AUTOMATED TELLER MACHINE USING UMP SMART CARD IN DB2

WONG KAM MAN

UNIVERSITI MALAYSIA PAHANG

AUTOMATED TELLER MACHINE USING UMP SMART CARD IN DB2

WONG KAM MAN

A thesis submitted in partially fulfillment of the requirements for the award of degree of Bachelor of Computer Science (Computer System and Networking)

Faculty of Computer System & Software Engineering

Universiti Malaysia Pahang (UMP)

JUNE 2012

DECLARATION

I hereby declare that this thesis entitled "Automated Teller Machine using UMP smart card in DB2" is the result of my own research except as cited references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : .....

Supervisor : .....

Date : .....

SUPERVISOR'S DECLARATION

"I hereby declare that I have read this thesis and in my opinion this thesis is sufficient in terms of scope and quality for the award of the degree of Bachelor of Computer Science (Computer System and Networking)"

Signature :.....

Supervisor :.....

Date :.....

DEDICATION

Special dedicate to:

My beloved parents, Wong Sai Sooi and Kuan Lee Choo for their great support and encouragement. Their endless care and consultations taught me that never give up easily when facing any hard tasks because there is always a solution for the hardest task.

Dr. Mohamed Ariff bin Ameedeen as my supervisor for his great guidance and critics. He served as a great advisor in helping me to solve problems.

My friends

Thank you for your support and cooperation.

Sincerely,

Wong Kam Man

ACKNOWLEDGEMENT

Firstly, I am deeply indebted to many people who are directly or indirectly, responsible for this project coming into being. I am most grateful to the IBM experts, Mr.Shreekanth Kanagarajan and Mr. Gouthaman Jayaraman who get their space time and gave a lot of efforts in teaching me to develop this project using DB2 althought they are always busy.

I also would like to give a deep thank to Dr.Mohamed Ariff bin Ameedeen as my supervisor for his constructive criticism of my research in this project, and most of all his guidance and support for the whole project.

I am indebted to my lecturer, Ms. Junaida binti Sulaiman who taught me Object Oriented Programming by using JAVA programming for instilling me the basic knowledge of JAVA programming.

I am also indebted to my lecturer, PM.Ruzaini bin Abdullah Arshah who taught me Structure Query Language by using Oracle database for instilling me the basic knowledge of Structure Query Language and database logic.

Last but not least, a sincere thanks to my dearest parents and friends who endured this long process with me, always offering guidance and supports.

ABSTRACT

This document discusses about Automated Teller Machine using UMP smart card in DB2. In UMP, students, lecturer, or staff might bring more than one smart card with them and thus there is always make inconvenient for them. This project is developed to use an all-in-one smart card that can perform banking transaction and perform as student ID card or staff ID card as well. The proposed system will have two part, user interface and database. The database is installed in z/OS environment while the user interface is in Windows. To develop this proposed system, System Development Life Cycle (SDLC) has been chosen as methodology.

ABSTRAK

Dokumen ini bincang tentang *"Automated Teller Machine using UMP smart card in DB2"*. Di UMP, mahasiswa, lecturer and kerani mungkin membawa lebih daripada satu smart card dan inilah sentiasa menyebabkan kesusahan. Projek ini dibangunkan untuk penggunaan all-in-one smart card yang boleh melaksanakan transaksi perbankan dan digunakan sebagai matrik kad mahasiswa atau matrik kad kerani. Sistem yang dicadangkan dibahagi kepada dua bahagian iaitu user interface dan database. Database install di dalam persekitaran z/OS manakala user interface install di dalam Windows. Untuk membangunkan system yang dicadang ini, System Development Life Cycle (SDLC) telah dipilih sebagai metodologi.

TABLE OF CONTENTS

LIST OF TABLE

LIST OF FIGURES

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

This chapter briefly describes the Automated Teller Machine using UMP smart card in DB2. This chapter comprises five sections which are the background of the project, problem statements of the project, objectives and scope of the project.

1.1 Background of Proposed Study

The system that will be developed is Automated Teller Machine using Universiti Malaysia Pahang (UMP) smart card in DB2. It is a system where a multi-function smart card will be used on an ATM instead of using a normal smart card. Also, the system will use z/OS mainframe as the backbone core server to control the ATM system.

Automated Teller Machine (ATM) is actually a computer that is connected on a twenty-four hours real-time system to let end users perform banking transactions, make inquiries concerning the status of their accounts, pay bills and obtain other banking services in a public place. To use the ATM, each user needs to have his/her own ATM card that contains a PIN number to access their accounts.

Operating systems such as Windows or Linux are not sufficient to maximize the performance of the mainframe. The most suitable operating system for mainframe is

z/OS designed by IBM. Mainframe, famous for its heavy workload performance, is suitable for the backbone of Automated Teller Machine.

In UMP, everyone needs a matric card as the passport to enter the administrative offices, lecture halls, labs, and library. The matric card has no other functions except as mentioned above. These also implied on our Identity Card (IC). Our IC doesn't have any functions although it is a smart card which contains a chip on it.

1.2 Problem Statements

For the current system, end users need to bring a specific ATM card for a specific ATM, so the end user might end up having a lot of cards. Current smart cards such as Identity Cards (IC) or Student Cards have no any function. It is kind of a waste as they do not serve any purpose.

1.3 Research Objectives

The objectives of the proposed system are listed as the following:

1.3.1 To implement an all-in-on ATM card system that can be used to withdraw cash and check inquiries.

1.3.2 To make it convenient for end users so that they do not need to bring more than one card.

1.4 Scope of Study

The users of the proposed system are UMP students, staff and lecturers. UMP students, staffs and lecturers withdraw cash from ATM, use the top-up service ATM provide, e-payment, and use their smart card to enter administrative offices such as JHEPA, faculties, lecture halls and labs, and the library. There are some limitations as the proposed system equipped with cash withdrawal and check balance inquiry functions however just for testing purposes only.

CHAPTER 2

LITERATURE REVIEW

This chapter briefly describes the reviews on existing techniques related with Automated Teller Machine using UMP smart card on z/OS environment that will be developed later. This chapter comprises of history of Automated Teller Machine, Security of ATM, Cost issue of ATM, Types of Smart Card, Security of Smart Card, Advantages and Opportunities, Databases and Computers.

2.1 History of Automated Teller Machine (ATM)

As said by Bernardo Batiz-Lazo and Robert J.K. Reid (2011), Cash dispensers or Automated Teller Machine (ATM) in the United Kingdom emerged independently and in parallel with similar technologies in Japan, Sweden, and the United States. The Committee of London Clearing Banks (CLCB) formed an Electronics Subcommittee, which had a mandate to research possibilities of brining electronic computing to British banking and was pivotal in establishing the banks that would become early computer pioneers in 1955. Barclays Bank, Britain's biggest in terms of deposits, opened the first

dedicated center to house a mainframe computer in 1961. Soon, the first ATM was introduced at Barclays Banks in Enfield, United Kingdom in June 1967.

On most modem ATMs, the customer needs to insert a plastic ATM card with a chip or a magnetic strip that contains a unique card number and some security information such as a PIN number to access his/her banking account. ATM security is one of the issues that cause concern.

2.2 ATM Security issues

Security of ATM is an issue that always concerned by people. After we put in our card into the ATM, we are prompted to input a PIN before the cash withdrawal is procedure. As Coventry L., Angeli A. D. and Johnson G. (2003) mentioned, many people have PINs and password for a multitude of devices, from car radio and mobile phone, to the computer, web-based services and their bank information. Soon Kim C. and Mun-Kyu L. (2010) also noted that Personal identification number (PIN) is a common method to authenticate a user for various devices including ATMs, mobile phones, PDAs, door locks and so on. Normally we will choose to our own password or PINs for those devices. Coventry L. et al. (2008) stated that if people are permitted to choose their own passwords they tend to use the one which are easily guessed. Normally the passwords that people choose are related to their everyday life. The passwords are easy to remember, easily predicted, or some people might change all PINs to be the same. Moncur W. and Leplatre G. (2007) also said that PINs are easily forgotten and users circumvent the forgetfulness by write down theirs PINs, make the PINs all the same, or disclose them to friends and family. This make the PINs security is weak. Munjal N. and Moona R. (2009) noted that a fake outlet can acquire user's details from their magnetic card and even take their PIN without dispensing services. Soon Kim C. and Mun-Kyu L. (2010) stated that if someone observes the input procedure by looking over a user's shoulder or using a tiny camera, he can easily obtain the PIN. This kind of attack is called 'Shoulder surfing attack (SSA). According to Mohammed A.M.

Abdullah, F.H.A. Al-Dulaimi, Waleed Al-Nuaimy and Ali Al-Ataby (2011), a user gains access to a card if he/she enters the right PIN. These shown that our PINs or private information might be obtained by attacker. Also, attacker might withdraw money from the victim's account after getting the PINs.

It is very clear that, awareness is needed. Moncur W. and Leplatre G. (2007) stated that awareness is emerging of the need to design authentication with usability. There are many options to design the authentication, include biometrics, personalization and behavior, and graphical passwords. Coventry L. et al. (2008) noted that biometric techniques can confirm that a person is actually present without requiring the user to remember anything. While Moncur W. and Leplatre G. (2007) mentioned that graphical passwords offer greater usability, potentially greater security than knowledge-based passwords. Soon Kim C. and Mun-Kyu L. (2010) defined certain security aspects which are complexity of a random guessing attack, Resilience to human shoulder surfer, and Resilience to recording attack. Complexity of a random guessing attack is the number of choices an attacker faces when he tries to pass the PIN entry test by a random guessing attack. Resilience to human shoulder surfer is the amount of information that a human attacker without any recording device must memorize to recover PIN. Resilience to recording attack is the size of a set for PIN candidates consistent with the current transaction when the whole procedure is recorded by a camera. Munjal N. and Moona R. (2009) said that two levels of authentication, first using Public Key Infrastructure (PKI) based authentication and second using biometric trait like a fingerprint. They believe that this scheme prevents card forgery and phishing attacks. Cooke J.C. and Brewster R.L. (1993) mentioned that the security requirements in personal communication system are twofold; they are authentication and protection of information.

2.3 Cost Issues of ATM

Another issue of current ATM is cost issue. As mentioned by Munjal N. and Moona R. (2009), Network connectivity is indispensable to the working of the current

transaction model. Hence, the network component contributes heftily to the run time costs of the model. They also noted that unreliable network as another issues of ATM. Financial services that use the network, claim round the clock availability. Dedicated network connectivity is a possible downfall of the current model not only because it is difficult to achieve in distanced rural areas but also because networks are said to be inherently unreliable. Lastly, Burden of carrying multiple tokens is one of the critical issues of ATM. Users often carry multiple authentication tokens such as a magnetic stripe card, smart card, RF card etc. for multiple services like credit account, debit account etc. and for each of the subscribed financial institution and cause a lot of overhead for the users.

2.4 Types of Smart Card

Attoh-Okine N. O. and David Shen L. (1995) specified two types of smart cards, the contact-type or the contactless type while Katherine M. S. and J. Drew Procaccino (2002) categorized the smart card as either a memory card or a processing-enabled card. Both statements are actually correct. A smart card can either be a memory card or a processing-enabled card, as well as either contact type or contact-less type. Contact card requires physical contact between the card and the reader while contactless card is either close-range or long-range, reliable and more expensive compared to a contact card. According to Katherine M. S. and J. Drew Procaccino (2002), the simplest form of smart card is memory card, with limited capability to securely store personal information. Next is prepaid card, which transfer the electronic equivalent of cash to a vendor's digital cash register. The processor-enabled smart card is which based on semiconductor technology with a smart chip with few hundred bytes of RAM. The memory capacity of processor-enabled smart card to function as a multi application card, combining many functions such as credit card, debit card, stored value card, information management card, and loyalty card. They also mention that smart card has nonvolatile, read-only memory (ROM), random access memory (RAM) and central processing unit (CPU). Arafatur

Rahman et al. (2008) also state that smart card have electrical contacts and a thin metallic plate, which is an integrated circuit chip (IC) that containing a central processing unit (CPU), random access memory (RAM) and non-volatile data storage. Also, Schaefer R., Mueller W., Lopez A. M. and Sanchez D. D. (2007) state that the common of SIM cards for mobile phones, credit card sized contact or contactless cards, and card in USB connectors is their limited processing power and data storage.

Similarly with the current ATM system, the smart card that will be used in my proposed system is the contact type smart card which is a memory card that can store banking account ID, PIN number and some personal information such as name and Identity Card number.

Feature Component	Smart Card	
	Memory Card	Processor-Enable Card
Read Only Memory?	yes	yes
Random Access Memory?	no	yes
Microprocessor?	no	yes
Contact/Contactless Interface	contact, contactless or both	contact, contactless or both
Data certified secure (ITSEC*)?	no	yes
Example	phone card	multi-application cards

* information Technology Security Evaluation Certification represents a set of software and hardware security standards that have been adopted in Europe and Australia.

Figure 2.1 Memory Card versus Processor-Enable Card

2.5 Security of Smart Card

Basically, security of smart card can be categorized in authentication, authorization and transaction processing. As mentioned by Katherine M.S. and J.Drew Procaccino (2002), there are three categories of smart card applications which are authentication, authorization, and transaction processing. The security requirement of smart cards is authentication and protection of information (Cooke J.C. and Brewster

R.L., 1993). Attoh-Okine N.O. and David Shen L. (1995) also discussed the security requirement of smart cards in terms of account verification, user identity verification, information access restriction and prevention of card attemping. Financial service outlets need authentication, integrity, confidentiality, non-replication and non-repudiation (Munjal N. and Moona R., 2009).

We can conclude that the security of smart card mainly focus on authentication, protection of personal information and also information access restriction. ATM system needs very strong security to protect the user and banking transactions. Therefore, security of smart card needs to be frequently upgraded to always protect and secure each transaction and user information.

In my proposed system, to recognize users, we need to enter a correct PIN number which matches the PIN number stored inside the smart card. If the PIN number entered is wrong at most three times, the card will be temporarily blocked by the system and no banking transaction is allowed with the card.

2.6 Advantages and Opportunities

By using the smart card, Attoh-Okine and David Shen L. (1995) said that the flexibility of a system's structure, accountability and security and convenience for users might be improved. As mentioned by Katherine M. S. and J. Drew Procaccino (2002), new traffic offences could be updated to a person's smart card within minutes of the offense. The smart card also has the potential to facilitate storage of demographic information for voting purposes. This is clear that by using the smart card, the effectiveness of the system would be improved. Using smart card benefits to the security of personal communication systems (Cooke J.C. and Brewster R.L., 1993) which means, the smart card improves the security of the systems as mentioned by Attoh-Okine and David Shen L. (1995).

2.7 Database

Database in mainframe is using DB2 while normal server using MySQL or Oracle as the database. Here, the different between DB2, MySQL and Oracle will be reviewed regarding the security, price, features, availability of platform, backup and recovery, and storage.

