Project Number: JHZ/YHM-HK01 - 48

Facilitating Course Database Management for Caritas Adult & Higher Education Service, Hong Kong

An Interactive Qualifying Project Report

submitted to the Faculty of

WORCESTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirements for the

Degree of Bachelor of Science

By

Victor Chen

Cuong Diep

De Ming Guan

John F Zeugner

Professor John F. Zeugner, Co-Advisor

Professor Yi Hua Ma, Co-Advisor

Abstract

This project, sponsored by the Caritas Adult & Higher Education Service in Hong Kong, created a new database management system for the 900 classes offered each term at Caritas' 26 educational sites. Through interviews the project team understood the deficiencies and redundancies in the current course registration and publication process and provided a streamlined, improved database entry system that built off the traditional components so as to provide the Caritas staff with minimal adjustment problems.

Acknowledgements

We would like to thank the Caritas Adult & Higher Education Service—Hong Kong; without their presence and support our project would not have come about. We would like to thank Amy Chan, our project coordinator, for sponsoring the Hong Kong project site and for her support throughout the project. We would like to thank Catherine Li, our liaison, for her insight into Caritas' course management system, for her guidance of our work and for her help and presence in our project. We would like to thank Anthony Leung, our technical advisor, for his vision and technical directions, his help in developing the database application and for being our mentor for this project. We would like to thank the staff of Caritas' Office of Academic Affairs for their assistance and friendship, which made our project experience in Hong Kong much more enjoyable. Thank you to all those in the IGSD who helped to reopen the Hong Kong project site; without them we would never have gotten this great experience. We would like to extend special thanks to Dean Paul Davis and Associate Dean Richard Vaz for their suggestions and comments pertaining to our project. Last, but not least, we thank our project advisors Professor Zeugner and Professor Ma for their relentless help, support and guidance, for working so very hard to reopen the Hong Kong project site and for giving us a chance.

Authorship

The Hong Kong IQP Caritas Database Team, Victor Chen, Cuong Diep and De Ming Guan wrote all the contents of this report.

Contents

Executive Summary

This Interactive Qualifying Project team worked directly in conjunction with Caritas Adult & Higher Education Services (CAHES) Office of Academic Affairs to design and implement a course database management system to facilitate the storage, retrieval and distribution of course information. The CAHES division of Caritas—Hong Kong, a nonprofit and non-government charity organization, offers in excess of seven hundred different courses. These courses are targeted toward adults who are interested in broadening their résumé of skills and those who did not complete the education of the Hong Kong school system and want a second chance. CAHES came to the conclusion that their use of paper records and Microsoft Excel worksheets to organize and store their course information was inefficient and vulnerable to typographical errors.

The initial interviews with the liaisons for this project revealed the current course information management system that CAHES employed. After courses are developed and then approved at the Office of Academic Affairs, they are stored in a Microsoft Excel worksheet. This worksheet is subsequently sent to each of CAHES' twenty-six educational sites where they decide upon which courses they will offer for the coming season and compile a list of logistical information pertaining to each class. This information is sent directly to the publisher of the course catalog booklet for printing, thus bypassing the Office of Academic Affairs and the possibility of checking for invalid course offerings and other errors. To expand the public availability and awareness of current and future course offerings, the Office of Academic Affairs manually retypes the course and class information from the course catalog booklet, an activity that is both

redundant and susceptible to error, and places it on the CAHES website and also on the Hong Kong government's Education and Manpower Bureau (EMB) website.

The proposed course information management system improves upon the current system through the use of a custom developed database application. In the new system, the educational sites submit their class information to the Office of Academic Affairs in the form of a specially formatted Excel worksheet instead of to the published course catalog. The decision to use Excel worksheets was derived from the facts that Excel files were currently in use, the staff was already familiar with the use of the Excel application and the new system was to improve upon the current system and not to replace it. The course information from the Office of Academic Affairs and the class information from each educational site are imported into the database application. With over nine hundred classes held per season and seven hundred different courses, the monumental task of verifying each class and each course led to the implementation of checker functions in the database application. The checker functions would verify the course and class worksheets for any errors before they can be imported into the database. The idea of a checker function was extended to each of the educational sites. Each educational site would check their own Excel worksheet of classes before submission, thus reducing the transferring of the worksheet to-and-from the Office of Academic Affairs for corrections. With the current season's courses and classes electronically stored, the task of selecting, retrieving and formatting the information for their respective recipients was done by the database application without error.

Educating the staff at the Office of Academic Affairs of the new system and database application was the final step of this project. Training sessions were held with the potential users of the database application where each aspect and functionality of the database application was explained and for future reference, a comprehensive user's manual documenting the database application was created.

In this IQP, the human factor was crucial for the successful fruition of this project. Design considerations pertaining to both the functions and user interface of the database application were greatly affected by the responses of the users during discussions and training sessions. The database application provided vital functionalities in order to fulfill its role in the enhanced system. We hope this project will reduce workload for the staff of CAHES and therefore allow CAHES to fulfill their pledge to their students better.

1. Introduction

In today's society, it is inconvenient for any large organization to store data on paper. Electronic data storage is easier to manage; more secure and has many other advantages over traditional paper storage methods especially when large storage is necessary. However, like everything in life, there are tradeoffs. There are advantages and disadvantages to the usage of electronic storage and different organizations might want to consider these tradeoffs for what is best suited for them. Electronic storage of information permits remote access of the stored information, allowing the same information to be catered simultaneously to multiple recipients thus increasing the consistency and accessibility of the data. Furthermore, electronic data storage provides enhanced security through the ability to dole out appropriate levels of access to different classifications of users. On the downside, electronic data storage is much more costly than paper data storage to setup and manage and if security breaches are to occur, they are much more severe because all the data are in one location for the intruder to steal, corrupt or destroy. CAHES is in the process of enhancing their data storage routine with the use of an electronic data storage system.

Caritas is the official social service and educational agency of the Roman Catholic Church. Caritas Organization has four major service divisions in Hong Kong: social work services, education services, medical services and community & hospitality services. Caritas Adult & Higher Education Service is one of the four branches under the education services division. CAHES began its services in 1963; its goal is to promote lifelong education. CAHES offers a program that trains students toward better education, working skills, community participation and personal development. There are a total of twenty-six educational sites registered with the education services. These schools offer programs with wide contents from basic information to higher-level education. For CAHES to store their entire course information on paper or even electronically without a database management system (DBMS) would be counterproductive. Using a DBMS to manage all their data manipulations should decrease workloads of the employees and allow increased security for their data. A database application that includes search and sort functions would help CAHES in fulfilling many tasks in a shorter amount of time. The goal of this project was to design and implement a database application for CAHES.

Traditionally, CAHES had all their data keyed in manually in an Excel file for their master database. They would later have to manually key it in once again for their websites and a third time for the Hong Kong government's Education and Manpower Bureau (EMB) website. They cannot use the same data file because the information needed for each use is different. This redundant entry allows for error and is time consuming. CAHES would like to develop a program that stores all of their data and outputs the chosen data to whichever location they desire. With such an application workloads would be lightened and errors will be minimized.

The Hong Kong government runs a continuing education website. They have requested CAHES if they would like for their information to be seen on this website. This website will help CAHES grow; already CAHES is among the biggest continuing education organizations in Hong Kong. It would be in CAHES' best interest to be included in this avenue of publicity. Currently, however, CAHES' participation in the EMB continuing education website would mean that CAHES would need manually to reenter their course information into another Excel file one more time.

There is much work that is necessary before such an application can be developed. Among them is designing the best data structure for CAHES, one that takes into account their needs and uses for the data. There are many aspects to consider when developing such a complicated system. Database management experts have come up with different methods to help evaluate relationships and attributes of data in order to structure any database procedurally. Among these are the Entity Relationship (E/R) Model and the Object Definition Language (ODL) Model. These models will show the relationships between the data in different ways so that a developer will then be able to extract the best structure from the two models.

Many other aspects are to be considered in order for a good database application to be developed. It is crucial to know about CAHES' current course management system and organization. Understanding CAHES and the organization of its classes helped in structuring the database. It allowed us to be able to fulfill more of CAHES' needs as the project progresses. We studied the general functions of a database application and what tools a good database program will provide for its users. Providing the tools is only half the task because it is also very important to make the tools easy to use. Most users will not spend their time on exploring the possible functions of a particular application. If a function is not made easily understandable then providing it will only be a waste of the implementers' time. The overall interface of the application should be user friendly. All tools and functions should be easily accessible and applicable. With all of these guidelines in mind, detailed information of the above issues helped in the implementation of a fully functional database application.

Development of an intricate database system requires careful planning. A methodology should be formed after all background information and objectives for the project has been acquired. The methodology needs to be a detailed strategy of how the problem is to be approached and conquered. The previous research will provide guidelines to be followed while creating the methodology.

It would be ideal for the final application to be able to sort and search through the classes of CAHES. It should be able to output four different preset formats. One for the CAHES database, another for the course catalog booklet, another for the CAHES website and then one for the Hong Kong government's EMB website. The employees of CAHES should be able to benefit from this new data storage method and possibly improve the service of the organization to the students.

The objective of this Interactive Qualifying Project was to solve the deficiencies of CAHES' current course information management system by analyzing the interview results to understand the shortcomings of the current human-machine interface. From these results we determined the weaknesses of the current system. These weaknesses will be then eliminated by our proposal of a new course information management system built around a database application to store their course information. At the end of this project, presentations will be made to ensure that the CAHES staff was knowledgeable of the enhanced course information management system. Training sessions will be held and a user manual for the database application will be prepared, so that users can easily adapt to the application and utilize the application with the utmost proficiency. We believe that this new system would reduce the workload of the CAHES staff and improve the integrity and availability of their course information.

2. Literature Review

2.1 Introduction

A major attraction of the Internet is its instant access to information. Each time users enter a website address into their web browsers, they are demanding that the information on that website be sent to them immediately. Meanwhile, accessed websites are storing information about their visitors and their visitors' preferences and habits. Online stores have their entire inventory online with up-to-the minute prices and availability. Universities have the schedules for each student online and ready for immediate access. All of this information about users, products, and schedules is not located in the abstract concept of cyberspace, but is stored in something very real called a database.

2.2 Advantages and Disadvantages of Databases

The use of databases in general provides many advantages including and not limited to improvements in information management, security and the enforcement of standards. There are three facets to information management: file redundancy, consistency and accessibility. An example would be a corporation with many departments, each of which have a copy of the current sales report on their local computers. First of all, a copy of the sales report in each department is redundant and depending on the size of the file, a waste of storage space. Secondly, each of the departments could be correcting errors or updating the sales report pertaining to their own department, thus each department could end up having a different version of supposedly the same sales report and the information will no longer be consistent throughout the corporation. And lastly, one department could need the most updated version of the sales report from another department, but they might not have access to the other department's updated figures and be unable to complete their work. A database will eliminate redundancy by keeping only one copy of the sales report in one central location. If the sales report ever needs updating, then the sole copy in the database will be updated and all of the departments will have access to the most up-to-date copy and be consistent. The security of the information is improved by having only one copy of the sales report available to users with the proper authority instead of having multiple copies on different computers that might not be as secure. And for future sales reports, the data must be entered into a database in a standardized method and therefore does not allow each department to have their own style of data entry, making the format of the sales report uniform. A database is not a panacea for information management; there are drawbacks to using a database.

The disadvantages of using a database include the cost of employing skilled and knowledgeable personnel, security breaches and database dependency. The creation and maintenance of a database require skilled IT (information technology) professionals. These IT professionals do not come cheaply. According to datamasters.com, an IT jobs website, a database administrator in the United States can make anywhere from $60,000 to $120,000 a year. For a non-profit organization such as Caritas, this money can be better spent for the students and facilities and the database left in the hands of voluntary academic team. Security has been mentioned before to be improved by a database. But if a security breach does occur, severity of the damage is increased because the intruder has access to all of an organization's data instead of just the information located in each individual computer. Once a database has made its home inside of an organization and become a fact of the workplace environment, then that organization may become too dependent on it. Any mishaps with the database could lead to the loss of all the organization's records. To prevent this scenario, periodic backups of the database must be made so that in case of such unfortunate events, the database can be restored with the most recent copy.

So far a general overview of what databases are and their advantages and disadvantages have been given. The next sections of this literature review deal with the more technical aspects of what databases are and the process of how they are designed.

2.3 Background on Database Management Systems

A database is a collection of information managed by a DBMS (database management system) (Ullman, 1997). Other forms of data collection and storage include catalogs, repositories of printed material and file systems. These methods house information equally as well as a database, but they lack the power and functionality that a database management system provides to their users. This project will deal mainly with the design and implementation of a database and the use of Microsoft Visual FoxPro. A brief explanation of database management systems and a more thorough treatment of databases will help the reader to understand and appreciate the scope of this project better.

