UTeM E-HEALTH INFORMATION SYNCHRONIZATION USING RASPBERRY PI

GAN JUN KAI

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

BORANG PENGESAHAN STATUS TESIS

JUDUL: UTeM E-HEALTH INFORMATION SYNCHRONIZATION USING RASPBERRY PI

SESI PENGAJIAN: 2015/2016

Saya GAN JUN KAI

mengaku membenarkan tesis (PSM/Sarjana/Doktor Falsafah) ini disimpan di Perpustakaan
Fakulti Teknologi Maklumat dan Komunikasi dengan syarat-syarat kegunaan seperti berikut:

1. Tesis dan projek adalah hakmilik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. ** Sila tandakan (/)

_____ SULIT (Mengandungi maklumat yangberdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)

_____ TERHAD (Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)

_____ TIDAK TERHAD

_____ (GAN JUN KAI)

_____ (MOHD HARIZ BIN NAIM @ MOHAYAT)

Alamat tetap: 20, Jalan Jasa 38,
Taman Mutiara Rini,
81300, Skudai, Johor.

Tarikh: 28/8/2016

Tarikh: 28/8/2016

CATATAN: * Tesis dimaksudkan sebagai Laporan Akhir Projek Sarjana Muda (PSM)
** Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa.

UTeM E-HEALTH INFORMATION SYNCHRONIZATION USING RASPBERRY PI

GAN JUN KAI

This report is submitted in partial fulfilment of the requirements for the Bachelor of Computer Science (Software Development)

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2016

DECLARATION

I hereby declare that this project report entitled

UTeM e-health information synchronization using Raspberry Pi

is written by me and is my own effort and that no part has been plagiarized without citations.

STUDENT : _____ Date: 28/8/2016

(GAN JUN KAI)

SUPERVISOR : _____ Date: 28/8/2016

(MOHD HARIZ BIN NAIM @ MOHAYAT)

DEDICATION

For my beloved parents.

The kindest persons I know.

ACKNOWLEDGEMENTS

I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals. I would like to extend my sincere thanks to all of them.

I am highly indebted to Mr. Mohd Hariz Bin Naim @ Mohayat, my supervisor, for his guidance, valuable and helpful suggestions during the planning and development of this work. His willingness to give his time so generously has been very much appreciated.

My appreciations also go to those people who have willingly helped me out with their abilities.

Lastly, I am also thankful towards all my family members for their support and encouragement throughout my study.

Thank you.

ABSTRACT

The objective of this work is to identify the usage of synchronization data using Raspberry Pi in e-Clinical Support System. This system is an automated backup server mechanism via a middleware that able to synchronize data between UTeM health center in real time. It is expected to prevent data lost due to database failures. Upon any database failure, the system will automatically restore the data lost into the database. This would reduce the time and resource for admin to configure and make backup and restore from the database as well. This system will be developed using Atom Editor, Python script, and also MySQL server. The methodology that has been used is the Waterfall model.

ABSTRAK

Objektif kajian ini adalah untuk mengenal pasti penggunaan data penyegerakan menggunakan Raspberry Pi untuk e-Clinical Support System. Sistem ini merupakan mekanisme pelayan sandaran automatik melalui middleware yang dapat menyegerakkan data antara pusat kesihatan UTeM dalam masa nyata. Ia dijangka untuk mengelakkan data yang hilang akibat kegagalan dalam pangkalan data. Atas apa-apa kegagalan dalam pangkalan data, sistem secara automatik akan mengembalikan data yang hilang ke dalam pangkalan data. Ini akan mengurangkan masa dan sumber untuk admin untuk mengkonfigurasi dan membuat sandaran dan memulihkan daripada pangkalan data juga. Sistem ini akan dibangunkan menggunakan Atom Editor, skrip Python, dan MySQL. Metodologi yang digunakan adalah Model Air Terjun.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter I

Introduction

1.1 Introduction

In this chapter, the problem statement about will be described in detail in section 1.2. Afterwards, the project objective will be discussed follow by the scope of the project in section 1.4. Next, the project significant and lastly the expected output will also be discussed in detail.