2.7.1 Security

Database such as Oracle or MySQL is normally used in Windows or UNIX system while DB2 is specially designed for z/OS mainframe. According to Trivedi G. (2010), Oracle server has a Database Management System (DBMS) that controls the stocking of data, recovering of data through adequate optimization techniques, security of the databases and tasks allowed for particular users and consistency and protection of data, including task archiving and search engines. While MySQL uses three parameters to authenticate a user namely user name, password and location, Oracle uses so many security features like username, password, profile, local authentication, external authentication, advance security enhancement and etc. (Shekhar R., 2011) Ebbers M. et al. (2009) noted that DBMS in z/OS is able to put confidential or sensitive data in a separate segment or table while in a Partitioned Data Set (PDS) or Visual Storage Access Method (VSAM) flat file, the application program gets access to every data element in the logical record.

2.7.2 Price

When we want to choose a suitable database server, price is one of a factor affects our decision. Trivedi G. (2010) mentioned that Oracle is more costly than DB2 although Object wise comparison is same in both RDBMS which mean triggers, functions, tables, bitmap indexes, tree indexes, PL/SQL and etc are in same in both Oracle server and DB2. On the other hand, MySQL is an open source database, and is completely free. (Shekhar R., 2011).

2.7.3 Features

According to Trivedi G. (2010), IBM DB2 UDB was specially created to support features of Business Intelligence directly in the database. These abilities include data mining, ETL, OLAP and other advance space features of analysis and statistics. Shekhar R. (2011) noted that MySQL database does not support any feature like Audit Vault on its server while Oracle, supports several extensions and programs on its database server for instance, Active Data Guard, Audit vault, Partitioning and Data Mining.

2.7.4 Availability of Platform

According to Trivedi G. (2010), Oracle server is almost available for every operating system like Windows, Unix or Linux. MySQL is almost same as Oracle server,

which has same available for every system. Trivedi G. (2010) also mentioned that DB2 database is also available for every operating system like Windows, Unix and Linux.

2.7.5 Backup and Recovery

Shekhar R. (2011) said that Oracle provides different type backup facilities such as cold backup, hot backup, export, import and data pump. Oracle offers most popular backup utility called Recovery Manager (RMAN) while MySQL has mysqldump and mysqlhotcopy backup utilities but there is no utility like RMAN in MySQL. As said by Ebbers M. et al. (2009), DB2 has a COPY utility to recover data and there is MERGECOPY utility to merge incremental copies with a full copy. RECOVER utility in DB2 can recover back to an image copy for a point-in-time recovery.

2.7.6 Storage

As said by Shekhar R. (2011), MySQL doesn't have Table space, Role management, snapshots, synonym and packages compare to Oracle. In contrast, Oracle has Table space, Role management, snapshots, synonym and packages. On the other hand, Ebbers M. et al. (2009) mentioned that DB2 has a storage group consists of a set of volumes on disks (DASD) that hold the data sets in which tables and indexes are actually stored. DB2 also has view, Table space and Index space.

2.8 Computers

According to Roper M. and Millar L. (1999), a computer is a programmable machine. There are several different types of computer such as Mainframe computers, Mini-computers, workstations and Personal computers. Here, the different between Personal computers and Mainframe computers will be reviews regarding its definition and functions, components and operating system.

2.8.1 Definition and functions

According to Ebbers M. et al. (2009), mainframe means computers that can support thousand of applications and input/output devices to simultaneously server thousands of users. Vigil J. and Price J. (2002) said that mainframe is used connect multiple users for large organization while personal computers are generally used for a single users. Roper M. and Millar L. (1999) also noted that personal computers (PCs), also called microcomputers, are the most popular type of computer in use nowadays.

2.8.2 Components

Vigil J. and Price J. (2002) said that both mainframe and personal computer have processors, storage, memory, operating systems and displays. Roper M. and Millar L. (1999) also noted that computer are made up of two parts which are hardware and software. Hardware components are central processing unit (CPU), memory, storage device, input devices, output devices, random access memory (RAM), storage, optical disk, hard drive and magnetic tape.

2.8.3 Operating System

According to Ebbers M. et al. (2009), mainframe operating systems are sophisticated products with substantially different characteristics and purposes. Beside z/OS, there are four other operating systems dominate mainframe usage which are z/VM, z/VSE, Linux for zSeries and z/TPF. Vigil J. and Price J. (2002) said that operating systems are different between the computer and mainframe. The mainframe's operating system is more complex because it is for the multiuser support which needs extra security and stability while personal computer's operating system is simple compare to mainframe because it is just for personal use which needs a normal security and stability. Personal computer always loads with Windows, Linux or Unix operating system. Vigil J. and Price (2002) also mentioned that, resetting a mainframe should not happen as often as a personal computer because a mainframe will affect hundreds to thousands of users while personal computers might just affects a few users.

2.9 Conclusion

Based on a thorough review of the literature, I have decided to use a smart card which is a contact type memory card that can perform multiple functions like banking transactions and as a student matric card which functions like a passport inside Universiti Malaysia Pahang (UMP). By using the multiple functions of the smart card, current system's structure, accountability, security, and convenience for users will be greatly improved. DB2 will be chosen as the backbone core server because of the mainframe's stability and security compare to normal server and personal computers, and also mainframe's performance which can support up to thousand of applications and input/output devices to simultaneously server thousands of users.

CHAPTER 3

METHODOLOGY

This chapter focuses on the main part of the system because it has the design, prototype, research, and related software and hardware for Automated Teller Machine using UMP Smart Card in DB2.

3.1 System Development Life Cycle (SDLC)

Methodology is a standard process to conduct all the necessary steps to analyze, design, implement, and maintain information systems. For the purposed system, System Development Life Cycle (SDLC) has been chosen as the methodology. SDLC is the overall process of developing, implementing, and retiring information systems through a multistep process from initiation, analysis, design, implementation and maintenance to disposal. SDLC has three primary objectives which are ensure that high quality systems are delivered, provide strong management controls over the projects, and maximize the productivity of the system staff. There are seven phases which are Planning, Analysis, Design, Development, Testing, Implementation and Operation and Maintenance will be used to develop this system.

3.2 The Justification Choosing System Development Life Cycle (SDLC)

Each system works so well individually because it is composed of a rigorous set of tasks which result in well-defined outputs. The precision and completeness of the task lists, data, and division of responsibilities are required because of the complexity of this system. To create system, we need a system to do system and SDLC is the system that we need to create a system. SDLC is the system to use to build and maintain software systems.

SDLC is needed in the proposed system because the development process is composed of many complex tasks which must be done in the right order to produce a successful result. In the proposed system, the Automated Transaction Machine system has a long life which will outlive the development. SDLC provide continuity to the proposed system which might have further implementation and development in the future.

3.3 The Steps of System Development Life Cycle (SDLC)

There are seven phases in SDLC:

- Planning
- Analysis
- Design
- Development
- Testing
- Implementation
- Operation and Maintenance

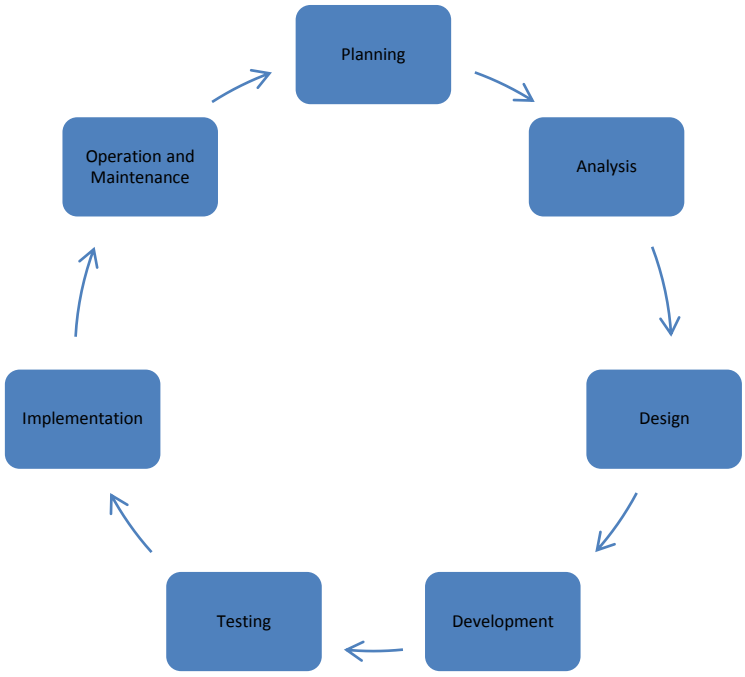


Figure 3.1: System Development Life Cycle (SDLC)

3.3.1 Planning

First phase in System Development Life Cycle (SDLC) is planning phase which identify the scope of the project and develop a schedule to allocate resource. Planning also include identifying the objective and problem statement. By identify all the scope, limitations, objective and the problem statement then we need to search all the information, documentations that related to our system and lastly we can produce a schedule for the work and allocate the resources. By using the gantt chart, all the tasks and jobs can be clearly divided and can be done in time and date.

3.3.2 Analysis

The second phase of System Development Life Cycle (SDLC) is Analysis. There are three parts to analysis which is determining requirement, structuring requirement and design strategy. We need to analyze end-user needs, identify the functioning of the proposed system, and research about current system by searching internet, comparing the documentations such as articles, information and journals. Analysis is very important because we must identify the main point of developing a new system and we also need to make sure the possible reason to make a new system. The possible reasons of making a new system is because we want to maximize the use of smart card, which is the matric ID card, in UMP, to reduce the inconvenient of bringing many cards with end user, and to reduce lost of the important smart card.

Therefore, an Automated Teller Machine using UMP smart card in DB2 will be developed.

3.3.3 Design

Design is the third phase of System Development Life Cycle (SDLC). Design including design the functions, features and operations. Functions, features and operations are described in details, including graphical user interface, process diagrams and other documentations. A new system will be described as a collection of modules or subsystems.

Design elements will be produced. All the design elements describe the desired software features in detail include functional hierarchy diagram, tables of database, process diagrams, flowchart, entity-relationship diagram and data flow diagram. These elements describe the system in sufficient details to help the skilled programmers to develop the system with minimal additional input design.

3.3.3.1 User Design

The user design includes the diagram such as context diagram, data flow diagram, flow chart and etc.

Figure 3.2 shows the overall simple relationship of the proposed system, Automated Teller Machine using UMP smart card in DB2. The end user who inserts the UMP smart card into the Automated Teller Machine (ATM) will need to key in their PIN number. Smart Card Reader will read the information of the smart card. After that end user can choose either to withdraw cash from the ATM or check the balance inquiries. Data is store in a database in a mainframe and a daily transaction report will be sent to the bank manager.

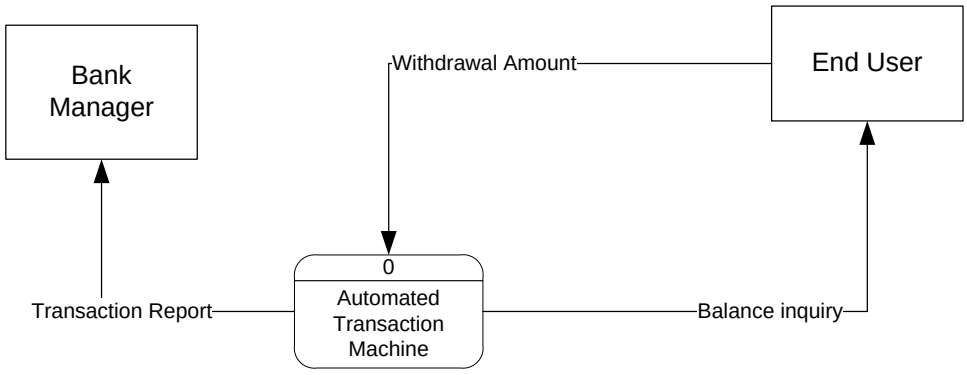


Figure 3.2 Context diagram of Automated Teller Machine using UMP smart card in DB2.

The flow chart is a type of diagram that represents an algorithm or process showing the steps as boxes of various kinds, and their order by connecting the boxes with arrows. Flowcharts are used in analyzing, designing, documenting or managing a process or program. Figure 3.3 show the flow chart for Automated Teller Machine using UMP smart card in DB2.

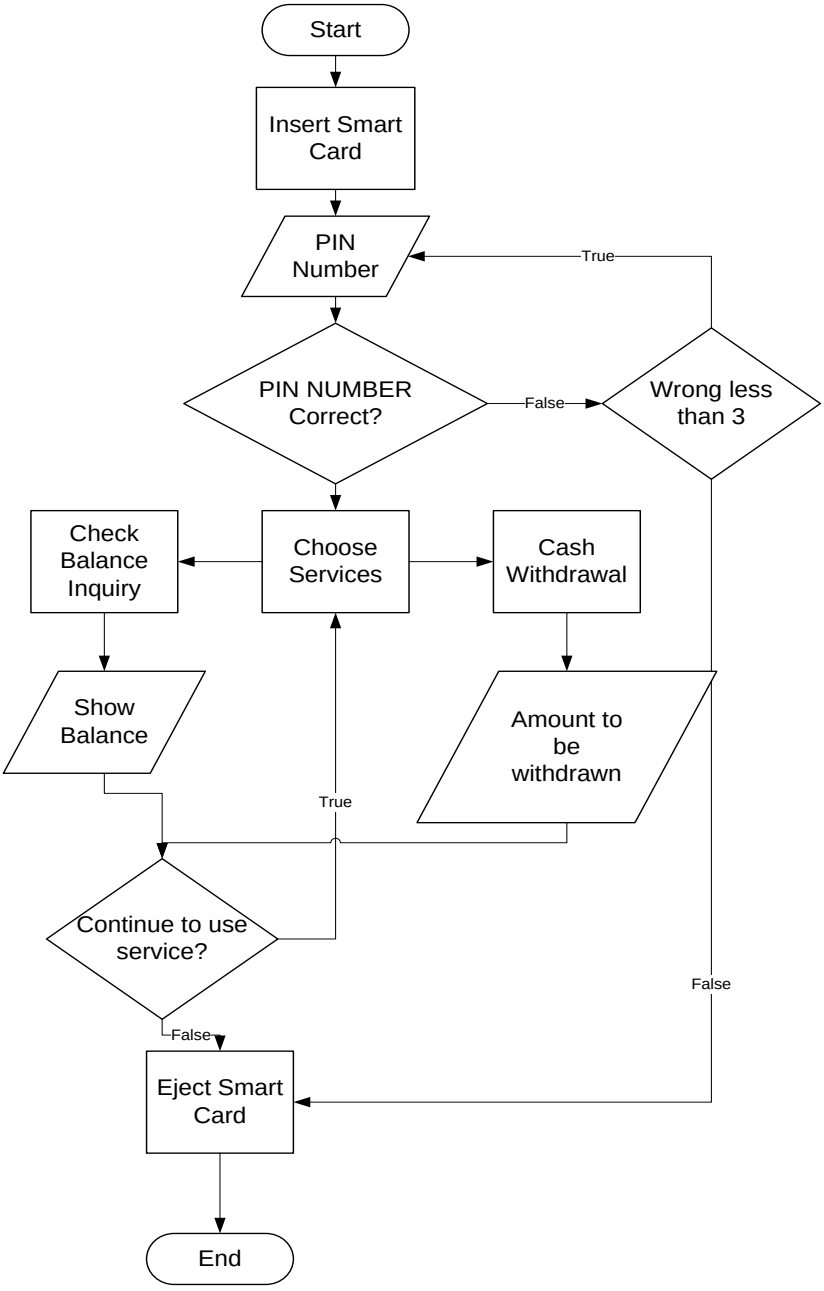


Figure 3.3 Flow Chart for Automated Teller Machine using UMP Smart Card

A data flow diagram is a graphical representation of the flow of data through a system. Data items flow from an external data source or an internal data store to an internal data store or an external data sink, via an internal process. Data flow diagram show all the relationship and interactions among the system, component and user in more details.