Database management systems evolved from file systems. A file system is how data is setup and stored electronically. A file system can be found on most modern operating systems such as Windows, MacOS and Linux. The two main components to a file system are *files* and *directories*. Files are a collection of data that has a distinct filename. Directories, more commonly known as folders, hold files and other directories and also have distinct directory names. Therefore there is one principal directory that holds all other directories and files. A fitting analogy would be a tree. The principal directory is the root of the tree. Each directory is a bough; branch or twig and the files are the leaves to this tree.

There are advantages and limitations to a file system. An obvious advantage of a file system arises from its use of electronic storage mediums to store its information, thus reducing the physical space used to house the data and allowing for greater data density as compared to catalogs and printed materials. Files and directories can be organized so that related files can be stored in the same directory and related directories can be stored in the same directory. A limitation to this storage method is that there is no way to correlate files that are partially related and that are stored in different directories. There are also no means to ask questions about the data, also known as a query, in the files to retrieve, modify or delete data that has met a certain criteria. Finally, files systems do not protect the data from being accessed and modified by two users simultaneously. Thus one user may unwittingly overwrite the changes made by another simultaneous user. Database management systems were created to address these limitations.

A database management system is expected to provide certain active and passive functionalities for the user. A DBMS must allow the user to create new databases and to determine the desired schema. A schema is how the data is set up using relationships between the contents of the data and not through the use of files and folders. A DBMS allows the user to query and to modify the data using a suitable data-manipulation language, such as SQL (Structured Query Language). In the background, a DBMS must provide means to keep the data safe from unauthorized users and to maintain the integrity of the data by not allowing simultaneous changes to the same piece of information (Ullman, 1997).

2.4 Database Design

The design of a database begins with the analysis of the information that will be stored in the database. The information is divided into identifiable items called *objects* and descriptions about the objects called *attributes*. There are two standard techniques used to model the data in this manner. The first is a more traditional and graphical approach called the entity-relationship (E/R) model. The second type of model is the newer Object Definition Language (ODL), which is more text oriented geared towards the direct implementation in an object-oriented programming languages. Neither of these models will be the actual framework for a database. Instead, the purpose of using these two models is to pick out the best properties from each of these models and then use them to create the relational data model that will be the actual framework for the database.

2.4.1 Entity-Relationship Model

The E/R model uses shapes and connecting lines to represent objects and their attributes and relationships (Ullman, 1997). Suppose that we want to make a database about books. A database of books would contain information on the title, author, publisher and date of publication and International Standard Book Number (ISBN) of each book. The entities, E/R model term for objects, in this database are Books, Authors and Publishers. It might seem surprising that a database about books would contain information on authors and publishers as separate entities and not consider them as just attributes of a book. The reason that books is not the only entity in this database is because unlike title, date of publication and ISBN, authors and publisher are more than just descriptions of a book, they are independent and identifiable objects that are *related* to books. The Books entity does not represent one book, but it represents all books in existence. The same applies for the Authors and Publishers entities. In an E/R diagram, blocks represent entities. The E/R diagram for the books database so far is shown is Figure 2.1.

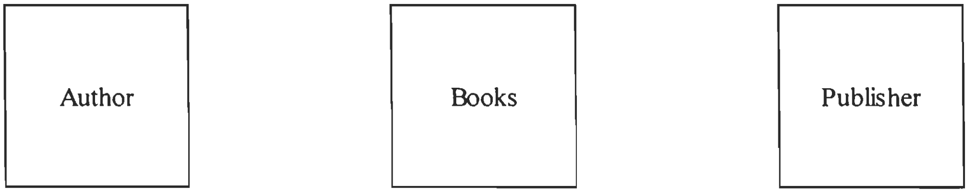


Figure 2.1 The entities in the books database.

The next step is to link each entity with their attributes. So far we know that the entity Books has the attributes title, dateOfPublication and ISBN. The Authors and Publishers entities have attributes of their own as well or else they would just be attributes of the Books entity. The entity Authors has the attributes name and dateOfBirth. The entity Publisher has the attributes name and address. (These attributes have been chosen semi-arbitrarily for the sake of having enough attributes to consider Authors and Publishers as entities.) Attributes that have names longer than one word have their names concatenated to form one, long attribute name. Attributes are represented as circles in E/R diagrams. A line connecting the entity with an attribute represents the relationship between an entity and its attributes. The updated E/R diagram is shown in Figure 2.2.

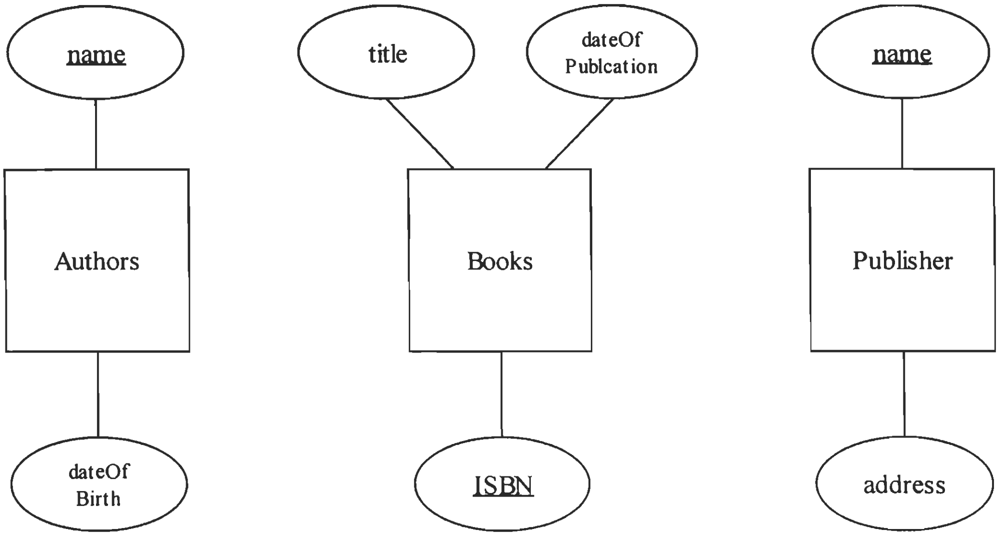


Figure 2.2 Entities with attributes in books database.

The attributes that are underlined are called *keys*. A key is the minimal set of attributes that is able to uniquely identify one object from the set of objects that an entity represents. The title of a book is not a key because more than one book can have the same title. An ISBN is a key because each book is given a unique ISBN number by the International ISBN Agency. When someone looks up a book by a certain title, then it is possible that more than one book will be found. But when looking up books using an ISBN, the key for books, only one book will be found.

The final step in creating an E/R diagram for the books database is to incorporate the relationships between the entities themselves. The relationship between two entities is called a *relationship.* Each relationship is represented by a diamond and is given a name describing that relationship. The relationship between the Authors and Books entities can be called *Author-of.* This relationship can be read as "Authors is the Author-of Books." The relation also could have been called *Written-by*, but that is only a matter of preference. A suitable relationship between the entities Books and Publisher is *Published.* Adding the relationships to the current E/R diagram yields the completed E/R diagram as shown in Figure 2.3.

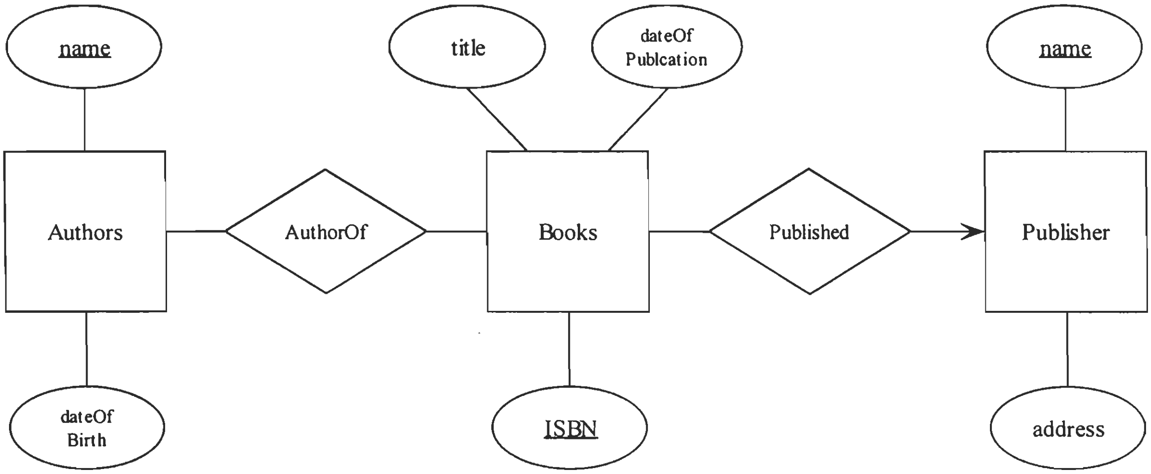


Figure 2.3 Completed E/R diagram for the book database.

An element regarding E/R diagrams that has not been discussed so far is the use of lines with arrowheads at the end, such as the one used in connecting the Published relationship with the Publisher entity in Figure 2.3. A line connecting an entity with a relationship means that the relationship is many-to-many. In the case of Authors and Books, the relationship states that an author can write many books and a book can be written by many authors. In the case of Publishers and Books, the relationship is many-to-one. It can be read as a publisher can publish many books but a book can only be published by one publisher. In other words, the arrowhead pointing to the Publisher entity means that there is only one publisher to any book. Furthermore, if the entities Authors and Publishers were in a relationship that had arrows pointing to both of them, then that relationship would be one-to-one and could be read as there is only one Publisher to any Author and only one Author for any Publisher. The Author-Publisher relationship was only used as an example of a one-to-one relationship and is not a part of the book database.

With the E/R model completed a different approach through the use of an ODL model can lead to different views of the same database.

2.4.2 Object Definition Language Model

The Object Definition Language is a pseudo programming language used to model the information in a collection of data. ODL's structure closely resembles the syntax of a C++ structure. An advantage of this design is that ODL can be quickly and easily ported to databases that are built around object-oriented programming (OOP) languages such as C++ and Smalltalk-based OODBMS (object-oriented database management systems). Databases modeled in ODL seem more formally structured and resemble actual database structures, but much like the E/R model, they are just one of the many ways to outline the information (Ullman, 1997).

The basic object in ODL is called a class as opposed to an entity in the E/R model. To define a class in ODL, the keyword *interface* is used, followed by the name of class. The attributes and relations for each class are stored inside of a set of curly brackets after the declaration of the class. Just as before, we will begin by just defining the class structures for the three classes in the books database example, as shown in Figure 2.4.

```
interface Book {
};
interface Author {
};
interface Publisher {
};
```

Figure 2.4 Declaration of classes for the book database.

Next comes the inclusion of attributes for each class. The assigning of attributes is a slightly more involved process than in the E/R model. Anyone familiar in programming will follow the example and understand the process more readily. To define an attribute, the keyword attribute followed by the type of attribute and then the name of the attribute. For example, a number without a decimal component is of the type integer and a word would be of the type string. The attributes title for Book, name for Author and name and address for Publisher are of the type string. The attributes dateOfPublication and ISBN for Book and dateOfBirth for Author are of the type integer. The declaration of a key is no longer done with the underlining of the key attribute as in the E/R model. A key is defined between the name of the class and the opening curly bracket. Placed inside a set of parentheses are the keyword key and the name of the attribute that is the key. The class declarations with the attributes included are shown in Figure 2.5.

```
interface Book (key ISBN) {
        attribute string title;
        attribute integer dateOfPublication;
        attribute integer ISBN;
        };

interface Author (key name) {
        attribute string name;
        attribute integer dateOfBirth;
        };

interface Publisher (key name) {
        attribute string name;
        attribute string address;
        };
```

Figure 2.5 Inclusion of attributes for the book database.

Lastly we will define the relationships between the classes. Once again, the declaration of a relationship begins with a keyword, in this case *relationship*, followed by which class the relationship is with and the name of the relationship. Also, it is standard to write the inverse of the relation on the following line by using the keyword *inverse*, the name of the related class followed by a set of double colons and then the appropriate name of the inverse relationship. All of these techniques may seem confusing, but an example relation from the class Book to Author should help as shown in Figure 2.6. The <*set*> notation denotes that there could be one or more authors related to this one book.

```
relationship<set> Author writtenBy
        inverse Author:: authorOf;
```

Figure 2.6 Sample relationship inside of the Book class.

Excluding the <set> notation would mean that all books could only have one author. The completed ODL model is shown in Figure 2.7.

The E/R diagram and the ODL model of the book database accomplish the same goals but through different means. The goal of using two different models is to discard their weaknesses and pick out their strengths and use them in the relational model that will be used as the structure for the implemented database.