UTeM e-health information synchronization using Raspberry Pi is a system that will synchronize all the data from the UTeM clinic's database in real time. Raspberry Pi is to use in replication technique for back up data and it is able to be done automatically and real-time. This would reduce the time and resource for admin to configure and make backup from the database. Upon any database failure, the Raspberry Pi would automatically restore the data lost into the application database.

1.2 Problem statement

Current system in UTeM is being setup without backup feature. All of the ECSS are depend on the only main server and get data from it. This can cause many problems once the server is disconnected.

- Unable to fetch patient information upon any database failures. Thus the staff has to use manual filling instead.

- Tedious manual backup technique will consume time and resources.

- All the patient information is not being distributed correctly between different UTeM health centers in different campus when the server is disconnected from the network since the data are unsynchronized.

1.3 Objective

- To develop an automated backup server mechanism via a middleware that able to synchronize data between UTeM health center in real time.

- To study and explore the potentiality of using Raspberry Pi as a database server.

- To prevent the client application unable to fetch data from server when the main server is down.

1.4 Scope

- For UTeM clinic

  The propose system is designed to be installed in UTeM server in order to synchronize the data from UTeM clinic's database.

1.5 Project Significance

The data is important to clinic because it is all the medical records for patients. It can be very useful for doctor to get to know the medical information of the patients and make the right decision. So it will be very troublesome if the data lost upon any database failures. This system can prevent the data lost in the same time save the time to operate backup manually by the staff. This system is hoped to ensure continuous access to real database since the middleware will acts as secondary database.

1.6 Expected Output

The system is expected to synchronize the data from UTeM clinic's database. Upon any database failure, the Raspberry Pi will automatically and temporarily replace the master or main database. Thus other applications that is using the main database will connect to the Raspberry Pi until the main database is up and running. It can prevent data lost due to database failures by restoring the data from the backup. The synchronization will be done automatically once there is new data to be inserted or modified. The system will make sure the data in the system is always be the latest data.

1.7 Conclusion

This chapter focus about the brief introduction to UTeM e-health information synchronization using Raspberry Pi, objective and problem statement of project, and expected result from this project. Literature and journal of previous researcher will be discussing and review in the next chapter.

Chapter II

Literature review and Project methodology

2.1 Introduction

In this chapter, literature review and methodology will be discussed in detail. Facts and finding section will discuss the domains which related to the proposed system followed by the other approaches which also applicable to the system. Next, the methodology and all the requirements used in this system will be described. Lastly, the milestones will be stated.

2.2 Facts and Finding

2.2.1 Raspberry Pi

The Raspberry Pi is a credit-card size computer from Raspberry Pi foundation which cost less than $40, and it offers a computer can interact with other components such as sensors and LED light bulb to create some creative and useful product. It has been used in many sectors of digital maker projects, from music machines to weather stations and cat feeder.

Raspberry Pi can be a desktop PC replacement while there are some limitations due to its architecture. It is a good machine to enable people of all ages to explore computing, learn about how to get to grips with computers since the price is hugely accessible. With the official Raspbian OS, the Pi is loaded with tools for learning how to program, from the drag-and-drop coding offered by Scratch and various guides for writing and debugging the programming language Python.

Raspberry Pi is also equipped with 26 general purpose input/output (GPIO) pins, electrical channels that allow the Pi to communicate with other devices or other components such as sensors, etc. It can get the input from a sensor and produce output from turning on an LED to sending a signal to another device.

Setting up the Pi is slightly more complicated than the average computer desktop. There are some easy to follow guides online, and NOOBS (New Out-Of-Box Software) installer makes getting the Pi up and running easier. NOOBS can install various operating systems such as the Pi's official Raspbian OS, Window 10 IoT, etc. The Pi is expected to function as a basic desktop PC, from a word processor to web browser.