From the first, end users put their smart card into the Automated Teller Machine (ATM) and key in the PIN number. ATM will identify the PIN and retrieve the user information at the user database. After that user can choose either they want to check balance inquiries or withdraw cash.

If the user chooses to check the balance inquiry, system will retrieve the balance inquiry data from the account database and show the data to the user.

If the user chooses to withdraw cash, user need to key in the amount they want to withdraw, then system will update the withdrawal history at the withdrawal history database. After that, a new balance inquiry will be updated.

At last, daily transaction report will be retrieved from both account database and history database and send to the bank manager.

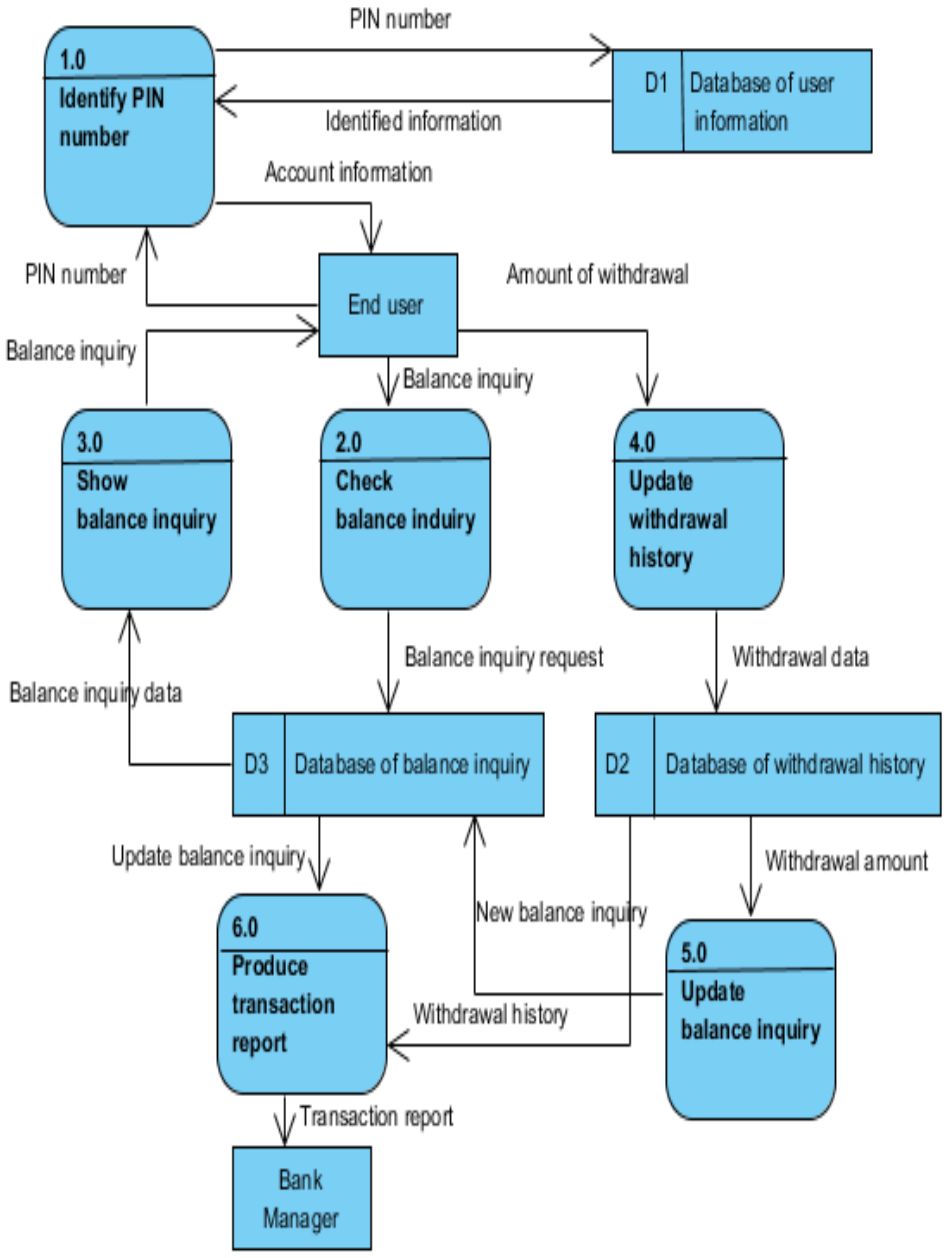


Figure 3.4 Data Flow Diagram of Automated Teller Machine using UMP smart card in DB2

Use case diagram is a type of behavioral diagram defined by and created from a use case analysis. Use case diagram is used to present a graphical overview of the functionality provided by a system in terms of actors, goals and any other dependencies

between those use cases. We can clearly depict the roles of the actor in the system via use case diagram.

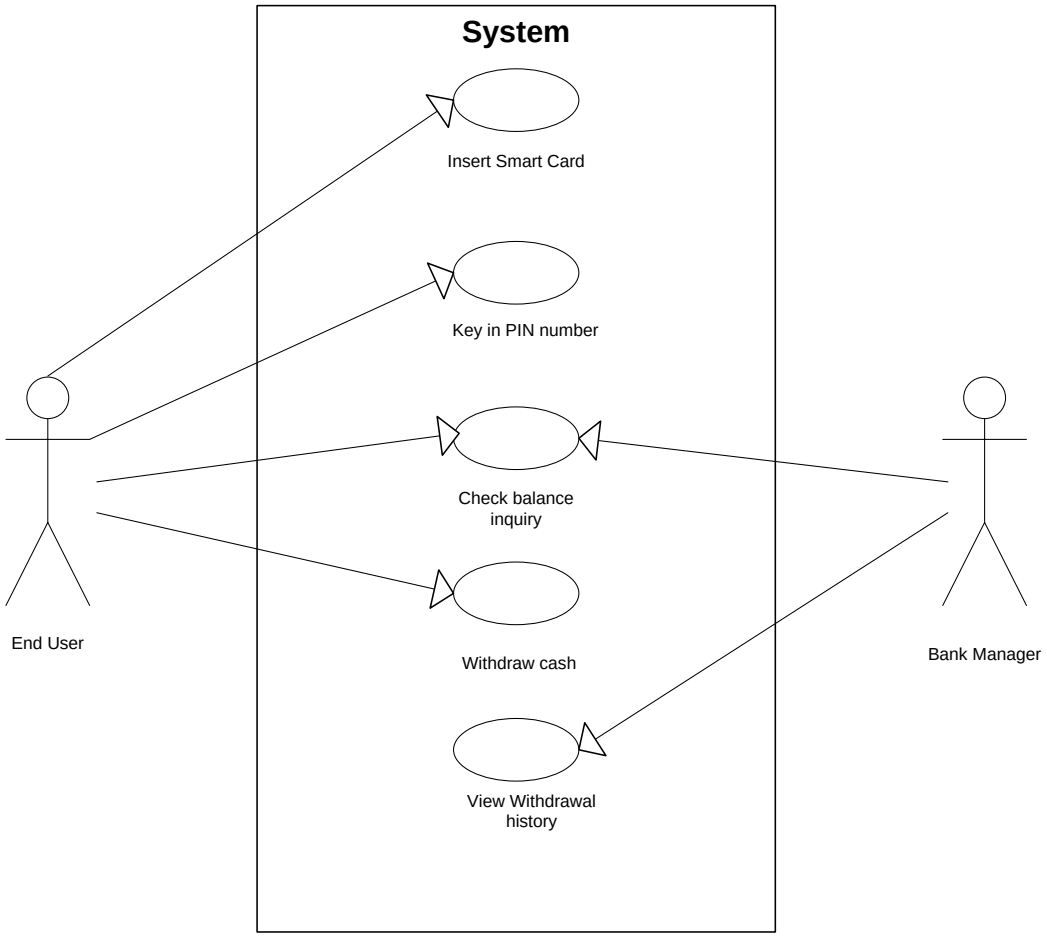


Figure 3.5 Use Case Diagram for Automated Teller Machine using UMP smart card in DB2

3.3.3.2 Database Design

Database design is the process of producing a detailed data model of a database. This data model includes all the needed design and parameters to generate a design in Data Definition Language, which can be used to create a database.

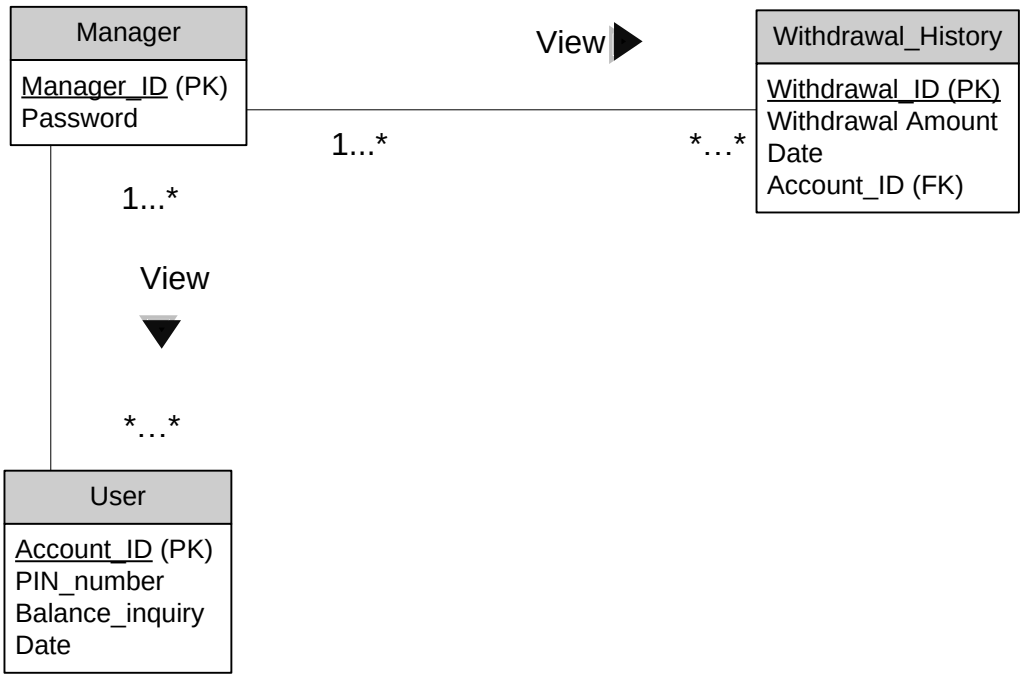


Figure 3.6 Entity Relationship diagram for Automated Teller Machine using UMP smart card in DB2

3.3.3.2.1 Data dictionary

Data dictionary is a document describing a database or collection of databases. Table 3.1 shows the data dictionary of Table Manager in the database. Manager_ID which is Primary Key and the Password is one of the attributes.

Field Name	**Description**	**Data Type**	**Constraint**
Manager_ID	ID for manager to login	VARCHAR(10)	Primary Key
Password	Password for manager to login	VARCHAR(25)	

Table 3.1 Data dictionary for Table Manager

Table 3.2 shows the data dictionary of Table User. In Table User, there are Account_ID which is primary key, PIN_number, Balance Inquiry and Date.

Field Name	**Description**	**Data Type**	**Constraint**
Account_ID	ID for User's account	VARCHAR(15)	Primary Key
PIN_number	Password for user to key in on ATM	NUMBER(10)	
Balance Inquiry	User's Balance	NUMBER(10,2)	
Date	Date when Balance updated	DATE	

Table 3.2 Data dictionary for Table User

Table 3.3 shows the data dictionary of Table Withdrawal_History. In Table Withdrawal_History, there are Withdrawal_ID, Withdrawal_Amount, Date and Account_ID.

Field Name	**Description**	**Data Type**	**Constraint**
Withdrawal_ID	To record each withdrawal	VARCHAR(15)	Primary Key
Withdrawal_Amount	Amount of Cash that withdrawal	NUMBER(10,2)	
Date	Date when withdrawal made	DATE	

Account_ID	Refer to user who make withdrawal	VARCHAR(15)	Foreign Key

Table 3.3 Data dictionary for Table Withdrawal_History

3.3.3.3 Interface Design

The Automated Teller Machine using UMP smart card in DB2 need interface design to let UMP students, staffs and lecturers to use the system. In the system, there are four main interfaces and additional interface might be added.

The first interface is PIN number entry interface which shown by Figure 3.7. This interface lets user to enter their own correct PIN number before they can use the system.

The second interface is main menu interface which shown by Figure 3.8. This interface lets user to choose either they want to check their balance inquiry or cash withdrawal.

The third interface is balance inquiry interface which shown by Figure 3.9. This interface lets user to check their balance inquiry for their account.

The fourth interface is cash withdrawal interface which shown by Figure 3.10. This interface lets user to choose the amount they want to withdraw.