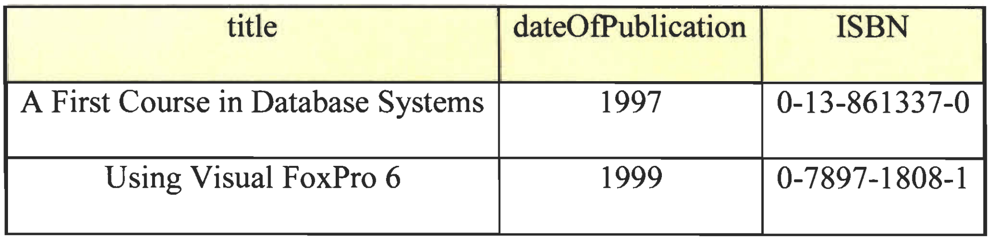
```
interface Book (key ISBN) {
        attribute string title;
        attribute integer dateOfPublication;
        attribute integer ISBN;
        relationship<set> Author writtenBy
            inverse Author::authorOf;
        relationship Publisher publishedBy
            inverse Published::published;
        };
    interface Author (key name) {
        attribute string name;
        attribute integer dateOfBirth;
        relationship<set> Book authorOf
            inverse Book::writtenBy
        };
    interface Publisher (key name) {
        attribute string name;
        attribute string address;
        relationship<set>Book Published
            inverse Book::PublishedBy
        };
```

Figure 2.7 Completed ODL model for the book database.

2.4.3 The Relational Data Model

In this section of the literature review we will discuss the relational data model. We will begin by describing the characteristics of the relational model. Then we will go into detail the process which E/R diagrams and ODL models are converted into the relational model by using the example books database as a guide. Lastly we will examine some of the modifications that must be applied to a relational database to improve its efficiency and consistency.

The relational data model can be described as a two-dimensional table called a *relation* (Ullman, 1997). Each table parallels an E/R entity or an ODL class because it has a name, represents an object and has attributes. The column headings in a relation are the attributes for that relation. As an example, Figure 2.8 shows the relation Book with same attributes as in the E/R diagram with the information from two books used in this literature review.



title	dateOfPublication	ISBN
A First Course in Database Systems	1997	0-13-861337-0
Using Visual FoxPro 6	1999	0-7897-1808-1

Figure 2.8 The Book relation with attributes.

Each row in the relation, besides the row of column headings, is known as a *tuple*. For each tuple, there is only one value for each of the attributes in the relation. A tuple belonging to a relation must be different from all of the other tuples in the same relation.

If two tuples in the same relation are exactly the same attribute for attribute, then they represent the same object and are unnecessarily repeated in the relation. This restriction does not mean that two tuples cannot have the same value for the same attribute. A case in point is that many books are published in the same year and are not the same book.

The reduction of a relation down to its names and attributes is called the schema for the relationship. The schema for the sample books relation is shown in Figure 2.9.

Book(title, dateOfPublication, ISBN)

Figure 2.9 The schema for the Book relation.

Similarly a tuple can be reduced down to its attributes as a way to display it separately from the rest of the relation, as shown in Figure 2.10. These relations up to now have only showed us how to describe different types of objects. They have yet to show the relationship between two different objects, a necessity that the E/R diagram and ODL model have already done. At this point we will understand why we went through the process of modeling the data in two different ways before designing the actual database.

(A First Course in Database Systems, 1997, 0-13-861337-0)

Figure 2.10 A single tuple.

3. Methodology

This IQP consists of two phases. The first phase of the project involves designing the database structure and the second phase entails the use Microsoft Visual FoxPro to implement and build the database application. The staff at the Office of Academic Affairs of Caritas Adult & Higher Education Service will be the primary users of this database application. The functionalities of the database application will include the ability to search and sort through CAHES' course database easily, check and restrict the entry of any invalid data into the database and the ability to produce Excel files with different data fields and formats. The Excel files with different formats are used for the following four purposes: the CAHES website's course search engine, the Hong Kong government's Education and Manpower Bureau website, CAHES course booklets and the master database copy at the central office. The government's EMB website has a database of all the continuing education courses available in Hong Kong, and as a result CAHES has to submit an Excel file with the required information for each of their courses to the EMB database. Currently, a CAHES staff member has to reenter the course information four different times each season, because each use has a different format requirement. In response CAHES office desires an application that can reduce the workload of their staff and manage the database in a more efficient way. The following methodology sections will describe the systematic procedures that are adapted for this project.

3.1 Design of the Database structure

The design criteria are the requirements and goals for a database. A central requirement for the database application is that it must comply with CAHES' current data storage methods. Thus the existing data files were examined and more information about the current system was gathered. It is very important to know what type of data and information is being dealt with before beginning the design of the database structure. Interviews were conducted with Catherine Li and Anthony Leung, our liaisons at CAHES, immediately after the first day project orientation. For Catherine's interview, the goal was to find out more information about the Caritas Organization, their course development procedures and their expectation for this project. Anthony was introduced to the project group as the technical mentor for this project; therefore Anthony's interview was focused on the technical aspects. Questions about the targeted users and goals of the database application were brought up during these interviews. The application's interface was implemented based on the computer knowledge of these users. The interviews with Catherine and Anthony were aimed at drawing out feedbacks regarding what information will be stored in the database, how the database will be used, who will be using it and what are their priorities.

At the conclusion of these interviews, all the gathered information and ideas were compiled into a comprehensive set of criteria for the database structure design. By examining these criteria and the data files a database structure was designed. The database structure design contains Entity Relationship Model (E/R) diagram as shown in Figure 3.1, tables as shown in Figure 3.2 and Object Definition Language Model (ODL) (Appendix G) (descriptions of these models are discussed in the literature review of this report).

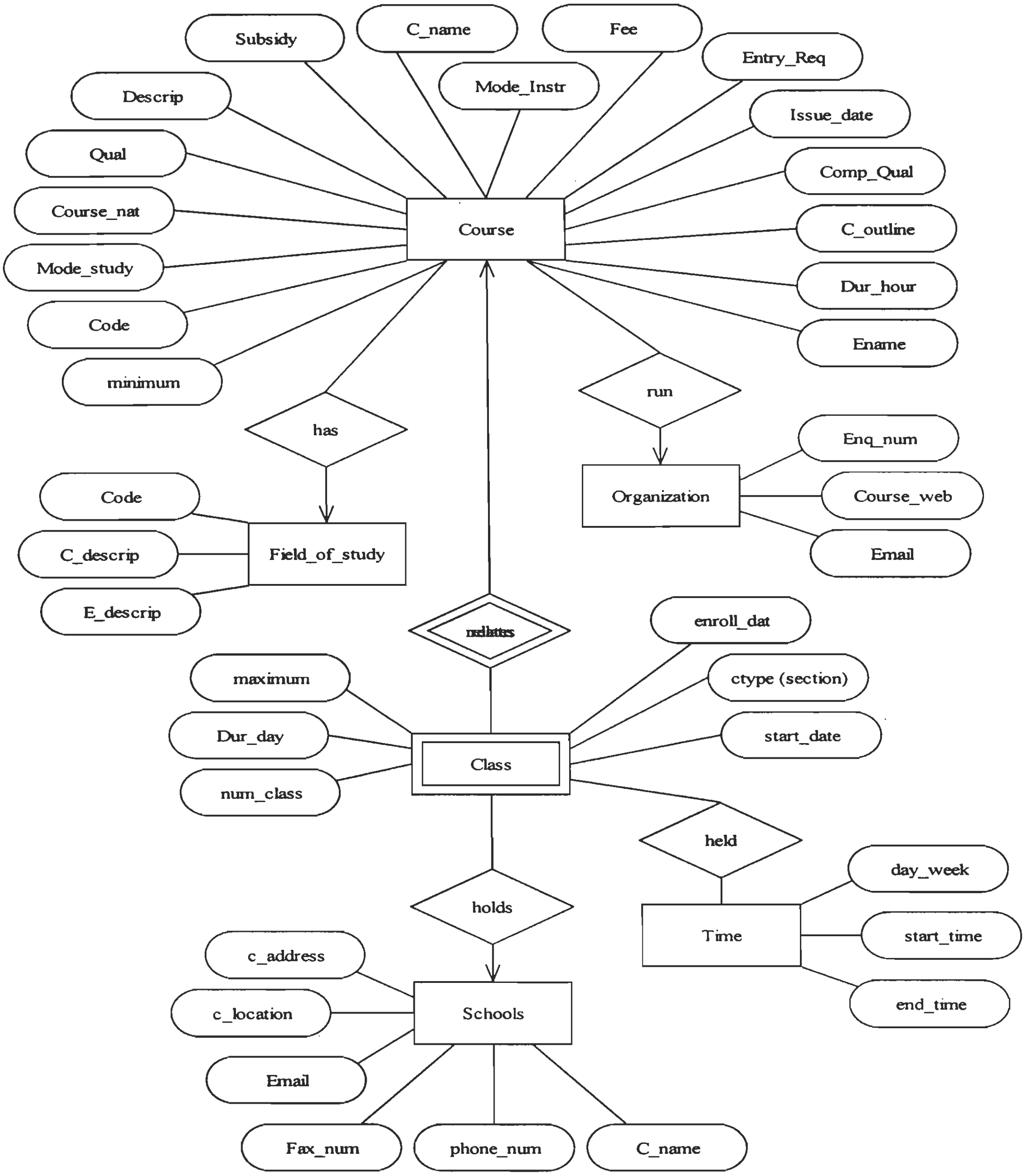
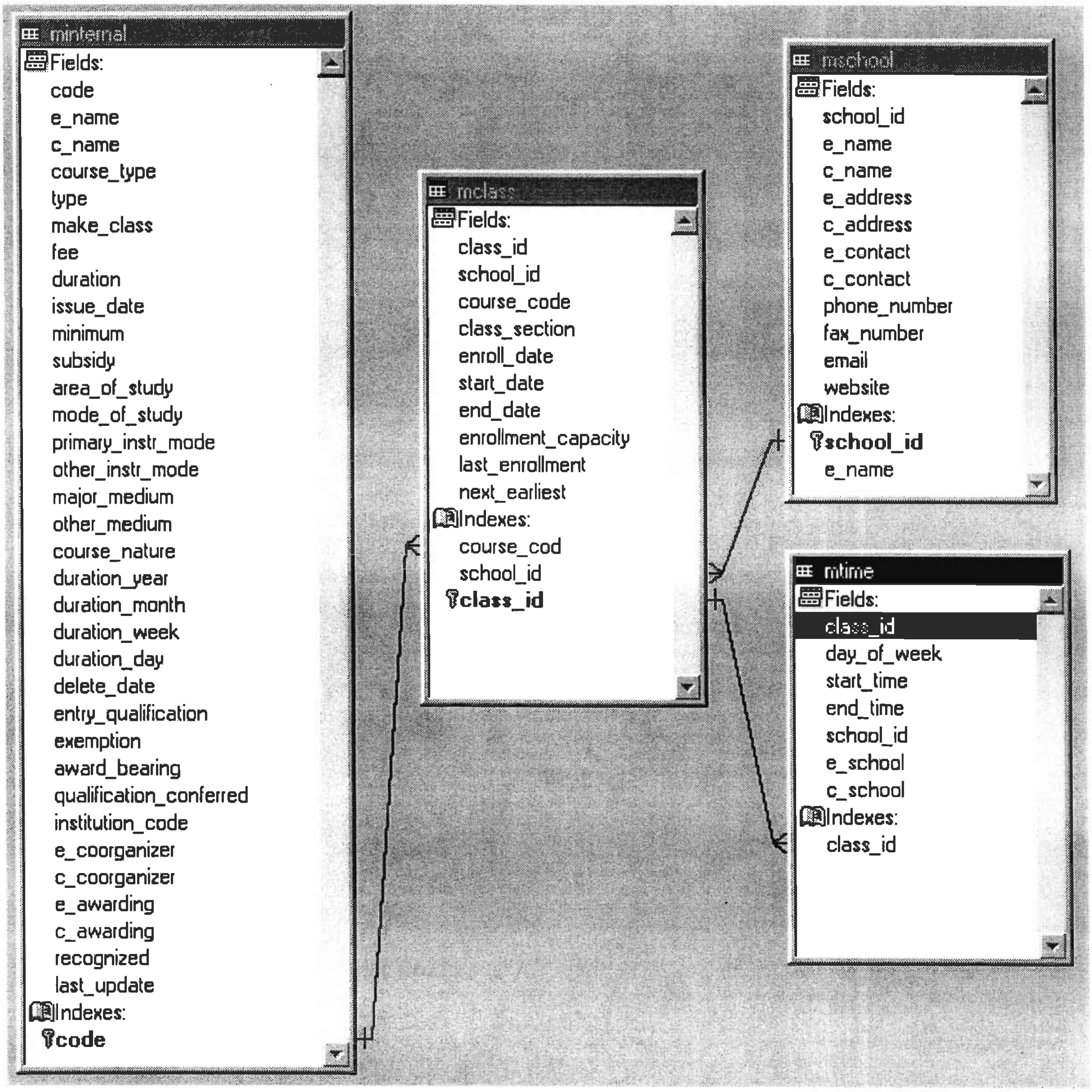


Figure 3.1 The Database's Entity Relationship model.



minternal

Fields:
code
e_name
c_name
course_type
type
make_class
fee
duration
issue_date
minimum
subsidy
area_of_study
mode_of_study
primary_instr_mode
other_instr_mode
major_medium
other_medium
course_nature
duration_year
duration_month
duration_week
duration_day
delete_date
entry_qualification
exemption
award_bearing
qualification_conferred
institution_code
e_coorganizer
c_coorganizer
e_awarding
c_awarding
recognized
last_update

Indexes:
code

mclass

Fields:
class_id
school_id
course_code
class_section
enroll_date
start_date
end_date
enrollment_capacity
last_enrollment
next_earliest

Indexes:
course_cod
school_id
class_id

mschool

Fields:
school_id
e_name
c_name
e_address
c_address
e_contact
c_contact
phone_number
fax_number
email
website

Indexes:
school_id
e_name

mtime

Fields:
class_id
day_of_week
start_time
end_time
school_id
e_school
c_school

Indexes:
class_id

Figure 3.2 The Database's structure in table format. Each table has a list of attributes and the connecting lines represent the relationships with other tables.