2.2.2 Database Replication

Replication enables data from a database server to be copied to one or more database servers so that all database servers share the same level of information. The main database server will be the master server and the rest will be the slave servers.

The replication is asynchronous by default. Slave servers do not need to connect to the master server all the time to receive the updates. All the updates will be stored in a binary log file in master. The slave will search the binary log file from the master and perform the update once it is connected to master.

2.2.2.1 Advantages

There are some advantages of using database replication:

- Scale-out solutions
- Data security
- Analytics
- Long-distance data distribution

Scale-out solutions

Replication is dividing the load with multiple slaves to improve performance. All changes must take place on the master server while reads may be held on one or more slaves. In this way the performance of writes will be dramatically increased while the increasing number of slaves will improve the read speed.

Data security

Database replication will ensure there is always a backup on the slave and the backup services run without corrupting the corresponding master data.

Analytics

Using slave data to perform analysis can prevent to affect the performance of the master while all live data are generated in master.

Long-distance data distribution

Generate a local copy of the data from the master if there is a branch office wish to work with a copy of main data. This model can prevent them from requiring permanent access to the master.

2.2.3 Technique

2.2.3.1 Database mirroring

Database mirroring is a recovery technique for increasing the availability of database which is used by Microsoft SQL Server. The purpose of using database mirroring is to prevent any data corruption or loss when the operation of a network is partially compromised.

Database mirroring involves two SQL Server instances on the same or different machines. The server acts as a primary instance is called as the principal while the rest server is known as the mirror.

Since the DBMS of UTeM clinic is using MySQL, database mirroring does not seem to support DBMS other than SQL Server. This is the reason dabase mirroring did not be used in this system. Furthermore, database mirroring will be removed from SQL Server in future versions.

2.2.4 Existing System

i. *IM020-09 (June 2009), An Automated System for Patient Record Management*

The automated system for patient record management is designed for St Francis Hospital Nsambya hospital which aims to solve the loss of patient and staff records. At the time, paper files consume a lot of the office space, slow recording, processing and retrieval of patients details. Accessing and sharing of information by various departments is tough due to poor data management.

It is a web base system developed in PHP with MySQL as database management system. The system provide backup feature will make database backup every day to prevent database failure.

ii. *Xiuju Zhan, Xiufeng Liu (October 2013), Design and Implementation of Clinic Appointment Registration System*

Clinic Appointment Registration System is a system developed in C#.Net with Microsoft Access 2003 as the

database to store medical data. The system consists of various data operation functions including appointment registration, data backup and recovery, etc.

The system provides data backup and restoration mechanism in order to improve system security since the data corruption is unavoidable because of the chance of computer system hardware crash, software failure, operator error, and malicious damage.