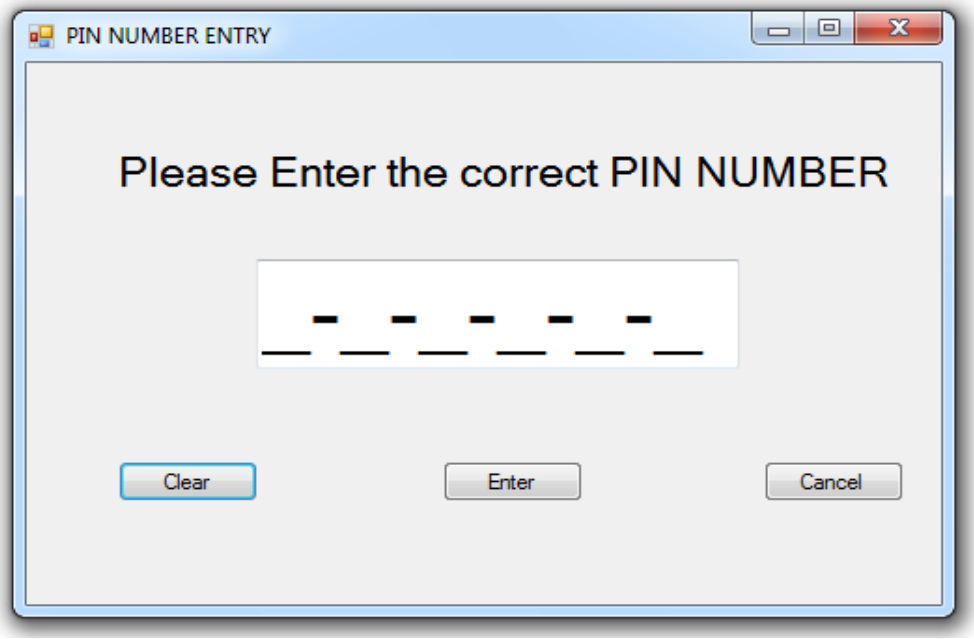


Figure 3.7 PIN Number Entry Interface

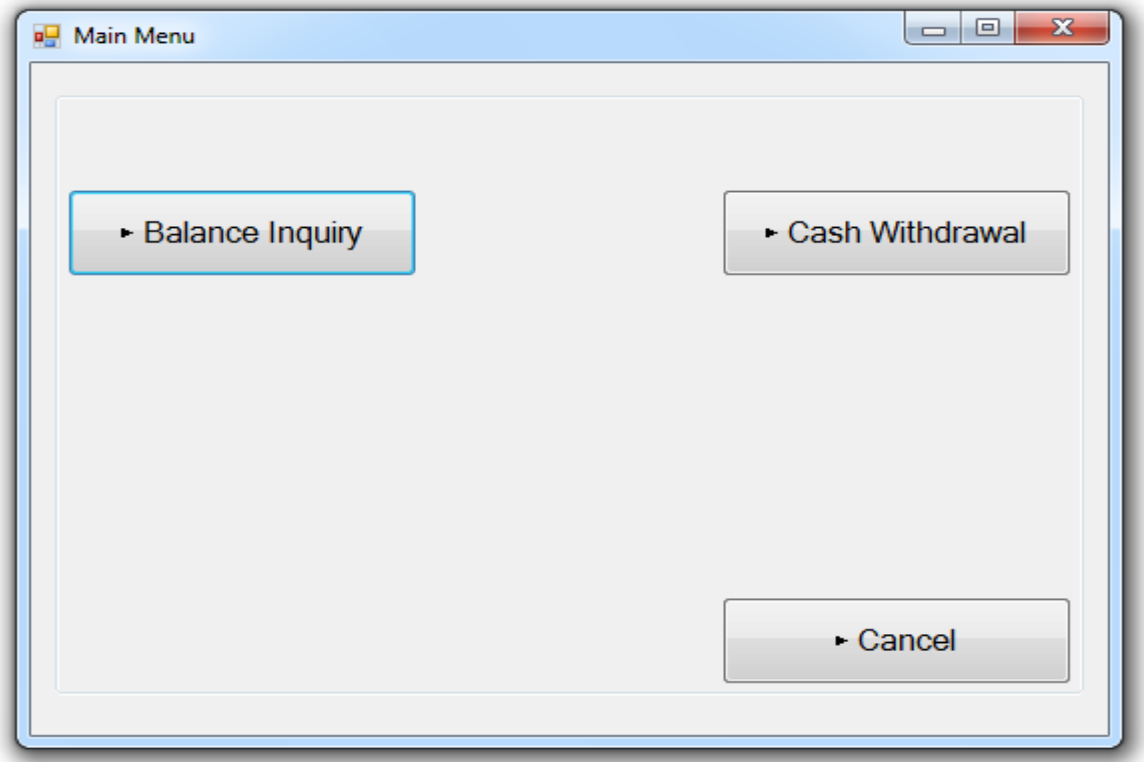


Figure 3.8 Main Menu Interface

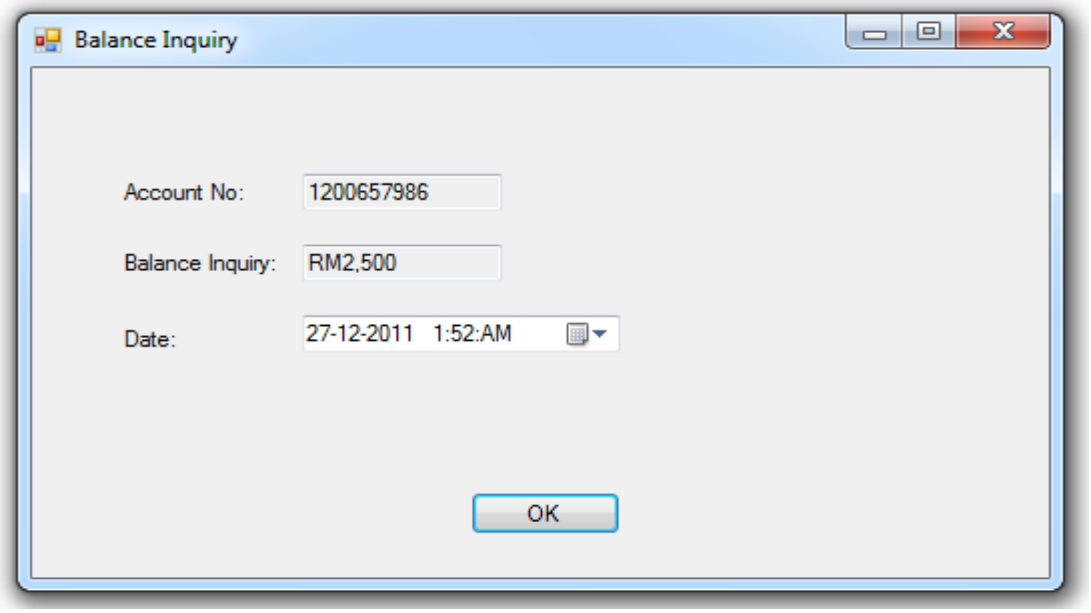


Figure 3.9 Balance Inquiry Interface

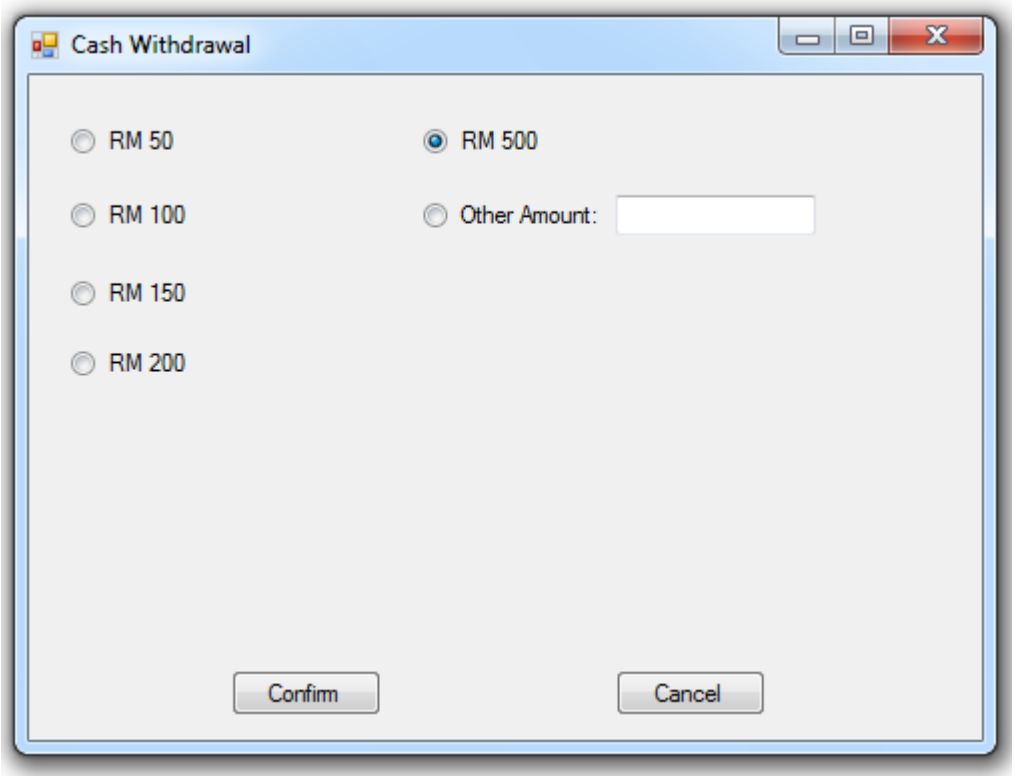


Figure 3.10 Cash Withdrawal Interface

3.3.4 Development

Development is the fourth phase on System Development Life Cycle (SDLC). Development turns the system specification into a functional system by using development tools. In this stage, all the modules stated in the design phase will be designed and coded by using Microsoft Visual Studio 2008, and Vista TN3270 terminal to link between the database in the mainframe z/OS environment and the Smart Card Reader. Also, the Smart Card Reader will be encoded by using the Smart Card Reader SDK to perform its function to retrieve user's information from a smart card.

3.3.5 Testing

Testing is the fifth phase in System Development Life Cycle (SDLC). Each functions of the proposed system will be tested over and over again to find out any error or bug which may cause the system fail to function perfectly. All the coding and interface will be tested so that the system should function properly. There are many types of testing and the threes are often performed are:

- Unit testing
- System testing
- User acceptance testing

Unit testing is where the source code is tested to determine if they are fit for use. Every part of the system is tested separately to ensure the system can function well.

System testing is conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. System testing is performed on the entire system to ensure that the system can well function.

User acceptance testing generally involves running a suite of tests on the completed system. A particular operating condition of the user's environment or feature

of the system will be exercises, and will result in a pass or fail, or Boolean. The main purpose of user acceptance testing is help to ensure the system is user-friendly.

3.3.6 Implementation

Implementation is the sixth phase in System Development Life Cycle (SDLC). Modular and subsystem programming code should be accomplished in this stage. After completing previous phases, these phases will be integrated to make a complete system. If there were any error or updates, implementation will be done to upgrade the system as well. In the proposed system, many functions like cash transfer service are not included but for the further implementation in the future, these functions can be implemented using all the phase in System Development Life Cycle.

3.3.7 Operation and Maintenance

Every system needs maintenance. Software will definitely face problem or change once it is delivered to customer. Change could happen because of some unexpected input values into the system. In addition, the change could directly affect the system operation and functions. Therefore, maintenance should always be done to the system to make sure it is always functioning well.

3.4 Software and Hardware Requirement

To develop the proposed system, there are certain hardware and software requirement. Software requirement is listed at Table 3.4 and hardware requirement listed at Table 3.5.

3.4.1 Software requirement

Software	**Purpose**
Microsoft Word 2007	Documentation Work
Microsoft Visio 2007	Flow Chart design
Microsoft Project 2007	Gantt Chart design
Vista TN3270	Terminal for Mainframe z/OS
Netbeans 6.9.1	Graphical User Interface Design
Smart Card SDK	Software Development Kit for Smart Card Reader
Visual Paradigm	Data Flow Diagram design

Table 3.4 Software requirement

3.4.2 Hardware requirement

Hardware	**Purpose**
ASUS A43S Series	Develop project and documentation work
Samsung Harddisk 160GB	Project backup storage
Kingston Pendrive 8GB	Data transfer and project backup
Smart Card Reader	Retrieve data of a smart card

Table 3.5 Hardware requirement

CHAPTER 4

IMPLEMENTATION

This chapter focuses on the implementation phase of the system, which consists of implementation of coding, Graphical User Interface and Database.

4.1 Implementing JAVA

For the purpose of coding the system, JAVA programming language is used. The programming involves networking programming, main system programming and graphical user interface programming. Networking programming includes database programming and the smart card reader programming while the main system programming includes the coding for calculations. Graphical user interface programming includes the coding for all the interfaces.

JAVA is chosen because of JAVA has the smartcardio library which can be used to establish a connection between smart card reader with smart cards. Also, as JAVA is an object oriented programming language, functions and main system can be separated in classes and this is easy for the further implementation also.

4.2 Result of Automated Teller Machine using UMP smart card in DB2

The results of the development phase of this system have three categories which are graphical user interface, database and the connection of hardware which is the smart card reader and the smart card. The system output will demonstrate the system functionalities and roles based on the objectives that to be fulfilled. The database will show how the database is used, updated and created.

4.3 Graphical User Interface

In this phase, the graphical user interface design will be explained in detail about the system interface and its function. In the implementation phase, Netbean 6.9.1 has been used as a tool in designing the interface of Automated Teller Machine using UMP Smart Card in DB2. Graphical user interface is important because it represents the system and act as the medium for human and computer interaction, so that the design is made to be as easy understandable, usable and friendly-user to the user.

4.3.1 Main Menu Form

The main menu shows the first interface of the system. The smart card reader needs to be connected. If the system detect there is no present of the smart card reader, it will pop up a dialog box to inform that the smart card reader is not

detected. After that, a smart card need to be put on the smart card reader to be read and to process into the second interface.

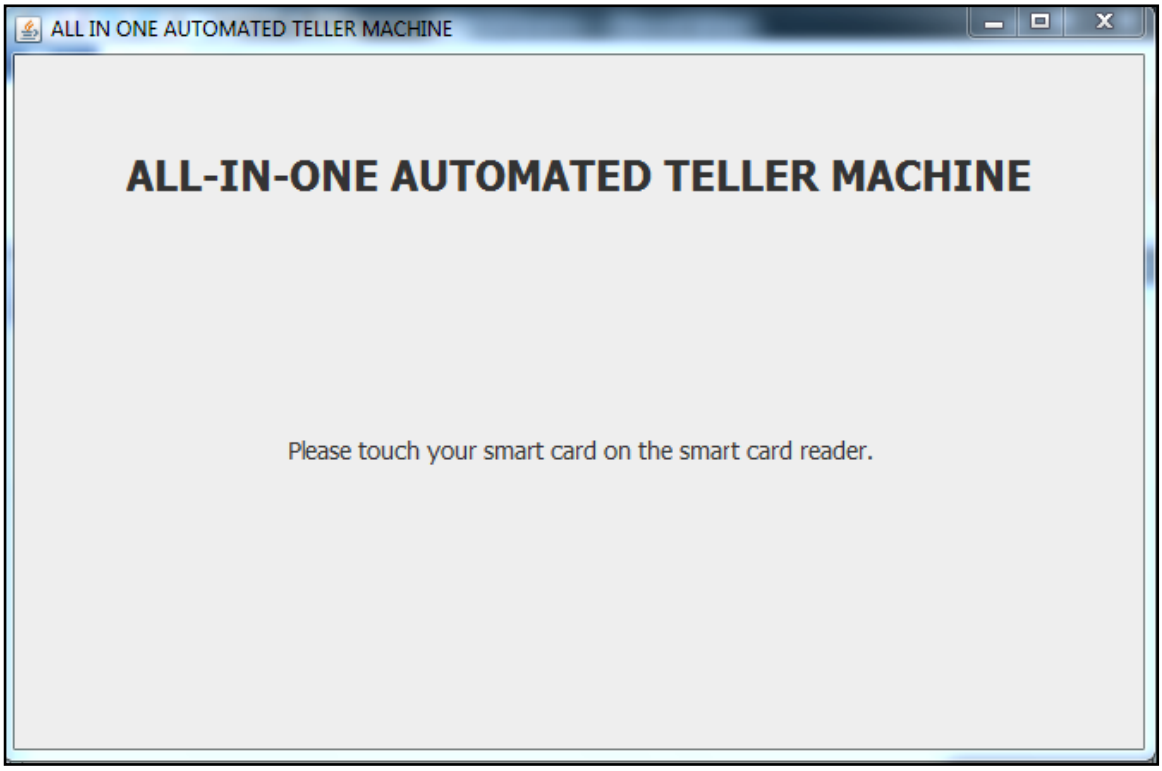


Figure 4.1: Menu Frame

Input	Procedure	Output
Smart Card	User put the smart card on the smart card reader to access the bank choosing interface.	Show bank choosing interface

Table 4.1: Menu Frame Input-Output

4.3.2 Bank Choosing Frame

The Bank Choosing Frame's main function is to show the smart card can access how many banks' automated transaction services. Figure 4.2 show the Bank Choosing Frame which the detected smart card can access both CIMX and BIMX bank's services. There is also a cancel button to let the user to cancel the transactions and take the smart card off from the smart card reader.

