Interface Design Master Pengguna Confirmation Staff

Figure 4.53 Interface Master Pengguna Confirmation Staff
q) Interface Design Master Pasien Confirmation Staff

Figure 4.54 Interface Master Pasien Confirmation Staff
4.2.2.2. Build System

In this stage the author doing the implementation base on, database design, application design, or interface design.

1. **Programing language and Component**

Programing language that author use to develop this system is PHP, CSS, javascrip. For database the author use MYSQL. For the source code from this application included in attachment.

2. **Tools**

Tools that used for develop this system which is NetBeans IDE 7.1 and Adobe Photoshop. The code can be run using application server, the author used Xampp Package for windows.

4.3.1. **Implementation**

After design and coding is done, so next step is implementation the output to the system then testing the system that has been build.

4.2.3.1. **Installation**

After develop the system there is some requirements for installing the system, which is:
1. Hardware

For develop the system in this research, author using 1(One) PC (Personal Computer) or Laptop that has function as server and client, and this application will be implement in the PC (Personal Computer) in the IT staff Koja hospital Jakarta and 1(one) PC(Personal Computer). The specification of hardware will describe, which is:

a. Server

- Processor : Intel Core2Duo or more.
- Memory : 1GB or more.
- Hard disk : 250 GB or more.
- VGA Card : 32 MB.

b. Client

- Processor : minimum intel Pentium IV or more.
- Memory : 1GB or more.
- Hardisk : 250 GB.
- VGA Card : 32 MB.

1. Software

Minimum software requirement that will be used is:
• Operating System : Microsoft Windows XP
• Web Server : XAMPP Package for Windows

4.2.3.2. Testing (Blackbox)

For the testing stage the author using Black Box testing. Black-box testing is a method of software testing that examines the functionality of an application. Test results black-box testing are presented in the following table:

1.  Role Administrator:

<table>
<thead>
<tr>
<th>No</th>
<th>Case test</th>
<th>Requirement</th>
<th>Expected result</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Login</td>
<td>Administrator insert username and password.</td>
<td>Enter the system</td>
<td>OK</td>
</tr>
<tr>
<td>2</td>
<td>Logout</td>
<td>Administrator choose logout button.</td>
<td>Administrator exit from the system</td>
<td>OK</td>
</tr>
<tr>
<td>3</td>
<td>Input Rumah Sakit’s Data</td>
<td>Administrator choose input rumah button.</td>
<td>Administrator can input rumah sakit’s data.</td>
<td>OK</td>
</tr>
<tr>
<td>4</td>
<td>Edit Rumah Sakit’s</td>
<td>Administrator</td>
<td>Administrator</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>Data</td>
<td>Administrator choose edit rumah sakit’s button.</td>
<td>Administrator can edit Rumah Sakit’s data.</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>------</td>
<td>-------------------------------------------------</td>
<td>-------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Delete Rumah Sakit’s Data</td>
<td>Administrator choose delete rumah sakit’s button.</td>
<td>Administrator can delete rumah sakit’s data.</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Input Perawatan’s Data</td>
<td>Administrator choose Input perawatan’s button.</td>
<td>Administrator can input perawatan’s data.</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Edit Perawatan’s Data</td>
<td>Administrator choose edit perawatan’s button.</td>
<td>Administrator can edit perawatan’s data.</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Delete Perawatan’s Data</td>
<td>Administrator choose delete perawatan’s button.</td>
<td>Administrator can delete perawatan’s data.</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Input Spesialis’s Data</td>
<td>Administrator choose Input</td>
<td>Administrator can input</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Edit Spesialis’s Data</td>
<td>Administrator choose edit spesialis’s button. Administrator can edit perawatan’s data. OK</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Delete Spesialis’s Data</td>
<td>Administrator choose delete spesialis’s button. Administrator can delete perawatan’s data. OK</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Check Pasien’s Data And Confirmation Arrival Patient</td>
<td>Administrator choose check pasien’s Data And Confirmation Arrival Patient button. Administrator can check pasien’s Data And Confirmation Arrival Patient data. OK</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Print Referral Letter</td>
<td>Administrator choose Print Referral Letter button. Administrator can Print Referral Letter data. OK</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
2. Role Pelabuhan hospital staff:

Table 4.28 Role Pelabuhan hospital staff

<table>
<thead>
<tr>
<th>No</th>
<th>Case test</th>
<th>Requirement</th>
<th>Expected result</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>Delete Pasien’s Data</td>
<td>Administrator choose delete pasien’s button.</td>
<td>Administrator can delete pasien’s data.</td>
<td>OK</td>
</tr>
<tr>
<td>15</td>
<td>Input Pengguna’s Data</td>
<td>Administrator choose Input pengguna’s button.</td>
<td>Administrator can input pengguna’s data.</td>
<td>OK</td>
</tr>
<tr>
<td>16</td>
<td>Edit Pengguna’s Data</td>
<td>Administrator choose edit pengguna’s button.</td>
<td>Administrator can edit pengguna’s data.</td>
<td>OK</td>
</tr>
<tr>
<td>17</td>
<td>Delete Pengguna’s Data</td>
<td>Administrator choose delete pengguna’s button.</td>
<td>Administrator can delete pengguna’s data.</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>Input Pasien’s Data</td>
<td>Pelabuhan hospital staff choose Input pasien’s button.</td>
<td>Pelabuhan hospital staff can delete pengguna’s data.</td>
<td>OK</td>
</tr>
<tr>
<td>---</td>
<td>---------------------</td>
<td>------------------------------------------------------</td>
<td>-------------------------------------------------</td>
<td>----</td>
</tr>
<tr>
<td>2</td>
<td>Check Pasien’s Data</td>
<td>Pelabuhan hospital staff choose check pasien’s data button.</td>
<td>Pelabuhan hospital staff</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td>Print Referral Patient</td>
<td>Pelabuhan hospital staff choose Print Referral Letter button.</td>
<td>Pelabuhan hospital staff can Print Referral Letter data.</td>
<td>OK</td>
</tr>
<tr>
<td>4</td>
<td>Delete Pasien’s Data</td>
<td>Pelabuhan hospital staff choose delete pasien’s button.</td>
<td>Pelabuhan hospital staff can delete pasien’s data.</td>
<td>OK</td>
</tr>
<tr>
<td>5</td>
<td>Edit Pengguna’s Data</td>
<td>Pelabuhan hospital staff choose edit</td>
<td>Pelabuhan hospital staff can delete</td>
<td>OK</td>
</tr>
</tbody>
</table>
Table 4.29 Role Koja hospital staff

<table>
<thead>
<tr>
<th>No</th>
<th>Case test</th>
<th>Requirement</th>
<th>Expected result</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Input Perawatan’s Data</td>
<td>Koja hospital staff choose Input perawatan’s button.</td>
<td>Koja hospital staff can input perawatan’s data.</td>
<td>OK</td>
</tr>
<tr>
<td>2</td>
<td>Edit Perawatan’s Data</td>
<td>Koja hospital staff choose edit perawatan’s button.</td>
<td>Koja hospital staff can edit perawatan’s data.</td>
<td>OK</td>
</tr>
<tr>
<td>3</td>
<td>Delete Perawatan’s Data</td>
<td>Koja hospital staff choose delete perawatan’s</td>
<td>Koja hospital staff can delete perawatan’s data.</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>4</td>
<td>Check Arrival Patient</td>
<td>Koja hospital staff choose check pasien’s Data And Confirmation Arrival Patient button.</td>
<td>Koja hospital staff can check arrival patient data.</td>
<td>OK</td>
</tr>
<tr>
<td>5</td>
<td>Confirmation Arrival Patient</td>
<td>Koja hospital staff choose confirmation arrival patient button.</td>
<td>Koja hospital staff can Pprint referral letter data.</td>
<td>OK</td>
</tr>
<tr>
<td>6</td>
<td>Delete Pasien’s Data</td>
<td>Koja hospital staff choose delete pasien’s button.</td>
<td>Koja hospital staff can delete pasien’s data.</td>
<td>OK</td>
</tr>
<tr>
<td>7</td>
<td>Edit Pengguna’s Data</td>
<td>Koja hospital staff choose edit pengguna’s button.</td>
<td>Koja hospital staff can delete pengguna’s data.</td>
<td>OK</td>
</tr>
</tbody>
</table>
4. Role confirmation staff

