DEVELOPMENT OF SOFTWARE PART OF A SMART ATTENDANCE SYSTEM

NUR ADILAH BINTI KAMARUDIN

This report is submitted in partial fulfillment of the requirement for the award of
Bachelor of Electronic Engineering (Computer Engineering) With Honours

Faculty of Electronic and Computer Engineering
Universiti Teknikal Malaysia Melaka

MAY 2008

© Universiti Teknikal Malaysia Melaka

UNIVERSTI TEKNIKAL MALAYSIA MELAKA
FAKULTI KEJURUTERAAN ELEKTRONIK DAN KEJURUTERAAN KOMPUTER

BORANG PENGESAHAN STATUS LAPORAN
PROJEK SARJANA MUDA II

Tajuk Projek	:	DEVELOPMENT OF SOFTWARE PART OF A SMART ATTENDANCE SYSEM
Sesi Pengajian	:	2007/2008

Saya .....NUR ADILAH BINTI KAMARUDIN.....
(HURUF BESAR)
mengaku membenarkan Laporan Projek Sarjana Muda ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

1. Laporan adalah hakmilik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan laporan ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. Sila tandakan (√) :

☐	**SULIT***	(Mengandungi maklumat yang berdarjah 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**	

Disahkan oleh:

(TANDATANGAN PENULIS)

Alamat Tetap: 48, JLN AU5D/3,
LEMBAH KERAMAT
54200 KUALA LUMPUR

Tarikh: 9 MEI 2008

(COP DAN TANDATANGAN PENYELIA)

REDZUAN B ABDUL MANAP
Ketua Jabatan (Kej Telekomunikasi)
Fakulti Kej Elektronik dan Kej Komputer (FKEKK)
Universiti Teknikal Malaysia Melaka (UTeM)
Karung Berkunci 1200,
Ayer Keroh, 75450 Melaka

Tarikh: 9/5/08

© Universiti Teknikal Malaysia Melaka

DECLARATION

"I hereby declare that this thesis entitle DEVELOPMENT OF SOFTWAR PART OF A SMART ATTENDANCE SYSTEM is result of my own effort except for works that have been cited clearly in the references."

Signature : .....

Name : NUR ADILAH BT KAMARUDIN

Date : 9 MAY 2008

© Universiti Teknikal Malaysia Melaka

"I hereby declare that I have read this report and in my opinion this report is sufficient in terms of the scope and quality for the award of Bachelor of Electronic Engineering (Computer Engineering) With Honours."

Signature : .....

Supervisor's Name : EN. REDZUAN BIN ABDUL MANAP

Date : 9 MAY 2008

© Universiti Teknikal Malaysia Melaka

Dedicated to my lovely family, my beloved one and my dear friends

© Universiti Teknikal Malaysia Melaka

ACKNOWLEDGMENT

"In the name of God, the most gracious, the most compassionate"

I would like to thank and express my appreciation for the support that I received throughout my studies from my supervisor, Encik Redzuan bin Abdul Manap. His kindness and encouragement made me always confident and enlighten my curiosity during completion of the project. Without his continued support and interest, this thesis would not have been the same as presented here.

My sincere appreciation also extends to all my friends and colleagues have contributed to my studies in many ways. The exceptional help, support and friendship that I received from Siti Aisyah, Nazihan, Nur Amalina, Kamaliah Hanim, Mohd Khairul Amin and others who are not mentioned here. Their views and tips are useful indeed. I am grateful to all my family members for their supports.

"May Allah bless us with His Taufik and Hidayat. May we benefit from the knowledge He has given us. May we always be under His Protection and Guidence. May He forgive us for our sins, those we know and those we do not know. May He place us on the righteous path and steadfast our Imans. May He shower our one and true Prophet Muhammad Alaihisalam and hisfamil and followers, with eternal blessings. Amin amin, ya rabbal-alamin"

© Universiti Teknikal Malaysia Melaka

ABSTRACT

The software section of a smart attendance system is developed in this project used to monitor the student attendance complete with student information such as name, matrix number, date etc. The project work consists of the development of student's information database. Once the software part is done, this project will combine with the hardware part which is done by another student. This system software developed with extended Graphic User Interface by using Microsoft Visual Basic.Net and is integrate with Microsoft Access 2007 to build the database.

© Universiti Teknikal Malaysia Melaka

ABSTRAK

Perisian sistem kehadiran pintar telah dibangunkan dalam projek (*smart attendance system)*. Ia digunakan untuk memantau kehadiran pelajar dengan maklumat pelajar seperti nama, nombor matriks, tarikh dan sebagainya. Projek ini juga mengandungi pembangunan maklumat pelajar ke dalam pengkalan data. Selepas bahagian perisian telah lengkap siap, projek ini akan digabungkan dengan bahagian perkakasan yang dibuat oleh pelajar lain. Projek ini adalah untuk menewaskan sistem kehadiran lama. Sistem ini akan dimajukan dengan GUI yang dilanjutkan dengan menggunakan Microsoft Visual Basic.Net dan disepadukan dengan Microsoft Access 2007 untuk pembinaa pangkalan data.