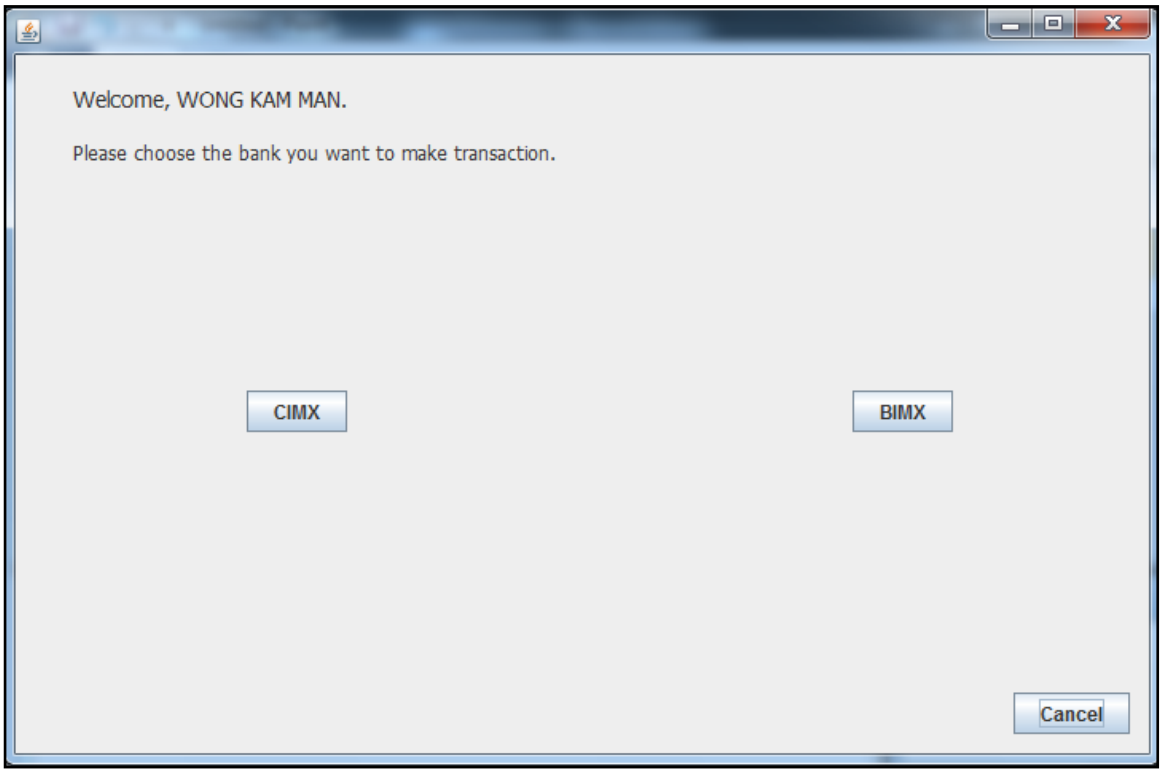


Figure 4.2: Bank Choosing Frame

Input	Procedure	Output
CIMX (button)	User chooses the CIMX bank service	Show Login Frame to let user enter their PIN number
BIMX (button)	User chooses the BIMX	Show Login Frame to let

	bank service	user enter their PIN number
Cancel (button)	User chooses to cancel the service and take the smart card off from the smart card reader.	Show Menu Frame
Welcome (label)	System detects the card holder name from the smart card and display a welcome message.	Card Holder name shown.

Table 4.2: Bank Choosing Input-Output

4.3.3 Login Frame

The Login Frame's main function is to let the user to login to use the automated transaction machine. Account ID is detected from smart card from the menu frame. There is a clear function to let the user to clear the password textbox if they key in the wrong password. Only the correct password can login to use the automated teller machine.

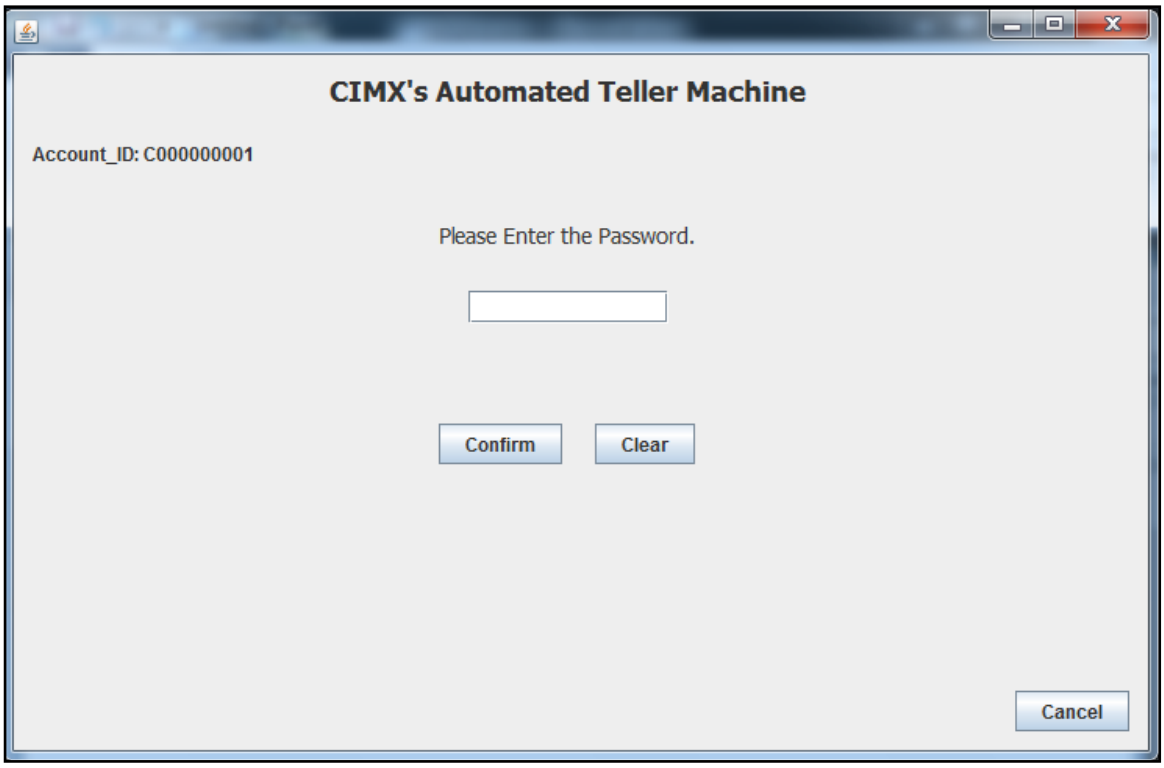


Figure 4.3: Login Frame

Input	Procedure	Output
Password (Textbox)	To let the user to key in their password	none
Confirm (button)	If the user confirm the password, system will login the user to let the user to use the system.	Show Services Frame after success login.
Clear (button)	If the user key in the wrong password, this button let user to clear the wrong password.	Wrong password will be cleared and the new password is waiting to be key in.
Cancel (button)	User chooses to cancel the	Show Menu Frame.

	service and take the smart card off from the smart card reader.	

Table 4.3: Login Input-Output

4.3.4 Services Frame

The Services Frame is the menu to let the user to choose the two functions which are Check Balance inquiry and Withdraw Balance. Check Balance Inquiry functions is to let the user to check their current account's balance and Withdraw Balance is the functions to let the user to withdraw the cash balance from the automated transaction machine.

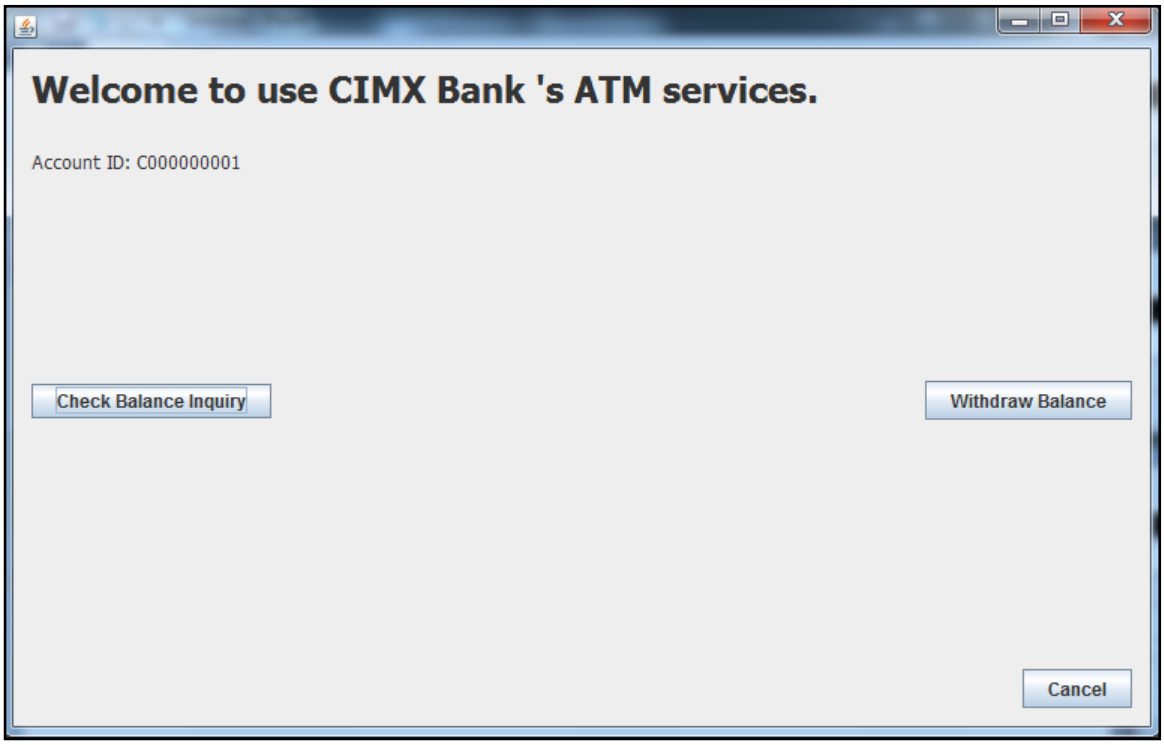


Figure 4.4: Services Frame

Input	Procedure	Output
Check Balance Inquiry (button)	User click the button to procedure to the Check Balance Inquiry Frame	Show Check Balance Inquiry Frame
Withdraw Balance (button)	User click the button to procedure to the Withdraw Balance Frame	Show Withdraw Balance Frame
Cancel (button)	User chooses to cancel the service and take the smart card off from the smart card reader.	Show the Menu Frame

Table 4.4: Services Input-Output

4.3.5 Check Balance Inquiry Frame

The Check Balance Inquiry Frame is to let the user to check their current account's balance inquiry. There are two buttons in the Check Balance Inquiry Frame and one label that show the current balance.

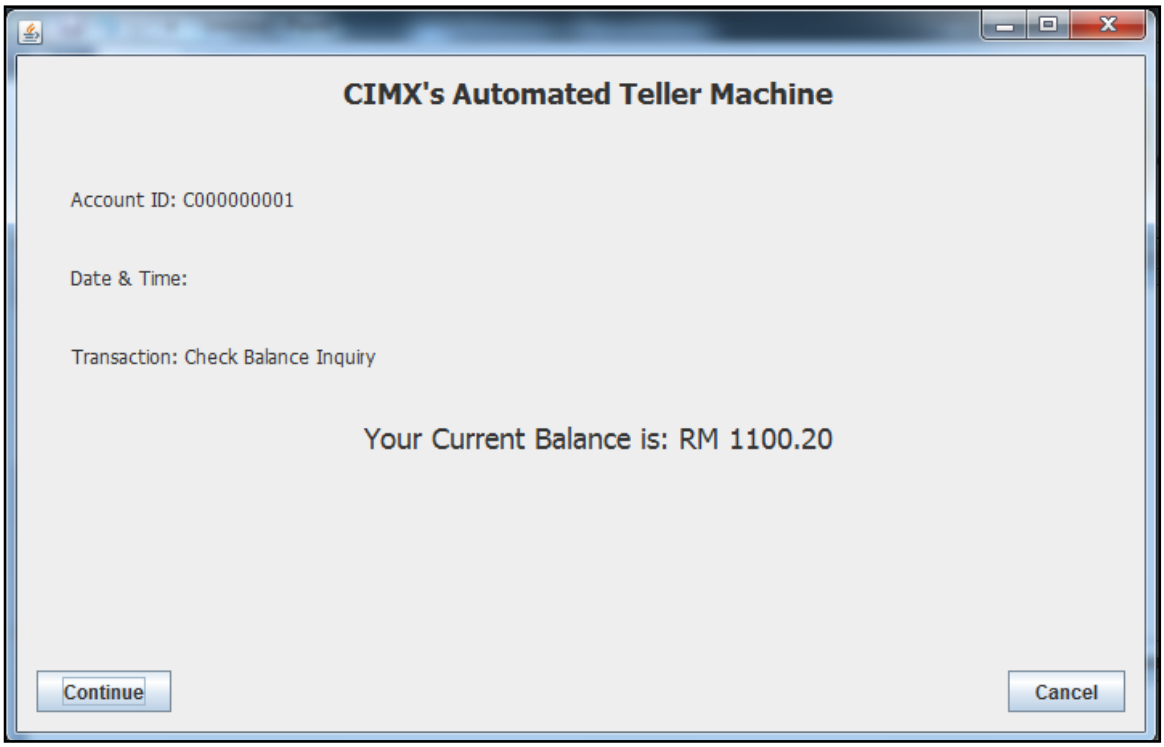


Figure 4.5: Check Balance Inquiry Frame

Input	Procedure	Output
Current Balance (label)	Check current balance from the database and show it to the user.	Show the current balance to the user in the label form.
Continue (button)	To let the user to continue to the Withdraw Balance service.	Show the Withdraw Balance Frame.
Cancel (button)	User chooses to cancel the service and take the smart card off from the smart card reader.	Show the Menu Frame

Table 4.5: Check Balance Inquiry Input-Output

4.3.6 Withdraw Balance Frame

The Withdraw Balance Frame is to let the user to withdraw balance from the automated transaction machine. In this frame, there are six buttons and one radio button with a text box. The buttons is to choose the common withdraw amount and radio button is to let the user to fill in the amount he wants to withdraw. The label will show the user his current balance and also the withdrawal limit he can make.