**Table 4.30** Role confirmation staff

<table>
<thead>
<tr>
<th>No</th>
<th>Case test</th>
<th>Requirement</th>
<th>Expected result</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Edit Perawatan’s Data</td>
<td>Confirmation staff choose edit perawatan’s button.</td>
<td>Confirmation staff can edit perawatan’s data.</td>
<td>OK</td>
</tr>
<tr>
<td>2</td>
<td>Delete Perawatan’s Data</td>
<td>Confirmation staff choose delete perawatan’s button.</td>
<td>Confirmation staff can delete perawatan’s data.</td>
<td>OK</td>
</tr>
<tr>
<td>3</td>
<td>Confirmation Room</td>
<td>Confirmation staff choose confirmation arrival patient button.</td>
<td>Confirmation staff can Pprint referral letter data.</td>
<td>OK</td>
</tr>
<tr>
<td>4</td>
<td>Delete Pasien’s Data</td>
<td>Confirmation staff choose</td>
<td>Confirmation staff can delete</td>
<td>OK</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td></td>
<td>delete pasien’s button.</td>
<td>pasien’s data.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Edit Pengguna’s Data</td>
<td>Confirmation staff choose edit pengguna’s button.</td>
<td>Confirmation staff can delete pengguna’s data.</td>
<td>OK</td>
</tr>
</tbody>
</table>
CHAPTER V
CONCLUSIONS

After conducting series of research and testing, in this chapter the author will outline the conclusions and recommendations from this research.

5.1. SUMMARY

1. This system can help the referral patient from Pelabuhan hospital get the treatment room in Koja hospital and also this system can be update available treatment room in Jakarta Koja hospital that will be seen by Pelabuhan hospital.

2. This system can guarantee the privacy of patient data because data transaction using database so irresponsible person cannot access it.

5.2. Recommendation

This system is still not perfect. There are some secession that can be done by other author to improve this system:

1. It is expected that the system provide the detail of a doctor that will handle referral patient and facility that referral patient needed.

2. System may be improved by designing more reliable and user friendly interface.
Reference


**National Institute Of Health,** 2013 *Department Of Health And Human Service.* Fiscal Year 2015


ATTACHMENT 1

Interview for e-Reservation Room System (Case study : Koja hospital Jakarta and Pelabuhan hospital Jakarta)

Interviewer : Muhammad Hanif
Correspondent: Muhammad Rahman S.Kom
Position : Head of Information Technology in Koja Hospital

1. How does it work about to send referral patient from Pelabuhan hospital Jakarta to Koja hospital Jakarta?

Sending referral patients from Pelabuhan hospital to Koja hospital still using manual, that is still using telephone between Koja hospital to another hospital and still use the guest book, and therefore if there is a patient who wants to be referred to Koja hospital often have some problems.

2. After knowing the process, I want to ask about whether to use the manual system in Koja hospital is hassles for handle the process of accept referral patient?
Of course, because if the activities for patients that will be referred from Pelabuhan hospital to Koja hospital still use the manual will be have a big problem in referral patients activities.

3. After getting the three points above, what do you expect?

I hope there is a system that will solve the problem for patients who would refer to Koja hospital, the system that have information about the available room, input the patient data, etc.
PEMERINTAH PROVINSI DAERAH KHRUSUS IBUKOTA JAKARTA
RUMAH SAKIT UMUM DAERAH KOJA
Jalan Deli No. 4 Tanjung Priok Telepon 43938478 Fax 4352401/4372273
JAKARTA
Kode Pos : 14220

SURAT KETERANGAN
Nomor : D/KP/864.34

Yang berlaku pada dalam dibawah ini:
Nama : dr. Togi Asman Sinaga, M. Kes
Nip : 196107121986121001
Jabatan : Direktur RSUD Koja
Unit Kerja : RSUD Koja

Dengan ini menerangkan bahwa:
Nama : Muhammad Hanif
NIM : 10809110008

Benar mahasiswa tersebut diatas telah melaksanakan penelitian di RSUD Koja TMT
November sampai desember 2013 dengan judul skripsi: e-Reservation Room System.

Demikian surat keterangan ini dibuat untuk dipergunakan dengan semestinya.