2.3 Project Methodology

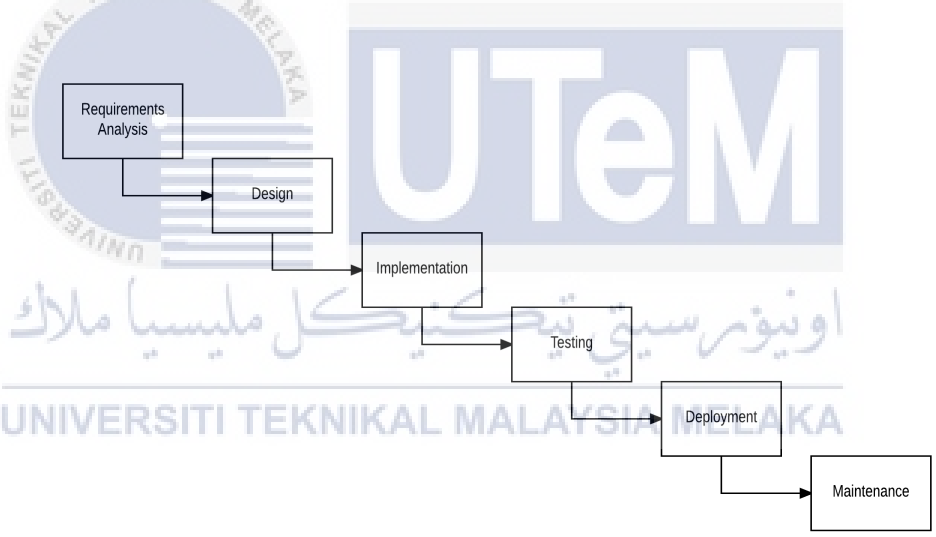

Figure 1: Illustration of Waterfall Model

Waterfall methodology is used in this project. Waterfall model is one of the popular system development life cycle (SDLC) models for software development. Winston W. Royse in 1970 was the first people to define this model. Waterfall is sequential design process. The current phase must be done before going to the next phase. Since it is sequential, once a step has been completed, developers cannot go back to the previous step.

The first phase of Waterfall model is the Requirement and Analysis phase. In this phase, all possible requirements of the proposed system to be

gathered and documented. These requirements are studied and are analyzed whether it is valid or invalid. In Chapter 1, the project title is set. The problem statements and the objectives are defined. The requirements are captured during the study and documented in Chapter 2. The literature review shows the idea of how to get started in the project and what other people is done in similar field. All the requirements including software requirement, hardware requirement is stated. Project schedule and milestone is listed. The analysis is done in Chapter 3. The problem of current system is analysed. The requirement of this project including functional requirement and non-functional requirement is introduced.

The second phase is the Design phase. All the gathered requirements from the previous phase are used to prepare system design. With the help from system design, the hardware and system requirements are specified, and the overall system architecture will be defined as well. In this project, the Design is in Chapter 4. The High-Level design of the system will be showed with the system architecture of the proposed system, scripting design, data design and etc. In system configuration section, the steps of configure both servers will be explained.

The next phase is the Implementation phase. In this phase, the system is starting to be developed. All the requirements and the specifications will be used and the actual code will be written by programmers. In Chapter 5, the software development environment setup section will explain all the software used in developing this system such as Atom Editor, MySQL. There will be some python scripts to handle the automation of switching the slave to replace master once master is disconnected to the network. The server setup will be stated as well in this chapter. The procedure of using version control will be showed in the next section.

The fourth phase is the Testing phase. After the system is developed, the system needs to be tested according to the designed test case. In this phase, the testing can be performed by the client to determine the defect of the system. The client also can provide review and feedback to improve the system. The

testing plan will be developed in Chapter 6. The testing plan will include test organization, test environment and test schedule. The test strategy will be explained and the test design will be described in the following section.

After the system is well tested, the system is deployed to the client environment in the Deployment phase. In this phase, the system is officially done and can be used in real world. In this phase the system will be deployed to UTeM.

The last phase is the Maintenance phase. In this phase, the client is using the system. As the problems are found due to some mistakes in the design phase or change in user requirements, the system will be improved and maintained in this phase.

2.4 Project Requirement

2.4.1 Software Requirement

- Atom editor
  Use to create and edit the python script

- Microsoft Word 2016
  Use to produce project report and other documentation

- Parallels Desktop
  Use to install Ubuntu 14.04 to emulate as master server to perform database replication

- MySQL
  Use to store data and perform database replication

- Lucidchart

Use to generate diagrams

2.4.2 Hardware Requirement

- MacBook Pro Retina
  Develop system and produce project report

- Raspberry Pi 2
  Use to act as slave server to perform database replication

2.4.3 Other Requirement

- **Network connection**
  Both master server and slave server have to be connected within same network.

2.5 Project Schedule and Milestones

Table 1: The project schedule and milestones

Week	Activity	Note/Action
1 22 Feb – 28 Feb	Discussion of Tittle and Proposal Submission	Deliverable - **Proposal** Action - Student
		Deliverable - **Proposal Presentation (PP)** Action – Student
	Proposal assessment and verification	Action – Supervisor