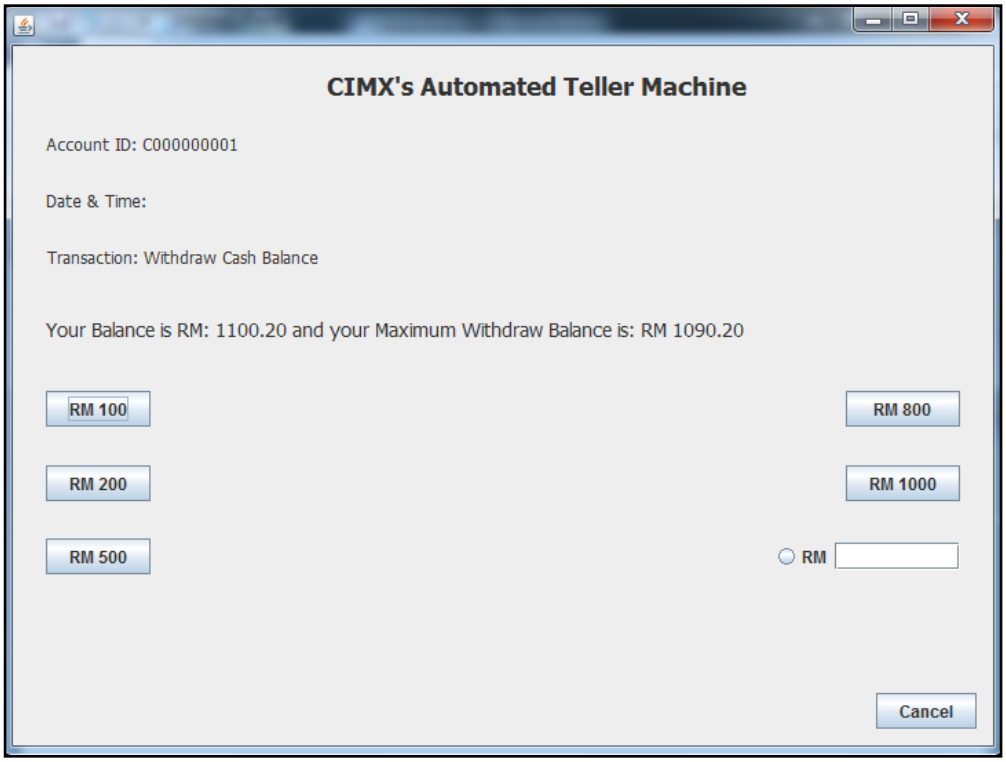


Figure 4.6: Withdraw Balance Frame

Input	Procedure	Output
Current and Withdrawal Limit (label)	System checks the Balance and the Withdrawal limit from the database and show to the user.	Show the Balance and the Withdrawal limit to the user.
RM 100 – 1000 (buttons)	To let user to choose the	Balance will be deducted

	withdrawal balance that commonly used by everyone.	and the Updated Balance Frame will be shown
RM with Textbox (Radio button)	To let user to key in the withdrawal balance that the user want to	Balance will be deducted and the Updated Balance Frame will be shown

Table 4.6: Withdrawal Balance Input-Output

4.3.7 Updated Balance Frame

The Updated Balance Frame will show the Balance before withdraw, withdraw balance and Balance after withdraw. There are two buttons which are Continue and Cancel is to let the user choose to continue to use the automated transaction machine or not to use it anymore.

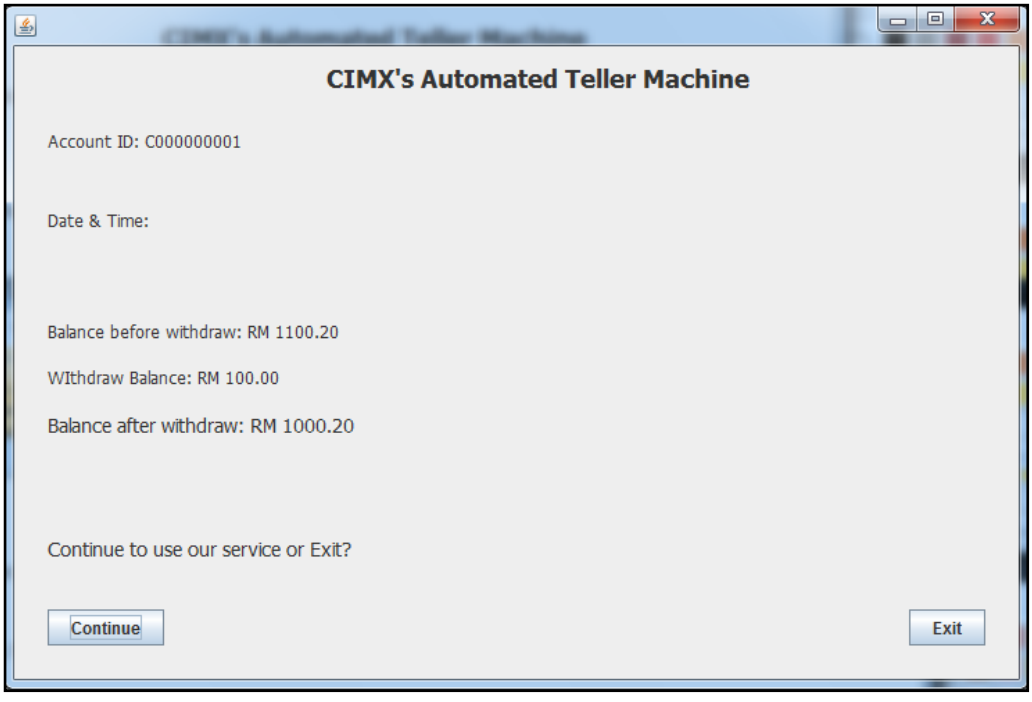


Figure 4.7: Updated Balance Frame

Input	Procedure	Output
Balance before withdraw (label)	Check the Balance before withdraw	Show the balance before withdraw in label
Withdraw Balance (label)	Check the withdraw balance that user make	Show the withdraw balance in label
Balance after withdraw (label)	Check the updated balance after the withdrawal	Show the updated balance in label
Continue (button)	To let the user choose to continue the service and procedure to the menu frame.	Show the menu Frame
Exit (button)	To let the user exit from the system and take the smart card off from the smart card reader	Exit from the system

Table 4.7: Updated Balance Input-Output

4.4 Coding Development

At coding development, it describe how the data being read from the smart card via the smart card reader, show current balance inquiry and also deduct the withdraw balance from the previous balance to get the Updated balance.

4.4.1 Read data from smart card

To read the data from the smart card, firstly we need to import smartcardio library. Then we need to make the system to get a list that all the smart card reader that are connected and use the detected smart card reader. Then we need to program the smart card reader to send a APDU to the smart card to let the smart card to send back a response APDU which contains the card id. In Figure 4.8 is the coding that to make the smart carder to perform its functions.

```
try {

    List<CardTerminal> terminals = factory.terminals().list();

    //System.out.println("Terminals: " + terminals);
    /*
    if (terminals.size() == 0) {
        //System.err.println("No terminals Found.");
        JOptionPane.showMessageDialog(null, "No Terminals Found.", "No Device Warning", 0);
        return;
    }
    */
    // Get the first terminal in the list
    CardTerminal terminal = terminals.get(0);
    if (terminal.isCardPresent()) {
    // Establish a connection with the card using
    // "T=0", "T=1", "T=CL" or "*"
    Card card = terminal.connect("*");
    //System.out.println("Card: " + card);
    CardChannel channel = card.getBasicChannel();

    byte[] baCommandAPDU = { (byte) 0xFF, (byte) 0xCA, (byte) 0x00, (byte) 0x00, (byte) 0x04};
    //System.out.printf("APDU >>>: " + HexToString.toString(baCommandAPDU));
    //System.out.println();
    ResponseAPDU r = channel.transmit(new CommandAPDU(baCommandAPDU));
    String rResponseAPDU = HexToString.toString(r.getBytes());
```

Figure 4.8: Coding for Smart Card Reader to read data from smart card

4.4.2 Convert Hexadecimal byte to Hexadecimal String

This function convert all the Hexadecimal byte APDU to a Hexadecimal String to make sure the card id can be read from the smart card correctly. With Hexadecimal byte, the System cannot show out the correct card id because print function cannot support Hexadecimal byte but Hexadecimal String.

```
import java.util.*;
import javax.smartcardio.*;

public class HexToString {

    //convert APDU to hexString
    public static String toString(byte[] b) {
        StringBuffer sb = new StringBuffer(b.length*2);
        for (int i = 0; i < b.length; i++) {
            int v = b[i] & 0xff;
            if (v < 16) {
                sb.append('0');
            }
            sb.append(Integer.toHexString(v));
        }
        return sb.toString().toUpperCase();
    }
```

Figure 4.9: Coding for Converting Hexadecimal byte to Hexadecimal String

4.4.3 Select data from database

This function is to execute query to the select the data from the database based on the data read from the smart card. The query is generated by the buttons and events or the action performed.

```
public void selectSQL(String query) {

    try {

        Class.forName ("com.mysql.jdbc.Driver");

        conn = DriverManager.getConnection (url+dbName,userName,password);

        stmt = conn.createStatement();
        ResultSet rs = stmt.executeQuery(query);

        while (rs.next()) {
            Balance = rs.getFloat(1);
            jLabel4.setText("Your Balance is RM: " + df.format(Balance)+ " and your Maximum Withdraw Balance is: RM " +
        }

        stmt.close();
        conn.close();

    } catch (Exception ex) {

    }
}
```

Figure 4.10: Coding for Select data from database

4.4.4 Update the Updated balance to database

This function execute the update query that generated after the withdraw balance is chosen and the balance is updated.

```
public void updateSQL (String query) {

    try {

        Class.forName ("com.mysql.jdbc.Driver");
        conn = DriverManager.getConnection (url+dbName,userName,password);

        stmt = conn.createStatement();
        stmt.executeUpdate(query);

        stmt.close();
        conn.close();
    }
    catch (Exception ex) {

    }
}
```

Figure 4.11: Coding for update the Updated Balance

4.4.5 Withdraw Balance and generate update query

This function is performed when one of the withdraw balance buttons is click and a updated balance query will be generate and send to updateSQL function to update the Updated Balance into database.

```
private void jButton2MouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    WithBalance= 100;
    if(WithBalance <= (Balance-10)) {
        FinalBalance=(Balance)-WithBalance;
        String Withquery = "UPDATE "+Table_Name+" SET Balance = "+FinalBalance+" where Account_ID= '"+Account_ID+"' ";
        updateSQL(Withquery);
        String Selectquery = "Select Balance FROM "+Table_Name+" where Account_ID= '"+Account_ID+"' ";
        BalanceSQL(Selectquery);

    }
    else {
        JOptionPane.showMessageDialog(null, "Your Withdraw Balance is over limit.", "Invalid Transaction", 1);
    }
}
```

Figure 4.12: Coding for Withdraw Balance and generate update query

4.4.6 Determine which bank can be accessed by the smart card

This function is used to determine which smart card can access which bank. The data read from the smart card will be compared with the data in the database and the button will be visible if the data are matched.

```
try {

    String userName = "root";
    String password = "081089";
    String url = "jdbc:mysql://localhost:3306/";
    String dbName = "atm";
    String query = "Select C_Account_ID, B_Account_ID FROM card_holder where card_id = '"+Card_id+"' ";

    Class.forName ("com.mysql.jdbc.Driver");

    conn = DriverManager.getConnection (url+dbName,userName,password);

    Statement stmt = conn.createStatement();
    ResultSet rs = stmt.executeQuery(query);

    while (rs.next()) {
    cimx = rs.getString(1);
    bimx = rs.getString(2);
        if (cimx != null) {
            jButton1.setVisible(true);
            jButton1.setText("CIMX");
        }
        if (bimx != null) {
            jButton2.setVisible(true);
            jButton2.setText("BIMX");
        }
    }
```

Figure 4.13: Coding for determine bank that can be accessed by the smart cards.

4.4.7 Cancel button to cancel the transaction and back to the main menu frame

This function is to cancel the transaction and back to the main menu frame once the user click the cancel button and remove the smart card from the smart card reader. If the smart card doesn't being removed, user will be notified by a pop up dialog box to remind the user to take the smart card from the smart card reader.

```
private void jButton3MouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    TerminalFactory factory = TerminalFactory.getDefault();

    try {

        List<CardTerminal> terminals = factory.terminals().list();
        CardTerminal terminal = terminals.get(0);
        if (terminal.isCardPresent()) {
            JOptionPane.showMessageDialog(null, "Please remember to remove your card.", "Remove Card Warning", 1);
        }
        else {
            dispose();
            new ATM_1st_TouchCard();
        }
    }
    catch (Exception e) {

    }
}
```

Figure 4.14: Coding for cancel the transaction and back to the menu frame

4.4.8 Enter Password and Login to use the automated teller machine system

This function performs the login function to let the user to login and use the automated teller machine system after scanning the smart card and choosing the bank. If the correct password is entered, the system will process to the next interface, else it will pop up a dialog box to inform the user that the password is not correct.

```
try {

   String userName = "root";
   String password = "081089";
   String url = "jdbc:mysql://localhost:3306/";
   String dbName = "atm";
   String query = "Select Password FROM "+Table_Name+" where Account_ID= '"+Account_ID+"' ";
   Class.forName ("com.mysql.jdbc.Driver");

   conn = DriverManager.getConnection (url+dbName,userName,password);

   Statement stmt = conn.createStatement();
   ResultSet rs = stmt.executeQuery(query);

   while (rs.next()) {
       Password = rs.getString(1);
       if(passwordfield.equals(Password)) {
           dispose();
           new ATM_4th_ChooseService(Table_Name, Account_ID);
           }
       else {
           JOptionPane.showMessageDialog(null, "Invalid Password.", "Password Error", 1);
           jPasswordField1.setText("");
           jPasswordField1.requestFocusInWindow();
       }
   }
```

Figure 4.15: Coding for enter password and login to use the system

4.5 Interacting with Database

The card id is retrieved via the smart card reader, after that the card id will be used to interact with the database. The database have 3 tables. The tables are Card_Holder, CIMX and BIMX. Card_Holder Table store the Card ID, and the Account_ID for both CIMX and BIMX bank. CIMX and BIMX Table both store the same information which are Account ID, Password, Holder_Name, Balance, and Last Withdrawal Date.