During the designing phase, there were many decision-making situations that had to be faced. For example, it was critical to know which fields were required and which ones were optional, what were the keys for the database entities and how did these entities relate to each other. The completion of the design was a major stepping-stone for this whole project; therefore every piece of information must be taken into consideration. After the database structure was refined and improved, the next phase was to implement the database application that involved intensive coding using Microsoft Visual FoxPro.

3.2 Implementation of the Database Application

The Implementation part of this project was very important because it could highly affect the efficiency and functionality of the application. The project's outcome depended on how well the database application was implemented and whether its functions could satisfy CAHES' requirements. The implementation of the Database application was carried out using Microsoft Visual FoxPro (VFP). Microsoft Visual FoxPro is a program with a programming language of its own. It is part of the Microsoft Object Linking and Embedding (OLE) giving it a large advantage over the usage of other programs. This means that as part of the Microsoft Visual Studio package, it has the ability to embed its files into other Microsoft programs to be used and vice-versa with complete compatibility. VFP also supports two different types of programming methods, the standard procedural programming and the new Object Oriented Programming (OOP) giving it more flexibility and power. Like most Microsoft programs, VFP is very easy to use and learn. It already contains a large library of tools and wizards for the programmers. In the United States, there are very few programmers that still use VFP; therefore it is very hard to do any research about it on the web. VFP is very popular as a database application builder in Hong Kong. Many organizations in Hong Kong have custom applications that are programmed using FoxPro. CAHES requested the application to be implemented by using VFP because programmers in Hong Kong are familiar with it and therefore if needed, specialized persons can be easily hired to maintain or upgrade the application.

Before coding each of the database application functionalities, a basic layout of the user interface was completed. The interface contained frames, dialog boxes, buttons, labels, text fields and Wizards. The design and layout were based on the interview feedbacks. Depending on the preferences of the users and the functions required, an interface layout was drawn accordingly. To ensure that the interface layout was practical and user friendly, mini demos were shown to Anthony and Catharine. These demos would access a dummy database application so that they could offer suggestions on further improvement. The goals for conducting these demos were to learn any unrevealed requirements and to gather additional ideas for the functionalities and graphical user interface (GUI) of the database application.

At this point, by going through the users' recommendations, necessary changes on the database structure and the GUI design were made. Improvements were made and each of the functionalities was then implemented and developed. After the database application was built, a complete test of each of the functionalities was necessary to ensure that the application was done properly. Tests were performed on a set of sample data to make sure that all aspects of the database application were operating correctly. The resulting bugs from the tests were fixed and a beta version of this application was released.

A test plan and the test version of the database application were given to the intended users. They were asked to go through each test on their own and to take notes on any problems encountered. Once all the reported problems had been isolated and fixed, a final version of the application was available. With the installed wizard, staff at CAHES can easily install this application onto any computer. Finally, with a perfectly working application and a manual of operation, the project team conducted necessary trainings for everyone that would be using this tool.

4. Results

CAHES can use the database application to reduce their staff's workload and to reduce human errors when entering data. The database will be used to store information on all of the courses offered by the twenty-six Adult and Higher Education Service (CAHES) schools. The users of the database application are from CAHES Academic Affairs office. This application will be installed on a computer located at Office of Academic Affairs. The application's input and output formats are both in the form of Excel files. CAHES wants the application to import Excel files as data input because their staff has already familiarized themselves with Excel and therefore data entry in Excel would be easier than entering course information using the database application's user interface. The application's output Excel files will be used in four ways: Office of Academic Affair's master copy, course catalog, CAHES's web site and to the Government's Education and Manpower Bureau web site. These output files will be in different formats according to each use's required fields.

4.1 Phase 1 (part one) – Interviews and Analysis

At the beginning of this IQP, interviews were arranged with the liaisons Catherine Li and Anthony Leung at Caritas Adult and Higher Education Services. These interviews went well and were helpful to clarify the project goals and objectives. For Ms. Li's interview, background information on CAHES and the process that they go through for course development were discussed (See Appendix B for Ms. Li's interview). Mr. Leung gave a very good description of the project's technical aspects and talked about their previous course data management system. (See Appendix C for Mr. Leung's interview and Figure 4.1.1 for the current course information management system flowchart)

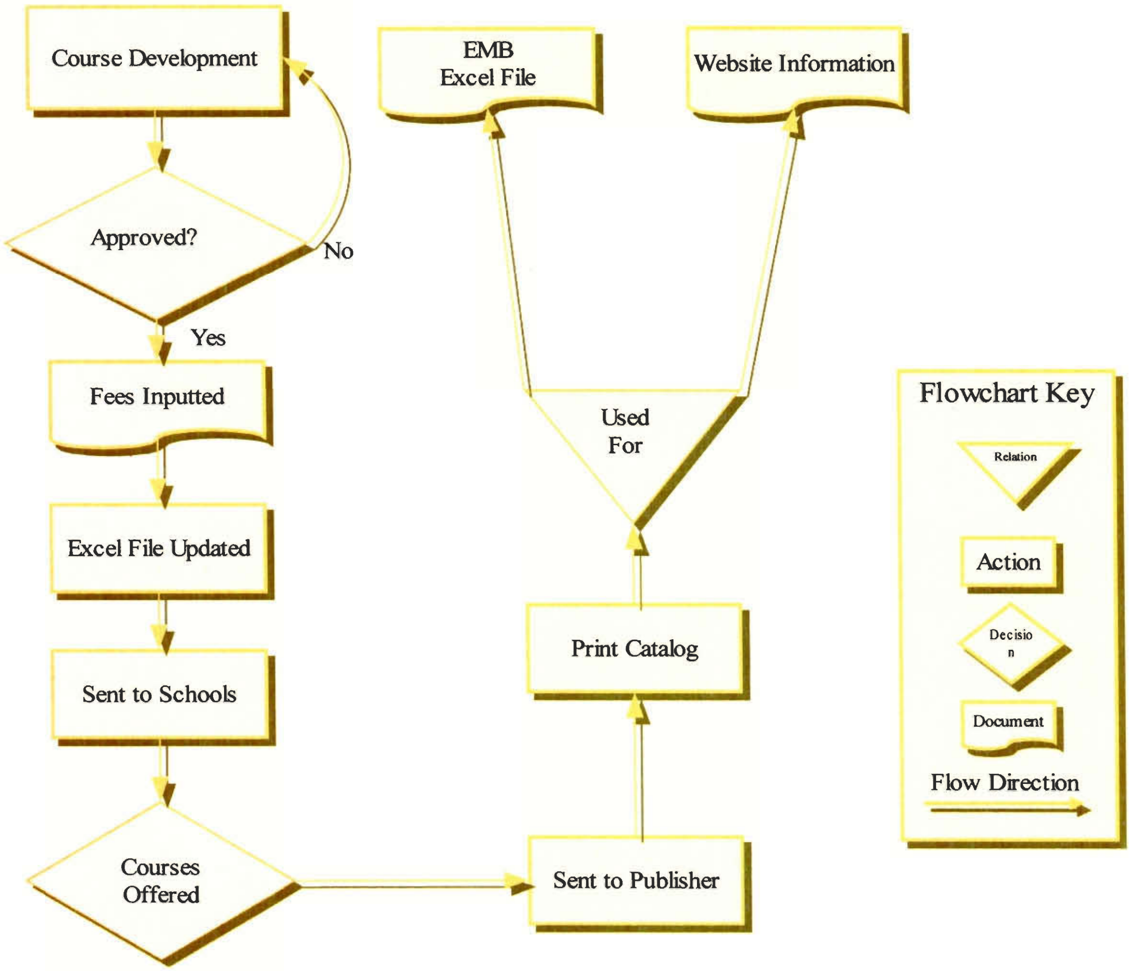


Figure 4.1.1 CAHES' current course information management flowchart.

Also discussions on the project's progress and some of the application technical issues were held during the weekly meetings with Mr. Leung and Ms. Li. After the most recent work our application development was demonstrated, they would give feedbacks and help solve problems that were encountered. With the information gathered from the interviews and meetings, a much better understanding of the current course management system aided in formulating the requirements for a new course management system.

CAHES's Office of Academic Affairs is primarily responsible for course development. The assistant organizers in the Office of Academic Affairs and the principles of each CAHES schools can propose new courses. Once a new course proposal is done, it will then be submitted to Ms. Amy Chan, Organizer of Academic Affairs Office, for approval. If approved, Rachel in the Coordination Office will input the course fee. A staff in the Academic Affairs Office, Rebecca will then update their course database which is just an Excel worksheet with the following fields: Course Code, English Name, Chinese Name, Class Type, Class Make, Fee, Duration, Issue Date and Minimum (the Minimum field is the minimum of students needed to enroll in the class before it is opened) (See Appendix D). Once the Excel file has been updated with the new courses, the file will be distributed to the schools' principals. The principals at each of the twenty-six schools then decide which courses they will offer, setup sections for multiple offerings of a single course and plan the times that each class will be held. To share their course offerings, the schools forward this information to the course catalog booklet publisher. This course catalog is for the students, parents and the public who are interested in taking a course at a Caritas School. Recently, the Hong Kong Government's Education and Manpower Bureau (EMB) began to host a website that contains a database storing all of the continuing education courses available in Hong Kong. As a result, CAHES has to submit to EMB an Excel file with information on the courses offered by all of the Caritas Schools. Since CAHES's Office of Academic Affairs does not have all the course information required by EMB (See Appendix E for an abbreviated list of EMB required fields), a staff member at the Academic Affairs Office, Kitty, has to search through the course catalog and type up all the course information into a specially formatted EMB Excel worksheet. Besides typing up information for the EMB website, there is also an employee from the Audio and Visual office who maintains the Caritas' Homepage. He also has to type up the course descriptions for all the courses for the website.

This system lacks efficiency and is prone to minor errors. Just by reviewing the course development Excel worksheet that Rebecca typed up (some of which is shown in appendix D), there are seven hundred and thirty-three records that represent the seven hundred and thirty-three different types of courses that are offered in the CAHES schools. Some schools offer the same course multiple times per season. Kitty must enter a multitude of diverse and repetitive information to the EMB's Excel worksheet and it is very easy for her to include accidental typographical errors into these Excel worksheets. That is the reason CAHES wants an application to include the ability to import Excel data and produce differently formatted outputs. Another function that CAHES wants the database application to provide is the ability to avoid unapproved courses being added to the database. As for the current system, the schools might unintentionally add some unapproved course to their list and send it to the catalog publisher. There may be fields changed by mistake, such as the course code, time, fee and etc. After the CAHES staff discovers such errors, it will be too late because the catalog will have already been published and distributed to the public and prospective students might try to enroll in the courses that are not official to the Office of Academic Affairs. To avoid this scenario, pre-formatted Excel worksheets were created to serve as a template for each school to fill in with the appropriate information for them to submit to the office of Academic Affairs. The users of the application will then import these Excel worksheets into the relational database that is located inside of the application. Before each record (information about each class) is inserted, the application will check all the fields by comparing the record with the master database's records and reject entries that have mismatched fields. The proposed course information management system will enhance the current system through the use of the database application. (See Figure 4.1.2 for the proposed course information management system flowchart)