© Universiti Teknikal Malaysia Melaka

TABLE OF CONTENT

© Universiti Teknikal Malaysia Melaka

© Universiti Teknikal Malaysia Melaka

© Universiti Teknikal Malaysia Melaka

LIST OF TABLE

© Universiti Teknikal Malaysia Melaka

LIST OF FIGURE

© Universiti Teknikal Malaysia Melaka

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

The current attendance system is not efficient enough. Therefore, a software part of the smart attendance system is proposed in this project. It is used to monitor student attendance and the system is also designed to contain information such as name, matrix number, date and etc. This project work includes the development of student information into a database. Then, the developed system software is to be combined with the hardware of fingerprint and card reader to make the complete system. Then it is connected to a personal computer. The database is developing through the interface using Microsoft Visual Basic.Net and the data is stored to the database using Microsoft Access 2007.

1.3 PROBLEM STATEMENT

At the moment, most of the attendance systems that are being used in universities still are written a piece of paper. For classes, tutorial and laboratory session the student

© Universiti Teknikal Malaysia Melaka

still have to sign their signature on the attendance sheet. This method is not flexible because the risk of losing the attendance data is very high. If the attendance sheet is missing, the attendance data will be lost. Other than that, unethical problem may occur such as cheating in signature. For example, a student does not attend his class but his attendance form has been signed by other student. This system is proposed to overcome these problems. Besides that, since the proposed system also record the time, the lecturer can monitor the punctuality of the students too.

1.2 OBJECTIVE

The objectives of this proposed project are:

a) Overcome the old attendance system which the attendance has to be taken manually.
b) Design a database that supports such a system that has been mentioned earlier.
c) Develop a smart attendance system that be implemented in UTeM classes, laboratory and etc. by combining the software with the proposed hardware

1.4 METHODOLOGY

1. Literature reviews
   - Information on GUI and database is found
2. Database development
   - The database is designed according to the data required
   - The appropriate GUI for the system is also develop
3. Testing Database
   - Once the database development work completed, the system is tested whether is working successfully

© Universiti Teknikal Malaysia Melaka

4. Testing System
   - The entire system is then tested
5. Troubleshooting
   - If the system does not work, the troubleshooting has to be done to identify the problem
6. Thesis Writing and Presentation

1.5 THESIS LAYOUT

Chapter I: Introduction

An overview or introduction of this project is designed in Chapter I. In this chapter the objectives, problem statement and the methodology for this project are also described

Chapter II: Literature Review

General information of developing database and the corresponding software is learned in Chapter II. It has a list of software that related to and possible to be used in this project.

Chapter III: Methodology

In this chapter, all research, findings and approaches that relevant is explained. The methodology and process flow of this project are also shown in this chapter.

Chapter IV: Result

All the current project result is shown in Chapter IV such as screening of the interface. The entire windows that are needed in this project are also included.

Chapter V: Discussion and Conclusion

The whole process involves in this project is discussed in Chapter V. The achievement of the project objective is also being concluded. Further work that possible to be done to upgrade the project is suggested.

© Universiti Teknikal Malaysia Melaka

CHAPTER II

LITERATURE REVIEWS

In this chapter, the studies on which Microsoft Visual is suitable for my GUI and also database development which will communicate with the GUI using Microsoft Access to store database are included using Microsoft Visual Basic.Net is chosen for the interface and Microsoft Access 2007 is used for database development. These two software are easy to understand making them suitable for the project.

2.1 OVERVIEW

Smart attendance system is an upgrade for the system that is used right now which is been taken manually. This system is effective to avoid student from cheating. Beside that, it can prevent any data lost if the attendance sheet is missing. Other than that, this system also helps avoiding student wasting. The smart attendance system has GUI to interact with the hardware device and can store information using database. This system will practically show the student detail, subject that been taken by the student and because this is an attendance system, it will display the time of student arrival for every classes, tutorial and lab session.

© Universiti Teknikal Malaysia Melaka

2.2 MICROSOFT VISUAL STUDIO

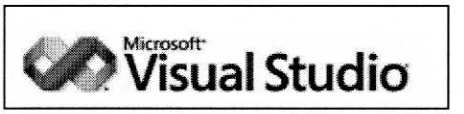

Figure 2.1: Visual Studio Logo

Microsoft Visual Studio is Microsoft's flagship software development product for computer programmers. It is centered on an integrated development environment which lets programmers create standalone applications, web sites, web applications, and web services that run on any platforms supported by Microsoft's .NET Framework. Supported platforms include Microsoft Windows servers and workstations, PocketPC, Smartphones, and World Wide Web browsers.