Jakarta, 28 September 2014
Direktur RSUD Koja
Provinsi DKI Jakarta

[Signature]

[Signature]

[Signature]
RUMAH SAKIT PELABUHAN
JAKARTA

Nomor : DL.57/3 / 1 / RSP.Jkt-2014
Klasifikasi : Biasa
Lampiran : -
Perihal : Persetujuan Riset

Jakarta, 12 Februari 2014

Kepada

Yth. Wdek Bld. Akademik Univ. Islam Negeri (UI) Syarif Hidayatullah
Jakarta - Fak. Sains dan Teknologi DI

CIPUTAT 15412


2. Sehubungan butir 1, pada prinsipnya kami menyetujui permohonan Bapak bagi mahasiswa/ atau nama:
   Nama : Muhammad Hanif
   NIM : 109091100008
   Jurusan/Semester : Teknik Informatika (International)/IX (sembilan)
   Program : S-1 (Strata 1)
   Tahun Akademik : Semester Ganjil 2013/2014
   Judul Skripsi : E-Reservation Room System

   Untuk melaksanakan penelitian, mengambil data dan wawancara yang digunakan untuk penyusunan skripsi, untuk tujuan pelaksanaannya dapat beroordinasi dengan Kabag. SDM dan TU Cq. Diklat RS Pelabuhan Jakarta.

3. Demikian kami sampaikan, atas perhatian dan kerjasamanya diucapkan terima kasih.

RUMAH SAKIT PELABUHAN JAKARTA

RDM. SYAHEL JUDA EFFENDI, MM

S.E., MM.Si., No HP. 26301200099

Tembusan :
1. Waka. Yanmed & Keperawatan
2. Kabag. Teknologi Sistem Informasi (TSI)

Jl. Kramat Jaya Tanjung Priok, Jakarta 14260
Telp. 021 - 4403026 (6 saku), Fax. 4403551 Kotak Pos 24 JKUTG.
<?php
if (!defined('BASEPATH')) exit('No direct script access allowed');

class Beranda extends CI_Controller {

    public function __construct() {
        parent::__construct();

        $this->model_utils->sessionExpired();

        $this->data['data'] = $this->model_core->dbSelectAll('pasien');

        $this->data['dataLaki'] = $this->model_core->dbSelectWhere('pasien', array('jk' => 'Laki-laki'));

        $this->data['dataPerempuan'] = $this->model_core->dbSelectWhere('pasien', array('jk' => 'Perempuan'));

        $this->data['dataDatang'] = $this->model_core->dbSelectWhere('pasien', array('datang' => 1));

        $this->data['dataBlmDatang'] = $this->model_core->dbSelectWhere('pasien', array('datang' => 0));

        $this->data['dataBlmMasuk'] = $this->model_core->dbSelectWhere('pasien', array('masuk_ruangan' => 0, 'datang' => 1));

        $this->data['dataSdhMasuk'] = $this->model_core->dbSelectWhere('pasien', array('masuk_ruangan' => 1, 'datang' => 1));
    }
}
```php
<?php
if (!defined('BASEPATH')) exit('No direct script access allowed');

class Login extends CI_Controller {

    public function __construct() {
        parent::__construct();
    }

    public function index() {
        if (!$this->input->post()) {
            redirect('?error=1');
        }

        $where = array('user' => $this->input->post('user'), 'katasandi' => md5($this->input->post('sandi')));
    }

    $where = array('user' => $this->input->post('user'), 'katasandi' => md5($this->input->post('sandi')));
```
$res = $this->model_core->dbLogin('user', $where);

if ($res->num_rows() == 1) {
    $row = $res->row_array();

    $session = array(
        'id' => $row['id'],
        'user' => $row['user'],
        'sebagai' => $row['sebagai'],
        'namalengkap' => $row['namalengkap'],
        'rs' => $row['rs']
    );

    $this->session->set_userdata($session);

    redirect('beranda');
} else {
    redirect('?error=2');
}
}
<?php

/*
 * @author      : * Project Name : Hanif
 * Generated    : Dec 22, 2013 - 4:27:23 PM
 * Filename     : pengguna.php
 * Encoding     : UTF-8
 */

if (!defined('BASEPATH'))
exit('No direct script access allowed');

class Pengguna extends CI_Controller {

    public function __construct() {
        parent::__construct();
        $this->model_utils->sessionExpired();
        $this->data['data'] = $this->model_core->dbSelectAll('user');
        $this->data['rs'] = $this->model_core->dbSelectAll('master_rs');
    }

    public function index() {
        if ($this->session->userdata('sebagai') != "Administrator") {
            // Code here
        }
    }
}
public function ubah() {
    $data = array('id' => $this->uri->segment(3));

    $this->data['current_rs'] = $this->model_core->dbSelectWhere('user', $data)->row_array();
    $this->load->view('pengguna', $this->data);
}