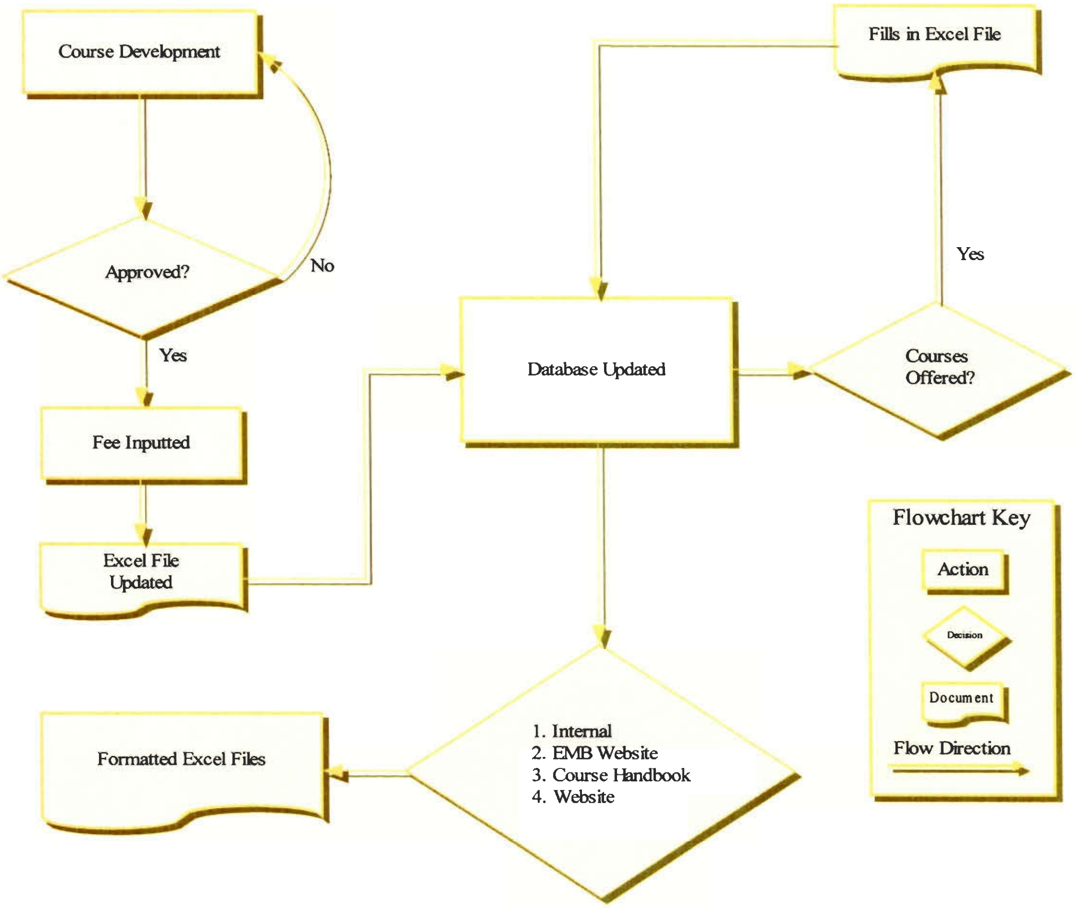


Figure 4.1.2 Proposed course information management system flowchart.

The master database will initially be filled with data concerning only officially approved courses and afterwards the class information from each school will be inserted. Our database is designed in such a way that it has the ability to store information pertaining to just courses. The first type of data that the application will import is the internal Excel worksheet that the Academic Affairs Office uses to approve its courses. The inputting of this file first acts as a filter so that when the Excel worksheets from the schools are imported, the course codes for each of the classes must match one of the course codes that are already in the database. If invalid course codes or any other errors are found, then the office will notify the school to fix the errors and resubmit the Excel files for reentry into the database. By adding information in this manner, the application will solve the existing flaws of the current system and improve the data management's efficiency.

Even the Academic Affairs Office's own internal Excel worksheet of approved courses, that is to serve as the foundation of the database, is not without errors. To test the import feature of the database application, one of Academic Affairs own Excel files was imported during demos (a sample of which can be found in Appendix D). The Excel file contained a listing of the 733 officially approved courses that CAHES offered for that particular season. The importing procedure was not quick and easy as was expected from a file from the main office. There were several errors in the Excel file that caused the import feature to fail before loading the entire Excel worksheet. The first cause of this problem was that some of the approved courses did not have a course code yet. So that when the application reached a row that did not have a course code, it interpreted the blank course code as having reached the end of the data and then appropriately terminated. The next error that the application detected was courses that had the same course code but were distinctly different, evidently a human input error. There were also inconsistencies in the duration field. At first glance this field seemed to hold only numbers that corresponded to how long the courses were in hours. The database application soon found that there were characters in this field such as "1 year" and "7 days." Treating all of the values in this field as characters temporarily solved this problem.

At other meetings, the errors found in the office's Excel file and possible solutions were discussed. The liaisons stated that there should not have been any empty course code fields and that in the future they will make sure that all of the course code fields have proper values and all course codes are unique. The repeated course code errors were confirmed to be typographical errors. Another feature was implemented to detect these errors and display a list with a description of the error so that the user can easily correct these errors. A solution to the duration data type problem was to create a new field that contained the unit of the duration and treat the numerical part of the duration as a number, such as year, month, week and day.

4.2 Phase 1 (part two) – Database Structure Design

The collected sample Excel files were examined to determine what type of data the application had to deal with. The format of the Excel worksheets was very important because the Excel files outputted by the database application had similar formats as the Excel files collected in the beginning and it would be helpful to preserve the consistency of the data. These Excel worksheets contained fields that had been filled in with information regarding different schools, courses and classes. They had fields with information regarding the CAHES schools, the classes held at these schools, information needed for the course handbook, information in the Caritas website and information needed for the Hong Kong government's continuing education website. The data that would be outputted for the different uses contained many overlapping fields and there were some fields that were left blank because they did not apply to CAHES courses. From the Excel files collected, there were obvious fields that were not required for the final output file therefore reducing the repeated fields and eliminating those that were not used would improve the organization of the data.

The data fields were extracted from all the Excel data files. These fields were then compiled onto a single piece of paper. This allowed for ease to identify repeated fields systematically and eliminated non-relevant ones. Fields that were not eliminated were separated into groups of attributes that described the same object. For example if the data field was time then it could be put into the group of attributes that described a class. The time field did not describe a school or a course, but it was obviously closely related to a class.

From the grouped data fields, relationships between the entities, which would be the objects described, could be established once the Entity Relationship model was built (See appendix F). In the process of producing the ER model, there were difficulties in relating and identifying all of the information. The data models had to be modified and improved many times to ensure that data were related correctly. Keys, attributes that could uniquely identify an entity, were assigned to each entity and the types of relationships between each entity were also determined. There are three types of relationship possible between entities: one to one, many to one and many to many. Explanation of the ER model can be found in the literature review chapter.

The completion of the ER model paralleled the construction of the Object Definition Language model (See Appendix F). This model contains one extra important point that the ER diagram does not; it contains data types. In this phase, the data type for each field would have to be determined. The data type was very important because it would later be used in the implementation of the application. The ER model allowed us to have an overview picture of the database, while the ODL gave a programmer's view in the form of classes, attributes and relationships.

The ER diagram and the ODL model were then used to design the relational database model (Appendix G). Following the method discussed in the literature review section, the two models were converted into two relational database models. The two relational database models were different because they were derived from two different sources. By determining the functional dependencies, decomposing the tables into smaller tables and eliminating tables that were just repetitions of other tables we came up with our final set of tables for the relational database model. The normalized relational model that was designed on paper was then implemented in Visual FoxPro.

The creation of the relational database in Visual FoxPro was the culmination of the data analysis and design portion of this project. The purpose of analyzing the data was to create data models; the purpose of the data models was to create a relational database model and the purpose of the relational database model was to construct a database in Visual FoxPro to hold all of the data. With the implementation of the relational database, our database application had a core that we could build around.

4.3 Phase 2 – Developing Database Application

The database application is implemented using Microsoft Visual FoxPro (See Figure 4.3.1a for application flowchart). A Visual FoxPro application includes one or more databases, a main program that sets up the application's system environment and a user interface comprised of forms, toolbars, and menus. Visual FoxPro's ability to use Object Linking and Embedding is a major advantage in gathering data for the databases needed in a FoxPro application. This feature incorporates the power and flexibility of applications such as Microsoft Excel, since the database application will be importing data from Excel files and also outputs in Excel formats as well. Visual FoxPro utilizes the Structured Query Language, a database sub-language, to allow users to retrieve filtered information from the database. "This database sub-language is becoming the standard interface to relational and non-relational DBMS."(Beynon-Davies, 1992) Visual FoxPro will definitely have all the tools to implement a well-designed database application.

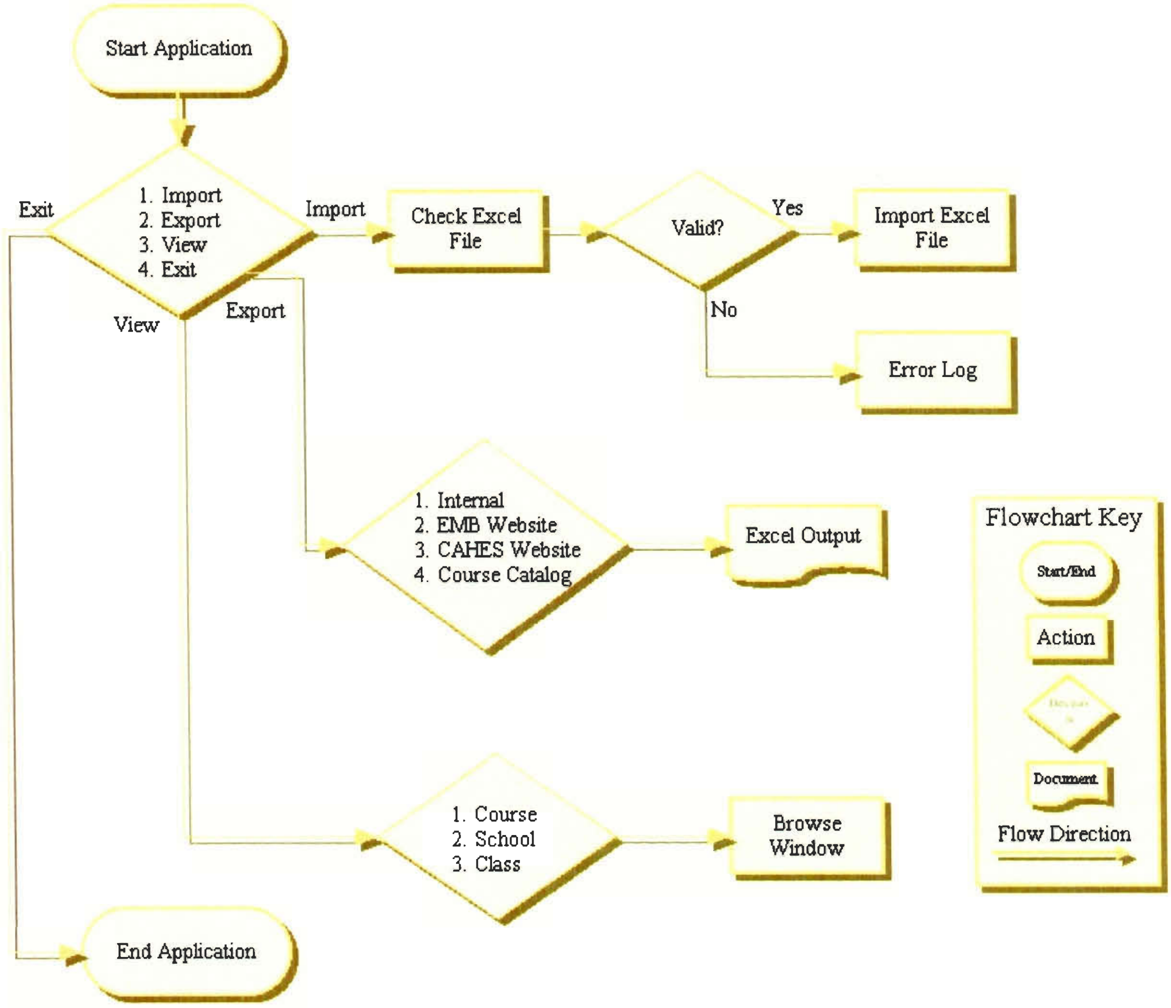


Figure 4.3.1a Course information database application flowchart.

For the developing application phase of this project, the first step was to focus on learning Visual FoxPro. During the learning stage, a few mini programs were created to perform tasks such as reading data from Excel Worksheets cell by cell, writing data to Excel Worksheets, load Excel worksheets, search and sorting through FoxPro tables and FoxPro forms for browsing the database. As a result an interface could be created to tie up all the functionalities by using the code from all the mini programs. The application was able to import data from Excel worksheets, output data to a formatted Excel templates, viewing the database by course, school or classes, modify, search and sort, add data records if necessary. These functions appear to the user in an appropriate form by the use of many different interface tools including menus, forms, navigation bars and dialogs. The application worked well in both Chinese and English Windows. The only problem appeared when run under English Windows was that the Chinese characters could not be viewed correctly.