Visual Studio includes the following:

a) Visual Basic 6.0
b) Visual Basic (.NET)
c) Visual C++
d) Visual C#
e) Visual J#
f) ASP.NET

2.2.1 Microsoft Visual Basic 6.0

Visual Basic (VB) is an event driven programming language and associated development environment from Microsoft for its COM programming model. Visual Basic was derived from BASIC and enables the rapid application development (RAD) of GUI applications, access to databases using data access object (DAO), remote data object (RDO), or activeX data object(ADO), and creation of ActiveX controls and objects. Scripting languages such as VBA and VBScript are syntactically similar to VB, but perform differently. A programmer can put together an application using the components provided with VB itself.

© Universiti Teknikal Malaysia Melaka

2.2.1.1 Characteristics in VB

VB has the following rare character:

a) Boolean constant `True` has numeric value −1. This is because the Boolean data type is stored as a 16-bit signed integer. In this construct −1 evaluates to 16 binary 1s (the Boolean value `True`), and 0 as 16 0s (the Boolean value `False`). This is apparent when performing a Not operation on a 16 bit signed integer value 0 which will return the integer value −1, in other words `True = Not False`. This inherent functionality becomes especially useful when performing logical operations on the individual bits of an integer such as `And`, `Or`, `Xor` and `Not`. This definition of `True` is also consistent with BASIC since the early 1970s Microsoft BASIC implementation and is also related to the characteristics of microprocessor instructions at the time.

b) Logical and bitwise operators are unified. This is unlike all the C-derived languages (such as Java or Perl), which have separate logical and bitwise operators. This again is a traditional feature of BASIC.

c) Variable array base. Arrays are declared by specifying the upper and lower bounds in a way similar to Pascal and Fortran. It is also possible to use the Option Base statement to set the default lower bound. Use of the Option Base statement can lead to confusion when reading VB code and is best avoided by always explicitly specifying the lower bound of the array. This lower bound is not limited to 0 or 1, because it can also be set by declaration. In this way, both the lower and upper bounds are programmable. In more subscript-limited languages, the lower bound of the array is not variable. This uncommon trait does not exist in Visual Basic .NET and VBScript[1].

© Universiti Teknikal Malaysia Melaka

OPTION BASE was introduced by ANSI, with the standard for ANSI Minimal BASIC in the late 1970s. The extension to using a form of DIM A(1976 TO 2020) was first introduced in BBC Basic, found on the BBC micro, which in turn was influenced by COMAL.

a) Relatively strong integration with the Windows operating system and the Component Object Model.
b) Banker's rounding as the default behavior when converting real numbers to integers.
c) Integers are automatically promoted to reals in expressions involving the normal division operator (/) so that division of an odd integer by an even integer produces the intuitively correct result. There is a specific integer divide operator (\) which does truncate.
d) By default, if a variable has not been declared or if no type declaration character is specified, the variable is of type `Variant`. However this can be changed with Deftype statements such as `DefInt`, `DefBool`, `DefVar`, `DefObj`, `DefStr`. There are 12 `Deftype` statements in total offered by Visual Basic 6.0. The default type may be overridden for a specific declaration by using a special suffix character on the variable name (# for Double, ! for Single, & for Long, % for Integer, $ for String, and @ for Currency) or using the key phrase As (type). VB can also be set in a mode that only explicitly declared variables can be used with the command `Option Explicit`.

2.2.1.2 Language Features

VB is designed to be easy to learn and it is use to allow programmers to create simple GUI applications and also develop fairly complex applications as well. Programming in VB is a combination of visually arranging components or controls on a form, specifying attributes and actions of those components, and writing additional lines

© Universiti Teknikal Malaysia Melaka

of code for more functionality. Since default attributes and actions are defined for the components, a simple program can be created without the programmer having to write many lines of code. Performance problems were experienced by earlier versions, but with faster computers and native code compilation this has become less of an issue.

Forms are created using drag and drop techniques. A tool is used to place controls (e.g., text boxes, buttons, etc.) on the form (window). Controls have attributes and event handlers associated with them. Default values are provided when the control is created, but may be changed by the programmer. Many attribute values can be modified during run time based on user actions or changes in the environment, providing a dynamic application. For example, code can be inserted into the form resize event handler to reposition a control so that it remains centered on the form, expands to fill up the form, etc. By inserting code into the event handler for a keypress in a text box, the program can automatically translate the case of the text being entered, or even prevent certain characters from being inserted.