When the card id is retrieved from the smart card reader, it will be compared to the card id that stored in the Card_Holder table. Once matches, it will indicate that the card id is valid for the system use. With those functions mentioned in previous part, such as determine which bank are accessible, withdraw balance and update, select current balance and login to the automated teller machine system, data and

information is updated, selected and stored into the database when those functions is performed.

UID	Account_ID	Password	Holder_Name	Balance	LAST_WITH_DATE
1	C000000001	123456	WONG KAM MAN	1000.19995	2012-05-21 08:57:39
NULL	NULL	NULL	NULL	NULL	NULL

Figure 4.16: Sample data for CIMX table

UID	Account_ID	Password	Holder_Name	Balance	LAST_WITH_DATE
1	B000000001	123456	WONG KAM MAN	3000.5	2012-05-21 04:27:37
NULL	NULL	NULL	NULL	NULL	NULL

Figure 4.17: Sample data for BIMX table

UID	Card_id	Holder_Name	C_Account_ID	B_Account_ID
1	93ACDC979000	WONG KAM MAN	C000000001	NULL
2	D370DB979000	WONG KAM MAN	C000000001	B000000001
3	E31DDF979000	WONG KAM MAN	NULL	B000000001
NULL	NULL	NULL	NULL	NULL

Figure 4.18: Sample data for Card_Holder table

CHAPTER 5

RESULT & DISCUSSION

This chapter focuses on the result and data analysis that had been acquired in previous chapter. The output of the system will be discussed too in this chapter. Besides that, the details which are including the outcome of the system and future research or work for the limitation about this system will also be discussed.

5.1 Outcome of the Project

In this project, users are able to use their smart card to check their current balances and withdraw the balance from the bank that they want to. User will touch the smart card on the smart card reader which shown in Figure 5.1. After that the system will get the data from the smart card and let the user to access to his available bank accounts which shown in Figure 5.2. . Figure 5.3 shows that after choosing bank, user needs to type in his password which is the PIN number for the bank account to get the authorized and access right to his bank account. Figure 5.4 show that user can choose either to check balance inquiry or to withdraw the balance. Figure 5.5 shows the current balance inquiry and Figure 5.6 shows the user is

withdrawing balance. Finally the system will show the updated balance and the withdraw amount which is in Figure 5.7.

Figure 5.1: User needs to touch the smart card on the smart card reader.

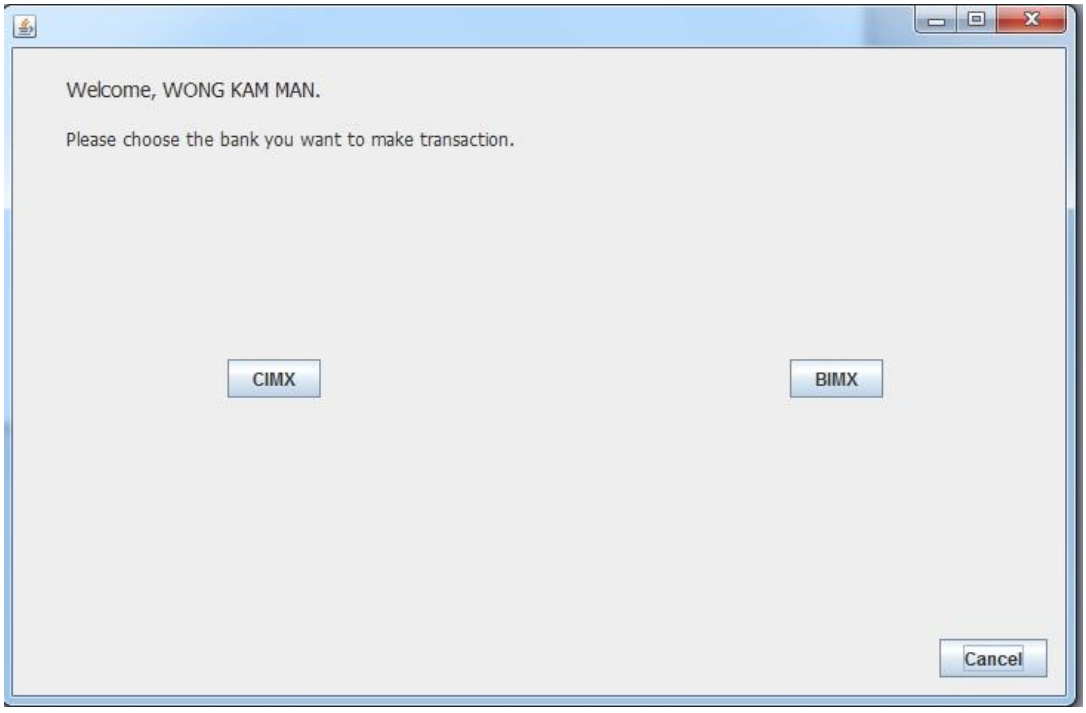


Figure 5.2: The available banks that user can use to do transactions.

Figure 5.3: User type in the password to gain access to the account.

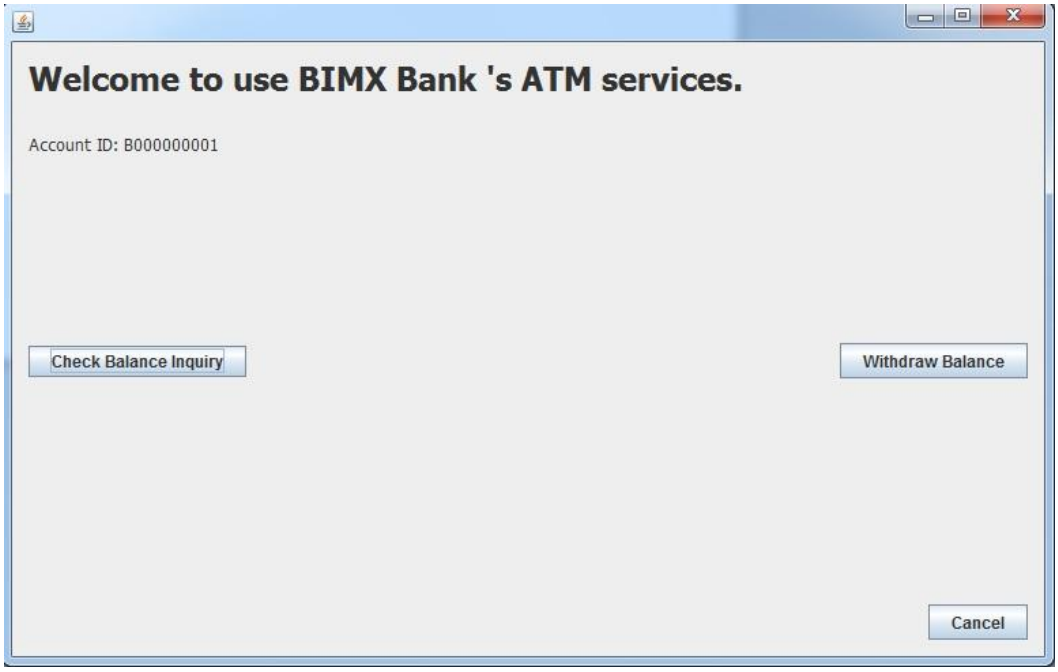


Figure 5.4: User can choose either to check balance inquiry or Withdraw Balance.

Figure 5.5: User can check the balance inquiry.

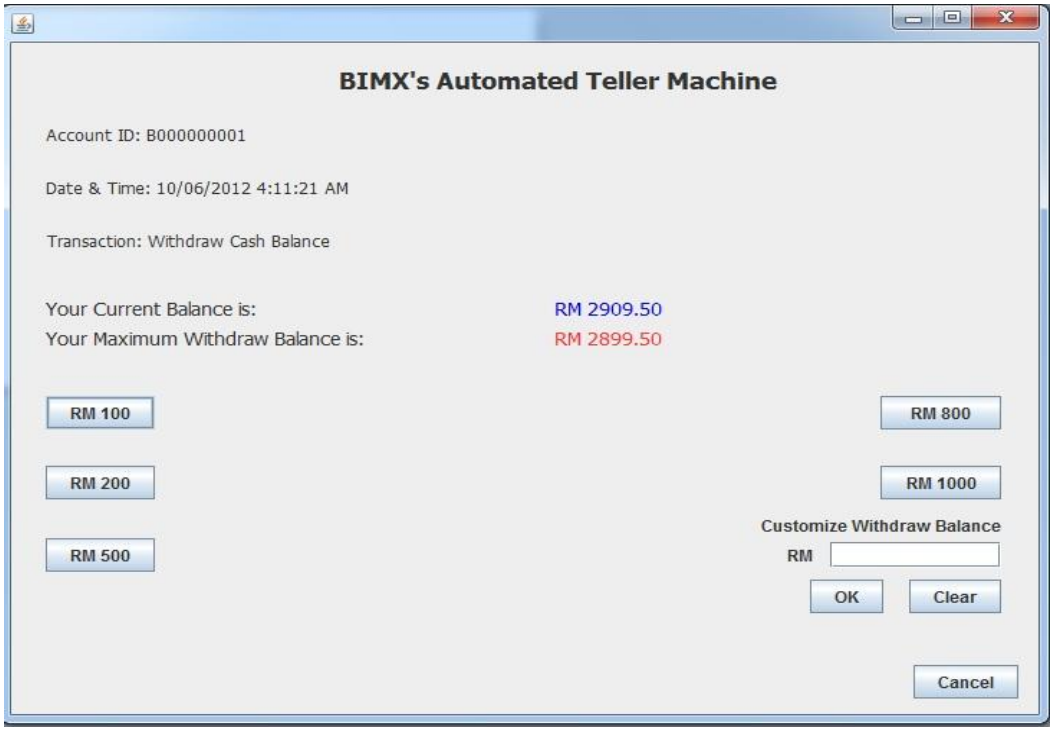


Figure 5.6: User is withdrawing the balance.

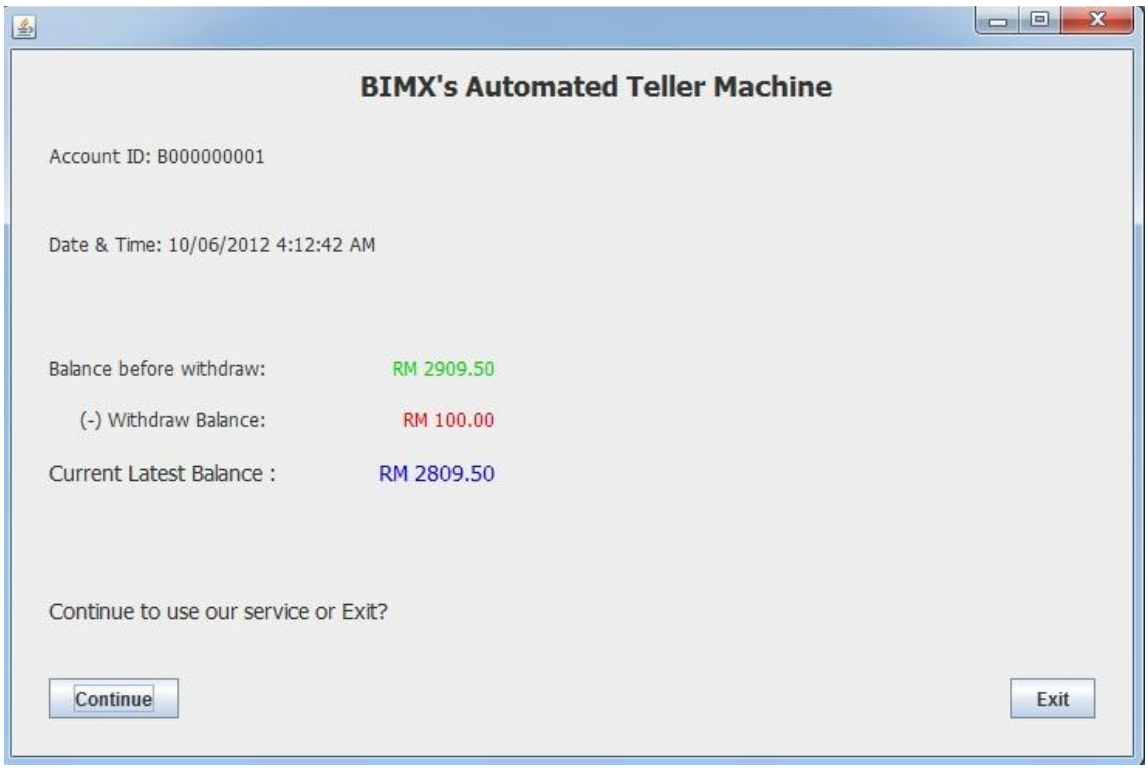


Figure 5.7: User can view the current balance and the amount he withdraw.

5.2 Discussion

This project is built in Netbeans 6.9.1 software using JAVA programming. To connect to DB2 in z/OS by using JAVA, JDBC driver need to be downloaded and installed. After installing the JDBC driver, there is a problem occurs where the DB2 in z/OS cannot be connected due to the access and authorized right. Because it is one of a high risk security issues, connecting to DB2 in z/OS is not possible. Therefore, DB2 in Windows is used to stimulate the DB2 in z.OS.

5.2.1 Limitations of the Project

Because the system is proposed just for testing purpose, this project is still having some limitations although it is successfully implemented. The limitations are stated as below:

i. The system is demonstrated with check balance inquiry and cash withdrawal functions only.
ii. The system cannot withdraw the real cash out.
iii. The data in database is dummy data only due to the confidentially of the real data.

5.3 Future Work

This system is just in an initial stage and there should be further improvement in the future since there is still having some limitations. This system could be

enhanced so that the system is really connected to the DB2 in z/OS. Also, it can be enhanced with other functions such as balance transfer, prepaid-reload and so on. Furthermore, the system should be implemented into a real Automated Teller Machine which can withdraw real cash.

CHAPTER 6

CONCLUSION

In conclusion, this project with the title of Automated Teller Machine using UMP smart card in DB2 has successfully built and achieved the objectives that have been stated on the Chapter 1. The user is able to use an all-in-one ATM card to withdraw balance and check balance inquiry. The system combines more than one banks ATM services to let the user to check their balance inquiry also withdraw cash. Therefore, users do not need to bring many cards with them and this will bring convenient to users. In addition, this system has the potential to become an effective Automated Teller Machine after further implementation by adding more functions such as cash transfer and prepaid reload in the future.