The application started with a form containing a menu with three menu items (Figure 4.3.1b), system and database and view. Within the system tab (Figure 4.3.2), users can choose to exit the application and database tab allows user to import and export the data to Excel files (Figure 4.3.3), view tab allows users to view the database by course code, school or class (Figure 4.3.4).

Course Database Tool
System
DataBase
View

Figure 4.3.1b This figure shows the application's title, icon and the menu bar.

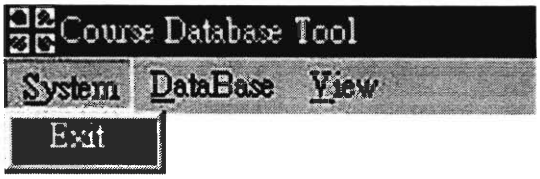


Figure 4.3.2 Exit Command

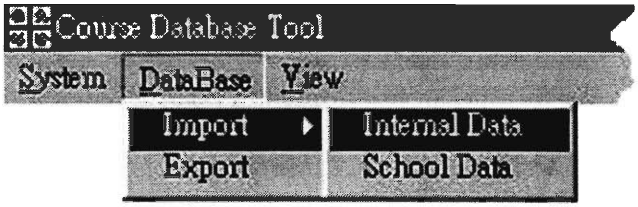


Figure 4.3.3 Commands for importing internal and school information. The export command outputs data into differently formatted Excel templates.

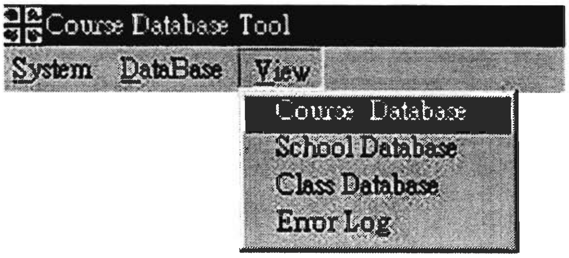
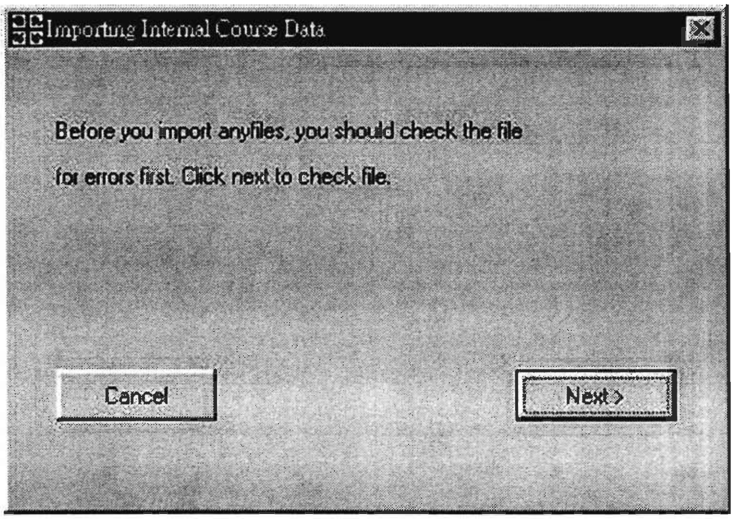


Figure 4.3.4 Commands for viewing the database by course, school or class. Users can also view the Error Log if an Excel file was rejected due to the data type errors.

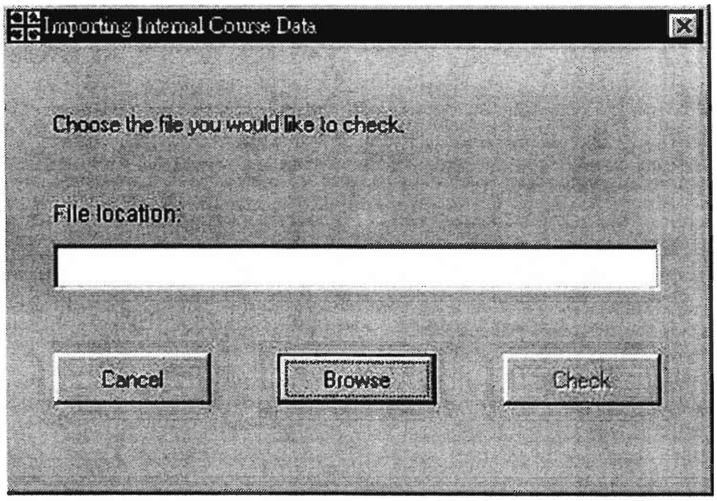
When the user clicks on Exit, the application will be terminated. If the user chooses Import under the Database tab, users can either import the internal Excel files or files from the schools. Before the data can be imported from an Excel file, the application will first check the file for any invalid data. When the import command is issued, a dialog box (figure 4.3.5) will notify the user that the application will have to go through the error checking process, users can click on Next to continue.



Before you import anyfiles, you should check the file for errors first. Click next to check file.

Figure 4.3.5 User can click on the Next button to continue.

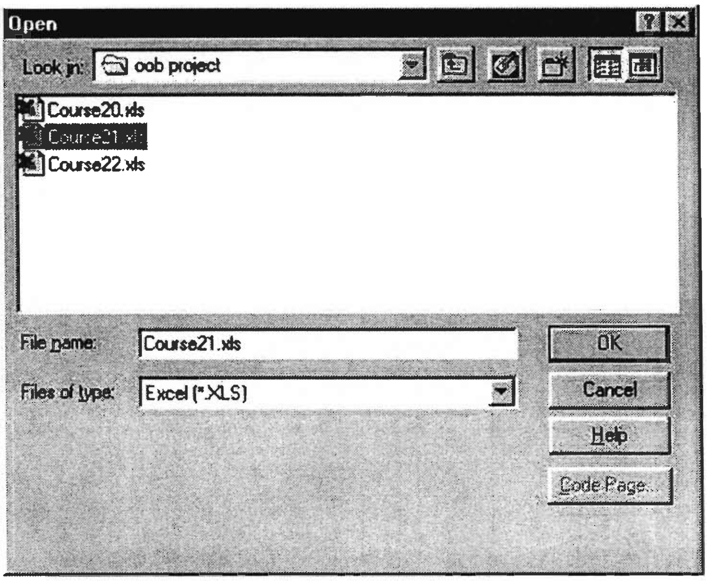
After the Next button is clicked, another dialog will prompt the users to browse for the Excel file (Figure 4.3.6). Users can click on Browse to open the browse window and choose a desired Excel file (Figure 4.3.7).



Choose the file you would like to check.

File location:

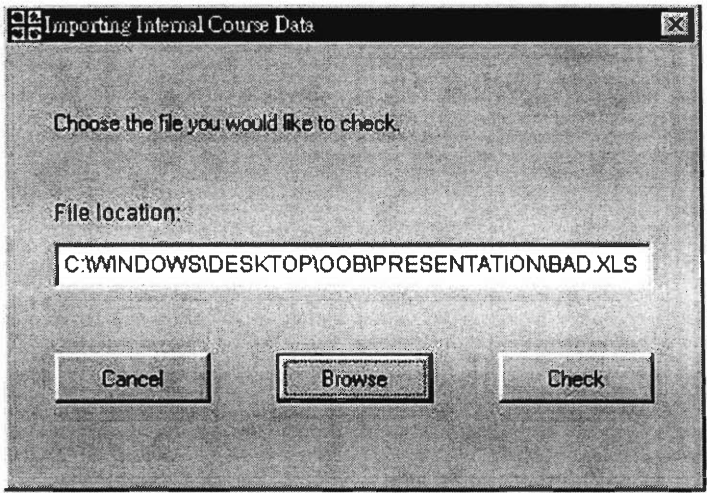
Figure 4.3.6 Prompt for user to browse for the Excel file.



File name: Course21.xls

Files of type: Excel (*.XLS)

Figure 4.3.7 Browse window for user to choose the Excel file to be imported. The user clicks on Ok button to bring back the previous dialog with the file path shown.

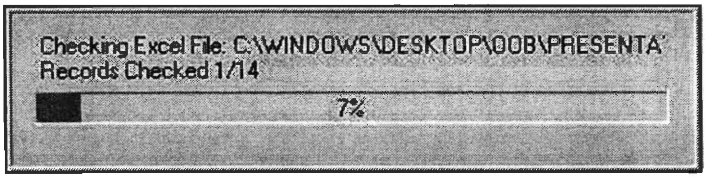


Choose the file you would like to check.

File location:

C:\WINDOWS\DESKTOP\OOB\PRESENTATION\BAD.XLS

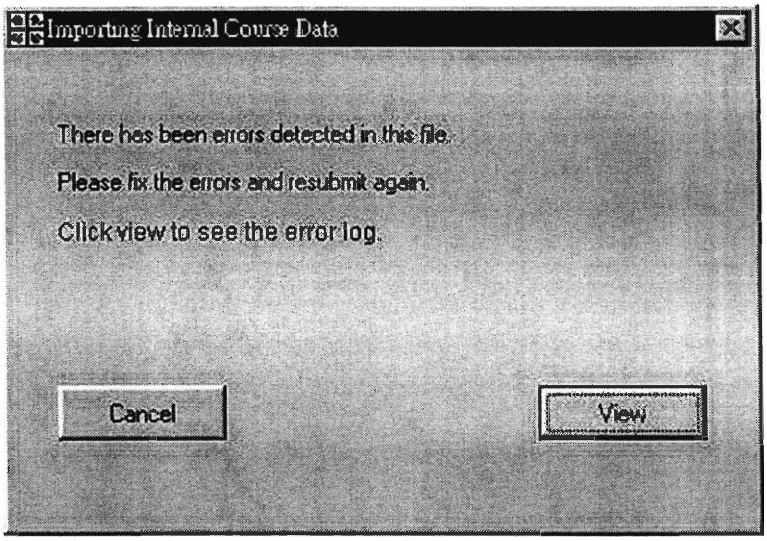
Figure 4.3.8 After a file has been chosen, the user can click on Check button to check the Excel File.



Checking Excel File: C:\WINDOWS\DESKTOP\OOB\PRESENTA
Records Checked 1/14

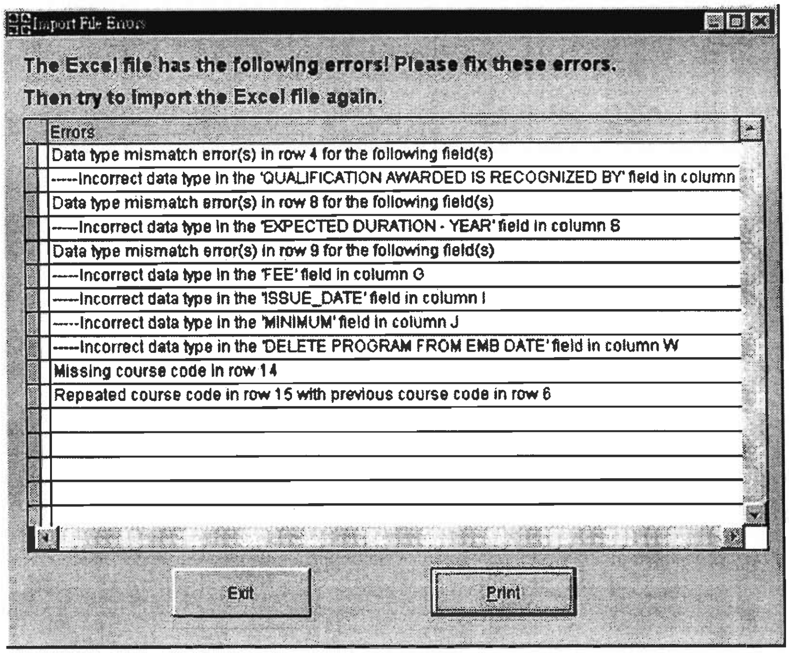
Figure 4.3.9 A status bar showing the number of records checked, after the user clicked on the Check button.

The status bar (Figure 4.3.9) disappears after all the records have been checked and another dialog will prompt the user to check if there are errors in the Excel file or not. If the Excel contains errors, a dialog prompts the user to click on the View button to see the error log (Figure 4.3.10). A browse window would come up immediately after the user clicks on the View button (Figure 4.3.11).



There has been errors detected in this file.
Please fix the errors and resubmit again.
Click view to see the error log.