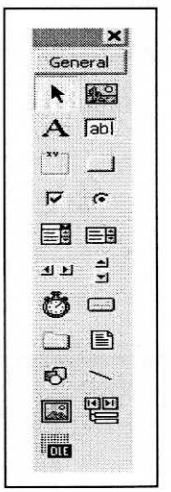

Figure 2.2: Tool box from Visual Basic 6.0

Visual Basic can create executables (EXE files), ActiveX controls, DLL files, but is primarily used to develop Windows applications and to interface web database systems. Dialog boxes with less functionality can be used to provide pop-up capabilities. Controls provide the basic functionality of the application, while programmers can insert

© Universiti Teknikal Malaysia Melaka

additional logic within the appropriate event handlers. For example, a drop-down combination box will automatically display its list and allow the user to select any element. An event handler is called when an item is selected, which can then execute additional code created by the programmer to perform some action based on which element was selected, such as populating a related list.

Alternatively, a VB component can have no user interface, and instead provide ActiveX objects to other programs via Component Object Model (COM). This allows for server-side processing or an add-in module.

The language is garbage collected using reference counting, has a large library of utility objects, and has basic object oriented support. Since the more common components are included in the default project template, the programmer seldom needs to specify additional libraries. Unlike many other programming languages, VB is generally not case sensitive, although it will transform keywords into a standard case configuration and force the case of variable names to conform to the case of the entry within the symbol table entry. String comparisons are case sensitive by default, but can be made case insensitive if so desired.

2.2.2 Microsoft Visual Basic.NET

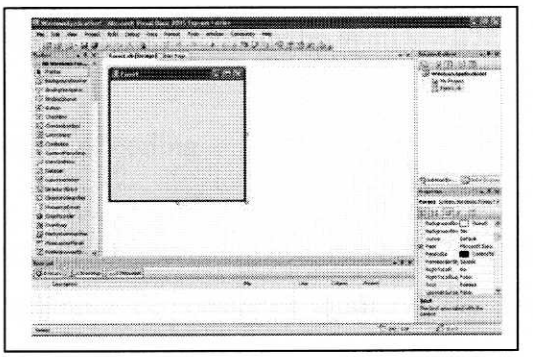

Figure 2.3: VB.NET interface

© Universiti Teknikal Malaysia Melaka

VB.NET is an object-oriented computer language that can be viewed as an evolution of Microsoft's Visual Basic (VB) implemented on the Microsoft .NET framework. Its introduction has been controversial, as significant changes were made that broke backward compatibility with older versions and caused a rift within the developer community.

The great majority of VB.NET developers use Visual Studio .NET as their Integrated Development Environment (IDE). SharpDevelop provides an open-source

alternative IDE. Like all .NET languages, programs written in VB.NET require the .NET framework to execute. An IDE, also known as integrated design environment and integrated debugging environment, is a type of computer software that assists computer programmers in developing software.

IDEs normally consist of a source code editor, a compiler and/or interpreter, build-automation tools, and usually a debugger. Sometimes a version control system and various tools to simplify the construction of a GUI are integrated as well. Many modern IDEs also integrate a class browser; an object inspector and a class hierarchy diagram, for use with object oriented software development. Although some multiple-language IDEs are in use, such as the Eclipse IDE, JDeveloper, Code::Blocks, Komodo IDE, NetBeans, Borland Developer Studio, KDevelop or Microsoft Visual Studio, typically an IDE is devoted to a specific programming language, as in the Visual Basic IDE or in the WinDev IDE.

2.2.3 Comparative samples of coding

There are several differences coding between both VB.NET and VB 6.0. This are the following simple example demonstrates similarity in syntax between VB and VB.NET. Sample of coding:

© Universiti Teknikal Malaysia Melaka

Table 2.1: Pop Up Message: Hello, World

VB 6.0	VB.NET

`Private Sub Command1_Click()`
`  MsgBox "Hello, World"`
`End Sub`
	`Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click`
	`  MessageBox.Show("Hello, World")`
	`End Sub`

The above examples will pops a message box saying ***"Hello, World"*** with an OK button.

All the procedure calls must be made with parentheses in VB.NET, whereas in VB6 there were different conventions for functions (parentheses required) and subs (no parentheses allowed, unless called using the keyword `Call`). The names `Command1` and `Button1` are not obligatory. However, these are default names for a command button in VB6 and VB.NET respectively.

The following example demonstrates a difference between VB6 and VB.NET. Both examples unload the active window.

Table 2.2: Example coding

VB 6.0	VB.NET

`Private Sub cmdClose_Click()`
`  Unload Me`
`End Sub`
	`Private Sub btnClose_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnClose.Click`
	`  Me.Close()`
	`End Sub`

The '`cmd`' prefix being replaced with the '`btn`' prefix, conforming to the new convention previously mentioned.

© Universiti Teknikal Malaysia Melaka