REFERENCES

A.Rahman, S. Azad, F.Anwar, R. Islam. (2008). *Design of smart card for automated toll collection at Jamuna Multpurpose Bridge in Bangladesh.* Paper presented at International Conference on Electrical and Computer Engineering, 2008.

Attoh-Okine N. O. and David Shen L. (1995). *Security Issues of Emerging Smart Cards Fare Collection Application in Mass Transit.* (95, 523-526) Retrieved from IEEE Xplore Digital Library database.

Bernardo Batiz-Lazo and Robert J.K. Reid. (2011). IEEE Annals of the History of Computing [Electronic Version]. *The Development of Cash-Dispensing Technology in the UK.*

Cooke J.C. and Brewster R.L. (1993). *The Use of Smart Card in Personal Communication System Security*. April 18-21, 1993. IEEE, 1993. 246-251.

Coventry L., Angeli A. D. and Johnson G. (2003). *Usability and biometric verification at the ATM interface.* Paper published in Proceedings of the SIGCHI conference on Human factors in computing systems.

Ebbers M., Kettner J., O'Brien W. and Ogden B. (2009). Redbooks. *Introduction to the New Mainframe: z/OS Basics.* (2nd ed.) IBM Corp.

Katherine M. S. and J.Drew Procaccino. (2002). Communication of the ACM. *Smart Card Evolution* (vol. 45, no.7, 83-88).

Mohammed A. M. Abdullah, F. H. A. Al-Dulaimi, Waleed Al-Nuaimy, Ali Al-Ataby. (2011). *Smart Card with iris recognition for high security access environment.* Paper presented at 1st Middle East Conference on Biomedical Engineering (MECBME), 2011.

Moncur W. and Leplâtre G. (2007). *Pictures at the ATM: exploring the usability of multiple graphical passwords.* Paper presented at Proceedings of the SIGCHI conference on Human factors in computing systems, 2007.

Munjal N. and Moona R. (2009). *Secure and Cost Effective Transaction Model for Financial Service.* October 12-14, 2009. St.Petersburg: IEEE, 2009. 1-6.

Roper M. and Millar L. (1999). *Managing Public Sector Records: A Study Programme – Managing Hospital*

Records.(1999). International Records Management Trust.

Schaefer R., Mueller W., López A.M. and Sánchez D. D. (2007). *Using smart cards for secure and device independent user interfaces.* Paper published in Proceeding of 4th international conference on mobile technology, applications, and systems and the 1st international symposium on Computer human interaction in mobile technology, 2007.

Shekhar, R. (2010-2011). Oracle and MySQL Comparison. *DBA Metrix Solutions*.

Retrieved 13 October, 2011 from

http://www.dbametrix.com/comp-oracle-mysql.html

Soon Kim C. and Mun-Kyu L. (2010). *Secure and user friendly PIN entry method.* Paper presented at Digest of Technical Papers International Conference on Consumer Electronics (ICCE), 2010.

Trivedi, G. (2010). IBM DB2 and Oracle Server Comparison and differences. *DBAMetrix Solutions*. Retrieved 13 October 2011 from http://www.dbametrix.com/db2-oracle-server.html

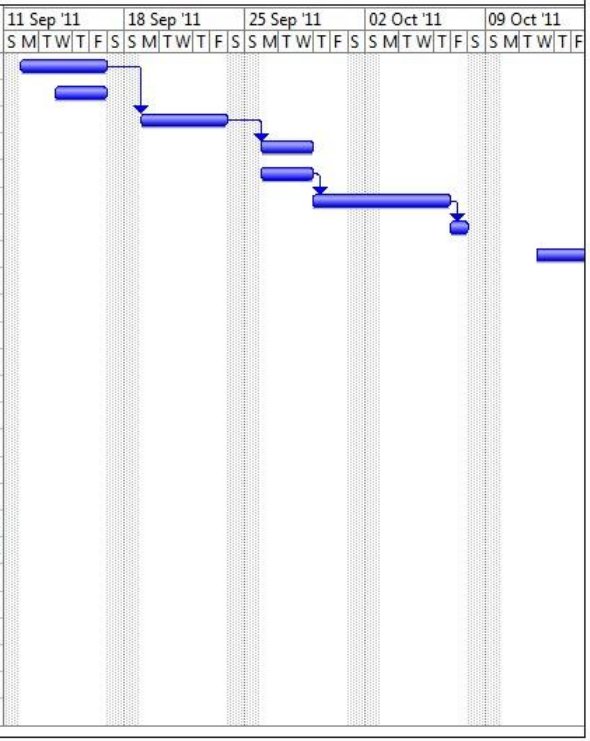
Vigil, J. POS 355. (2002, April 10). *Compare and Contrast Mainframes and PersonalComputers*. Retrieved 15 October 2011 from http://www.hippieloco.com/School/pos355/mainfram%20and%20pc.htm

APPENDIX A

GANTT CHART

Grantt Chart PSM

ID	Task Name	Duration	Start	Finish
1	Briefing of Introduction, Identify objectives and	5 days?	Mon 12-09-11	Fri 16-09-11
2	Plan schedule, make grantt chart	3 days?	Wed 14-09-11	Fri 16-09-11
3	Doing Chapter 1 Introduction	5 days?	Mon 19-09-11	Fri 23-09-11
4	Submit Chapter 1 and Briefing of Literature Revi	3 days?	Mon 26-09-11	Wed 28-09-11
5	Find Article and Journals	3 days?	Mon 26-09-11	Wed 28-09-11
6	Review Related Literature	6 days?	Thu 29-09-11	Thu 06-10-11
7	Submit Chapter 2, Discussion of device.	1 day?	Fri 07-10-11	Fri 07-10-11
8	Briefing of Methodology, discussion about arch	4 days?	Wed 12-10-11	Mon 17-10-11
9	Discussion about Methodology, Database desig	22 days?	Tue 18-10-11	Wed 16-11-11
10	Discussion about Methodology, user interface c	18 days?	Thu 17-11-11	Mon 12-12-11
11	Submit draft of Methodology, Result and Discu	4 days?	Tue 13-12-11	Fri 16-12-11
12	Submit Methodology, showing draft of final rep	6 days?	Mon 19-12-11	Mon 26-12-11
13	Submit draft of thesis and report PSM1, Overvie	3 days?	Wed 28-12-11	Fri 30-12-11
14	Preparation for presentation	2 days?	Fri 13-01-12	Mon 16-01-12
15	Discussion with Supervisor	1 day?	Tue 17-01-12	Tue 17-01-12
16	Presentation of PSM1	3 days?	Wed 18-01-12	Fri 20-01-12
17	Preparation for PSM2	2 days?	Mon 30-01-12	Tue 31-01-12
18	Coding	57 days	Wed 01-02-12	Thu 19-04-12
19	Testing and Repair Code	15 days	Fri 20-04-12	Thu 10-05-12
20	Implementing	7 days	Fri 11-05-12	Sat 19-05-12
21	Submit Chapter 4 & 5	1 day	Mon 21-05-12	Mon 21-05-12
22	Implementing and demo system	7 days	Tue 22-05-12	Wed 30-05-12
23	Complete and submit the thesis	7 days	Fri 01-06-12	Mon 11-06-12
24	Prepare for of PSM2 Presentation	4 days	Tue 12-06-12	Fri 15-06-12
25	Presentation of PSM2	2 days	Mon 18-06-12	Tue 19-06-12



APPENDIX B

USER MANUAL

SYSTEM: AUTOMATED TELLER MACHINE USING UMP SMART CARD IN DB2

1. **Touch the Smart Card**

Figure B1

i. Touch the smart card on the smart card reader.

ii. A Pop Up message box will be shown as Figure B1 if smart card cannot be detected.

2. Choose the available banks

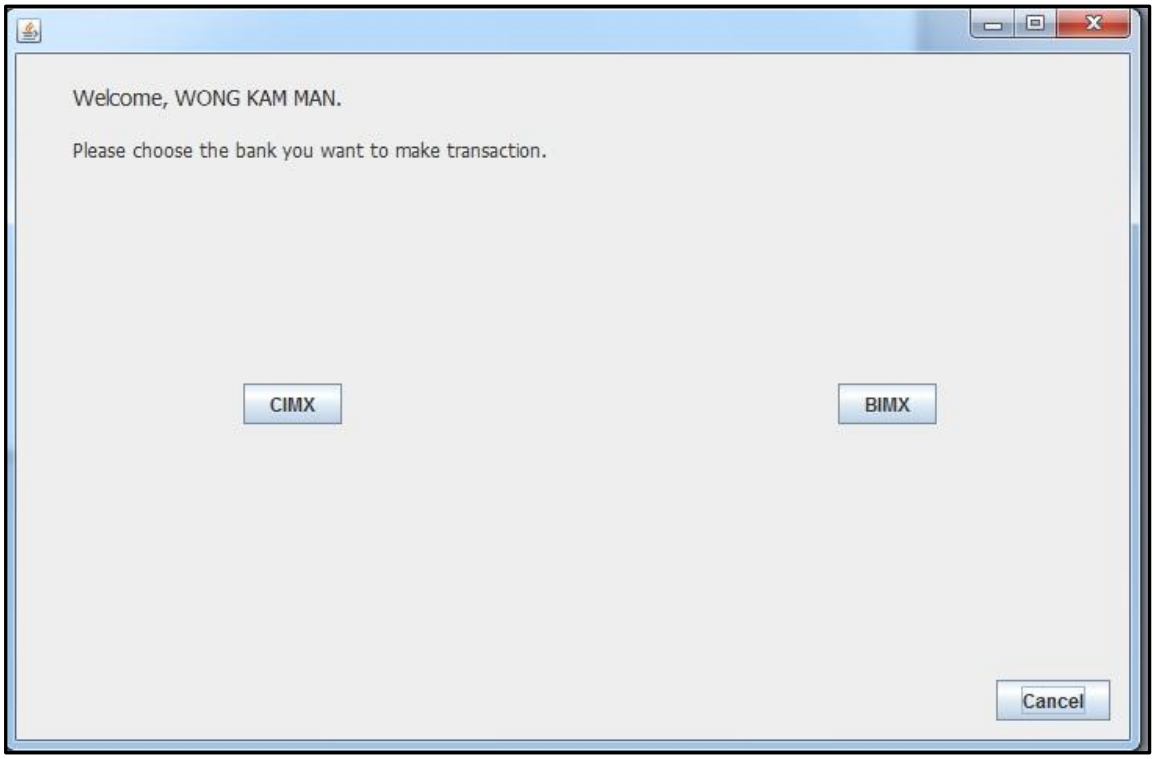


Figure B2

i. After smart card is detected, available banks will be shown as Figure B2.
ii. User can choose either CIMX or BIMX if the card can access multiple banks.

3. Enter the PIN Number

Figure B3

i. After choosing one of the bank, user must enter the correct PIN number to access to the ATM services provided by the bank as shown in Figure B3.
ii. Clear button is to let the user to clear the PIN Number that entered wrongly.

4. Choose the Services

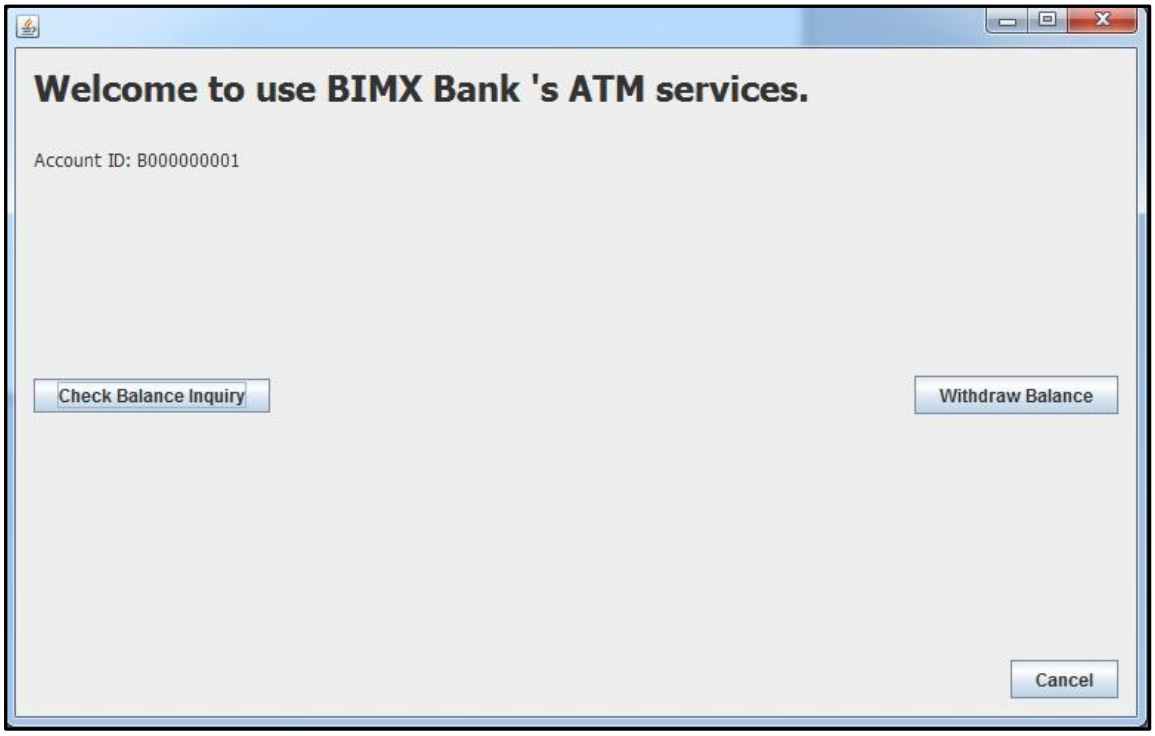


Figure B4

i. After entered the correct PIN number, user can choose either to check balance inquiry or to withdraw balance.

5. Check Balance inquiry

Figure B5

i. To check balance inquiry, user needs to choose check balance inquiry in step 4.
ii. Current Balance will be shown as in Figure B5
iii. To continue the service, press the continue button.
iv. To terminate the service, take the card off from the smart card reader and press the cancel button.

6. Withdraw Balance

Figure B6

i. To withdraw balance, user needs to choose withdraw balance in step 4.
ii. Current Balance and the maximum withdraw balance amount will be shown as in Figure B6.
iii. User can click on the button which are originally set with the withdraw amount.
iv. If user wants to withdraw amount that not originally set, user can fill in the amount at the Customize Withdraw Balance text box.
v. After fill in the Customize Withdraw Balance, user should press OK button to confirm the withdraw amount, cents are not allowed and will be ignored.
vi. To terminate the service, take the card off from the smart card reader and press the cancel button.

7. View Updated Current Balance

Figure B7

i. Balance before withdraw, Withdraw Balance and Latest Current Balance will be shown as in Figure B7.

ii. To continue use the services, press the continue button and user will back to the Choose services interface in step 4.

iii. To terminate the service, take the card off from the smart card reader and press the cancel button.