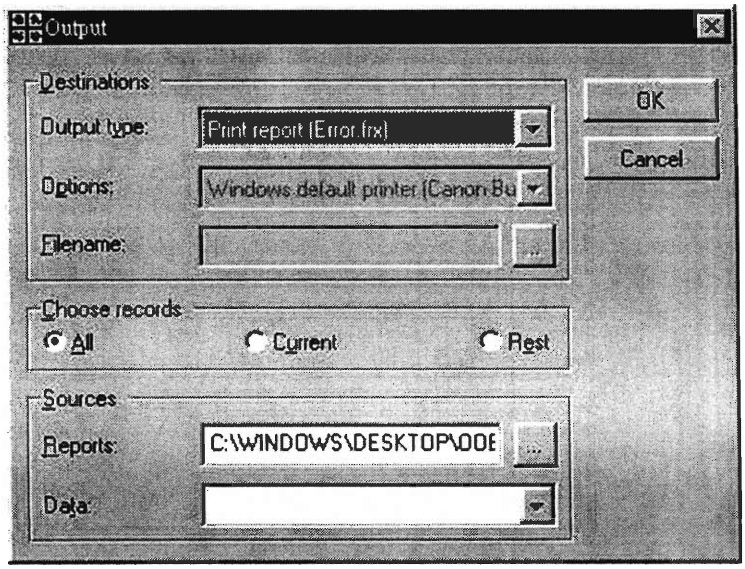
Figure 4.3.10 If the data file contains errors, this dialog will appear and prompt the user to click on the View button to see the error log.



The Excel file has the following errors! Please fix these errors.
Then try to import the Excel file again.

Errors
Data type mismatch error(s) in row 4 for the following field(s)
-----Incorrect data type in the 'QUALIFICATION AWARDED IS RECOGNIZED BY' field in column
Data type mismatch error(s) in row 8 for the following field(s)
-----Incorrect data type in the 'EXPECTED DURATION - YEAR' field in column S
Data type mismatch error(s) in row 9 for the following field(s)
-----Incorrect data type in the 'FEE' field in column G
-----Incorrect data type in the 'ISSUE_DATE' field in column I
-----Incorrect data type in the 'MINIMUM' field in column J
-----Incorrect data type in the 'DELETE PROGRAM FROM EMB DATE' field in column W
Missing course code in row 14
Repeated course code in row 15 with previous course code in row 6

Figure 4.3.11 A list of errors will be shown in a browse window after the user clicks on the View button. The user can click on the Print button to print out the list or errors. Using this list, the user can go to the Excel file and correct them.



Destinations

Output type: Print report (Error.frx)

Options: Windows default printer (Canon Bu

Filename:

Choose records

● All ○ Current ○ Rest

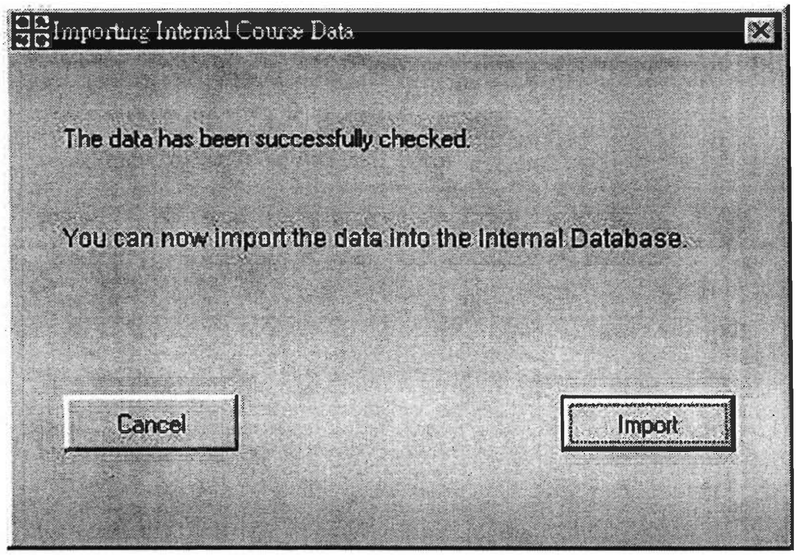
Sources

Reports: C:\WINDOWS\DESKTOP\DOE

Data:

Figure 4.3.12 This dialog appears when user chooses to print the error list.

At the error's browse window, the user can click on the Print button to bring up the Output dialog and choose to print out the error report. The user can use the print-out and fix all the known errors on the Excel file. After all the errors have been fixed, the user has to go through the importing process again which includes the checking phase to make sure all the errors have been fixed. After the checking function indicates that the Excel file contains no errors, the user can click on the Import button to start importing data into the course database (Figure 4.3.13).

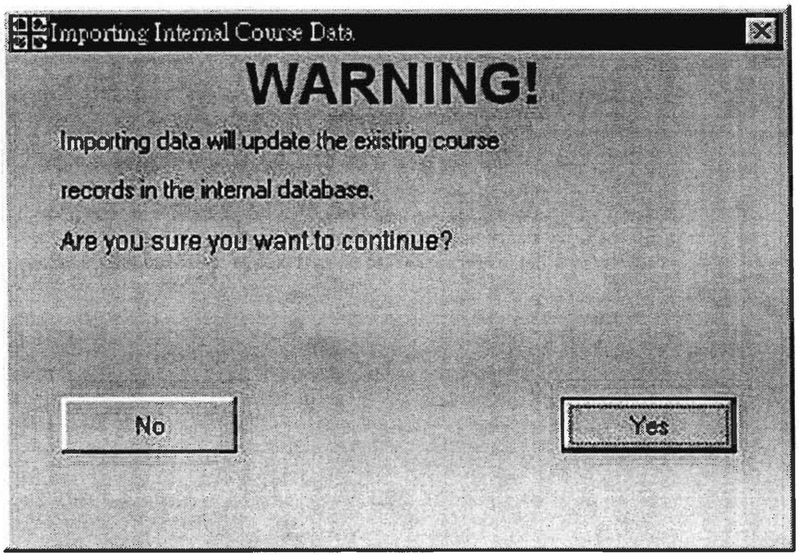


The data has been successfully checked.

You can now import the data into the Internal Database.

Figure 4.3.13 Notify the user that the Excel file has no error and to click on the Import button to start importing data.

But before the data can be imported, a dialog will warn the user that by importing data to the database will update the existing data (Figure 4.3.14).


Importing Internal Course Data

WARNING!

Importing data will update the existing course records in the internal database.

Are you sure you want to continue?

Figure 4.3.14 Warning the user that importing data will update the existing records in the database.

Users can click on the Yes button to agree importing data and update the current database. A status bar will be showing the number of records been updated to the database (Figure 4.3.15). When all the data has been imported, a dialog will prompt the user that all data has been successfully imported (Figure 4.3.16).

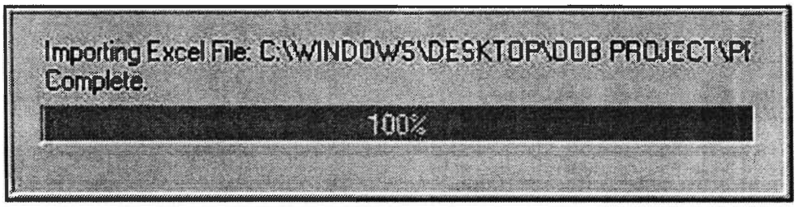
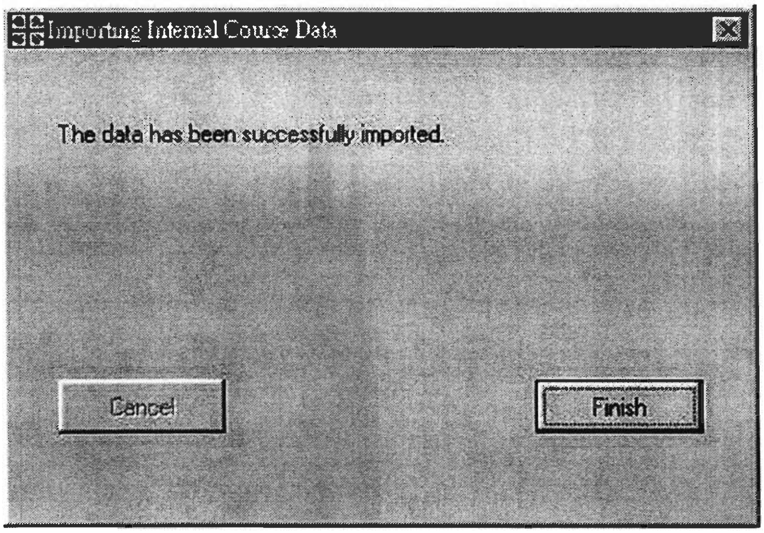

Importing Excel File: C:\WINDOWS\DESKTOP\DDB PROJECT\PI
Complete.

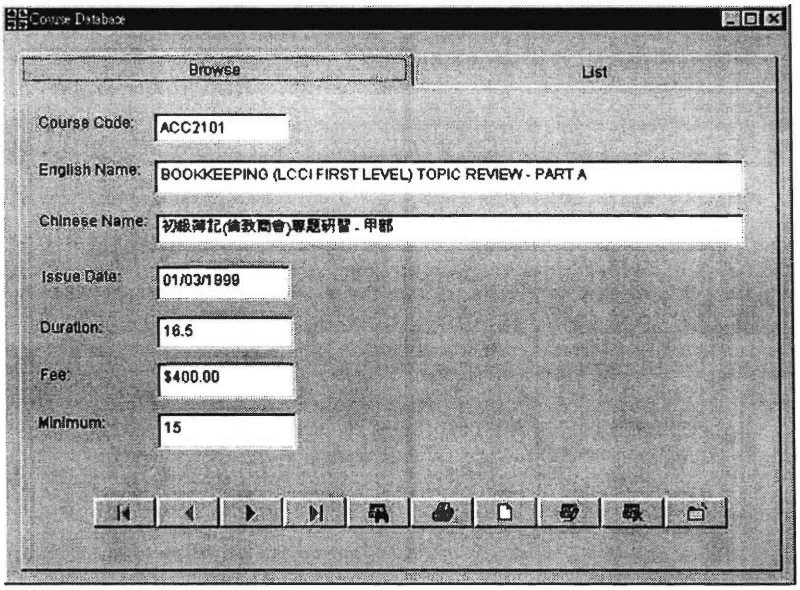
Figure 4.3.15 Status Bar show that all data has been imported



The data has been successfully imported.

Figure 4.3.16 The data has been successfully imported to the database.

The same importing procedure can be applied when importing the Excel file from the schools. After all the data files have been successfully inputted, the user can go to the View tab on the application menu to view the database by course, school, or class. The course browser lists all the courses that have been approved by the internal office (Figure 4.3.17). There are two views to this browser. The user can use the first view to browse through each course individually, for each course the browser shows the following fields: course code, course English and Chinese name, course issue date (date when the course was approved), duration (hours), course fee and the minimum number of students registered before a class can officially open. This browser has a navigation bar that has the functions to navigate, search, prints, edit, new and delete the database records. Also, there is a button for exiting this form. This browser provides an alternative way of viewing the data that contains that whole list of the courses in a grid format in an Excel worksheet (Figure 4.3.19). The user can use the search function to narrow down the records to be displayed on this browser; the search button would activate the search dialog where the user can sent the criteria for the search (Figure 4.3.18).



Course Code: ACC2101

English Name: BOOKKEEPING (LCCI FIRST LEVEL) TOPIC REVIEW - PART A

Chinese Name: 初級簿記(倫敦商會)專題研習 - 甲部

Issue Date: 01/03/1999

Duration: 16.5

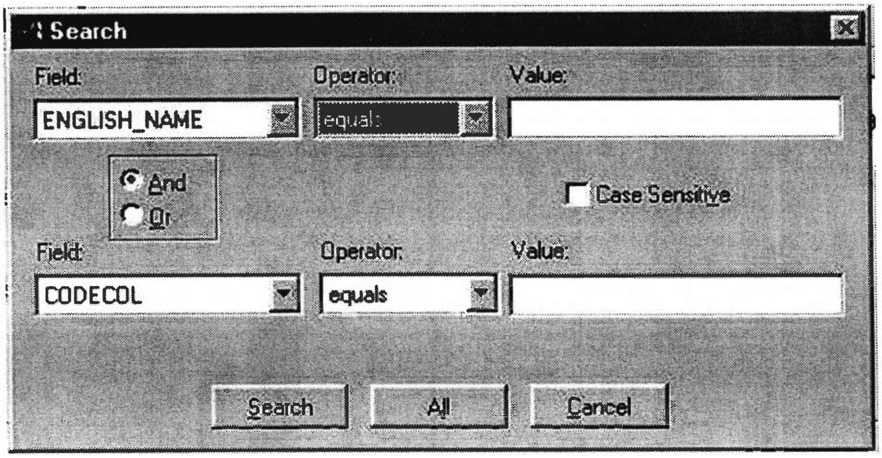
Fee: $400.00

Minimum: 15

Figure 4.3.17 Database browser window by Courses. Navigation bar buttons (from left to right):

1. *Go back to the very first record of this database.*
2. *Go back to the previous record.*
3. *Go to the next record.*
4. *Go the last record of this database.*
5. *Brings up the search and sort form.*
6. *Prints the records.*
7. *Adds new record*
8. *Edits the record*
9. *Deletes the record*

10: Exit the Database Viewer



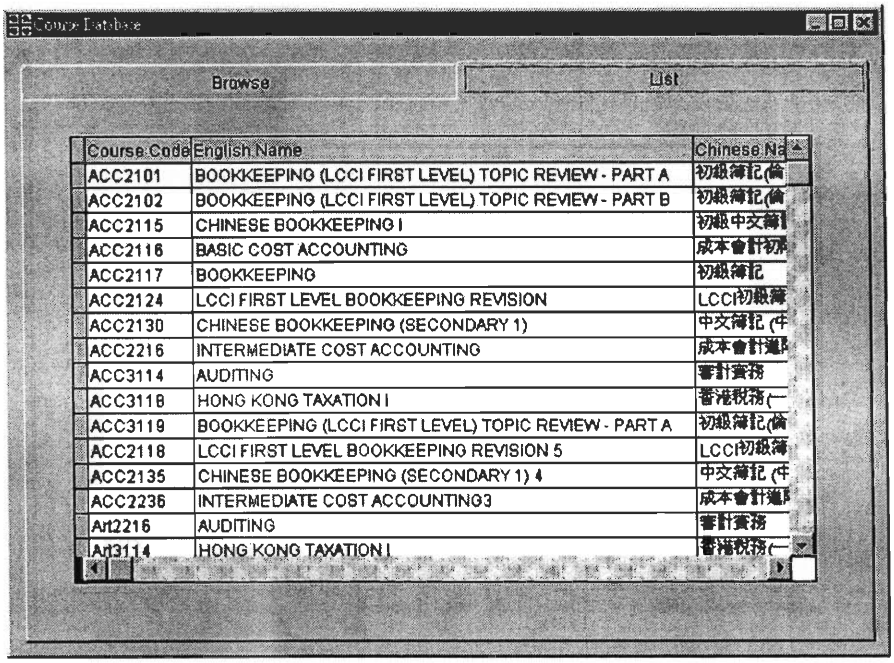
Field: ENGLISH_NAME Operator: equals Value:

● And
○ Or

□ Case Sensitive

Field: CODECOL Operator: equals Value:

Figure 4.3.18 Search Dialog.

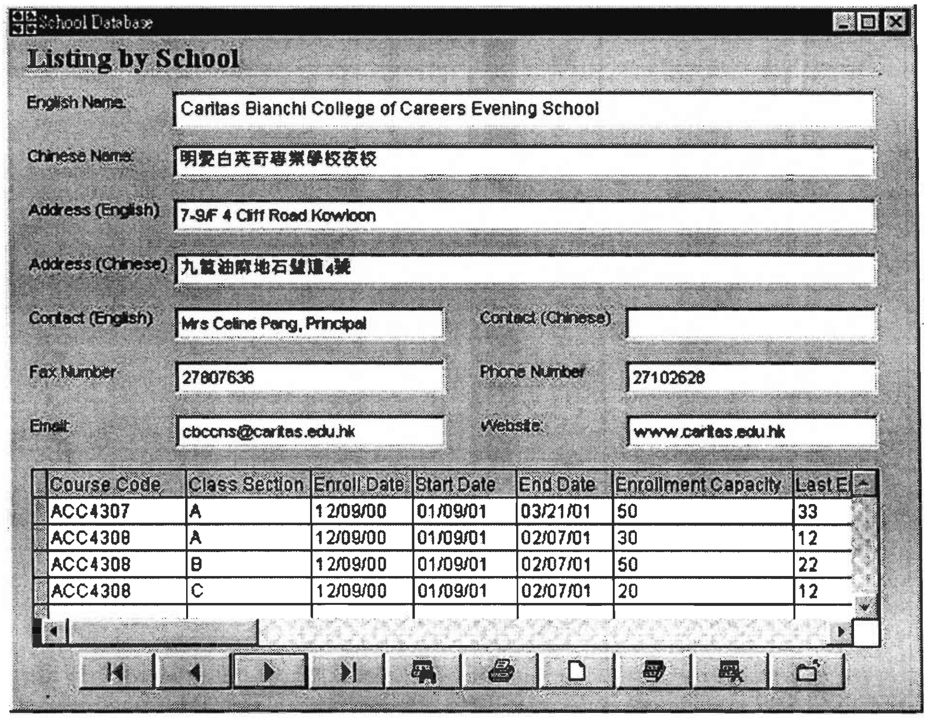


Course Code	English Name	Chinese Na
ACC2101	BOOKKEEPING (LCCI FIRST LEVEL) TOPIC REVIEW - PART A	初級簿記(倫
ACC2102	BOOKKEEPING (LCCI FIRST LEVEL) TOPIC REVIEW - PART B	初級簿記(倫
ACC2115	CHINESE BOOKKEEPING I	初級中文簿
ACC2116	BASIC COST ACCOUNTING	成本會計初
ACC2117	BOOKKEEPING	初級簿記
ACC2124	LCCI FIRST LEVEL BOOKKEEPING REVISION	LCCI初級簿
ACC2130	CHINESE BOOKKEEPING (SECONDARY 1)	中文簿記 (中
ACC2216	INTERMEDIATE COST ACCOUNTING	成本會計進
ACC3114	AUDITING	審計實務
ACC3118	HONG KONG TAXATION I	香港稅務(一
ACC3119	BOOKKEEPING (LCCI FIRST LEVEL) TOPIC REVIEW - PART A	初級簿記(倫
ACC2118	LCCI FIRST LEVEL BOOKKEEPING REVISION 5	LCCI初級簿
ACC2135	CHINESE BOOKKEEPING (SECONDARY 1) 4	中文簿記 (中
ACC2236	INTERMEDIATE COST ACCOUNTING3	成本會計進
Art2216	AUDITING	審計實務
Art3114	HONG KONG TAXATION I	香港稅務(一

Figure 4.3.19 A list of courses from the database.

The user can also view these records by school (Figure 4.3.20), in this browser the user can see what classes the schools are offering. It contains the following fields: school's English and Chinese name, school's English and Chinese address, contact person's

English and Chinese name, fax number, phone number, email address and school web site. Also, there is a grid table displaying all the classes offered in this school with course code, class section, enroll date, start date, end date, enrollment capacity, number of students enrolled last season and the date when the class will be offered again. This browser has the same navigation bar at the bottom of this window.



English Name: Caritas Bianchi College of Careers Evening School

Chinese Name: 明愛白英奇專業學校夜校

Address (English) 7-9/F 4 Cliff Road Kowloon

Address (Chinese) 九龍油麻地石壁道4號

Contact (English) Mrs Celine Pang, Principal

Contact (Chinese)

Fax Number 27807636

Phone Number 27102628

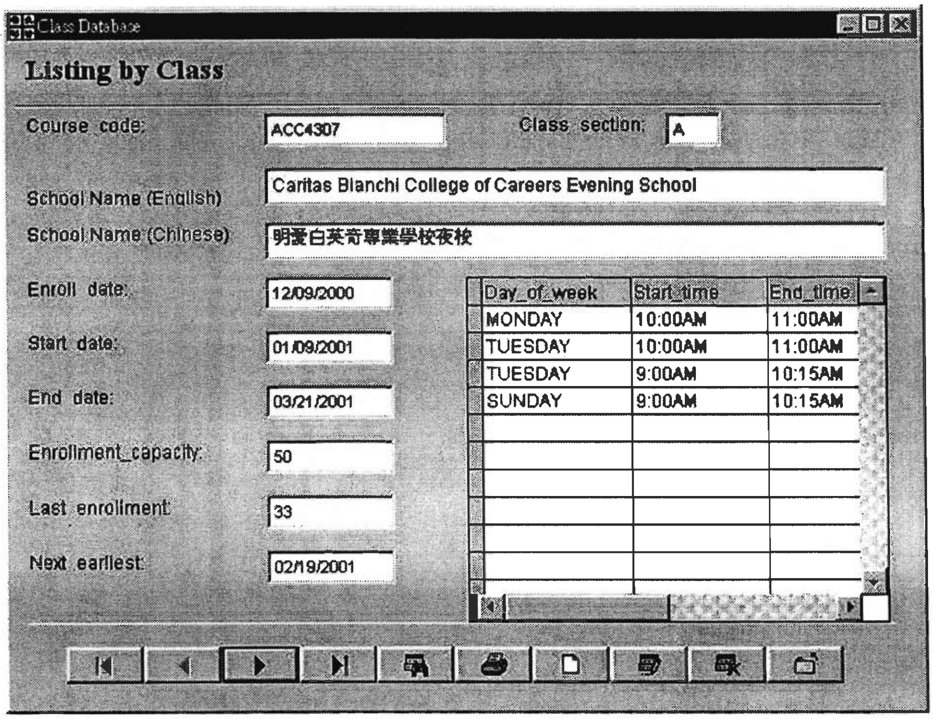
Email: cbccns@caritas.edu.hk

Website: www.caritas.edu.hk

Course Code	Class Section	Enroll Date	Start Date	End Date	Enrollment Capacity	Last E
ACC4307	A	12/09/00	01/09/01	03/21/01	50	33
ACC4308	A	12/09/00	01/09/01	02/07/01	30	12
ACC4308	B	12/09/00	01/09/01	02/07/01	50	22
ACC4308	C	12/09/00	01/09/01	02/07/01	20	12

Figure 4.3.20 Database browser by schools.

For the class database browser (Figure 4.3.21), it displays the course code, section, school's English name, school's Chinese name, enroll date, start date, end date, enrollment number, number of students enrolled last season and the date when this class will be offered. In addition to these information, there is a table showing the day and time when this class will meet. Just like the other two browsers, it also has the navigation bar.



Listing by Class

Course code: ACC4307

Class section: A

School Name (English) Caritas Bianchi College of Careers Evening School

School Name (Chinese) 明愛白英奇專業學校夜校

Enroll date: 12/09/2000

Start date: 01/09/2001

End date: 03/21/2001

Enrollment_capacity: 50

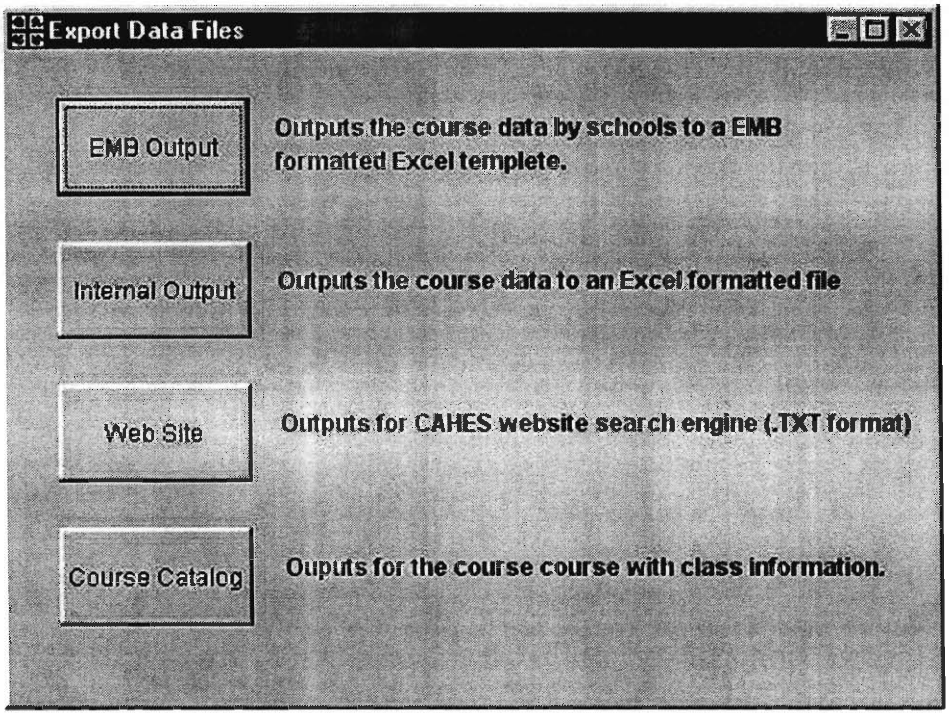
Last enrollment: 33

Next earliest: 02/19/2001

Day_of_week	Start_time	End_time
MONDAY	10:00AM	11:00AM
TUESDAY	10:00AM	11:00AM
TUESDAY	9:00AM	10:15AM
SUNDAY	9:00AM	10:15AM

Figure 4.3.21 Database browser by classes.

After all the data files have been imported into the database, users can export data to differently formatted Excel formats for various purposes. From the export wizard (Figure 4.2.22), users can choose what formatted data is to be exported.



Outputs the course data by schools to a EMB formatted Excel templete.

Outputs the course data to an Excel formatted file

Outputs for CAHES website search engine (.TXT format)

Ouputs for the course course with class information.

Figure4.2.22 Export Wizard

There are four output formats for users to choose from. The EMB Output will output a format for the use of the EMB website. This format is the most detailed output consisting of the most fields. The second output, the Internal Output, has the fields most frequently used by the Office of Academic Affairs. All the courses that Caritas has offered and will offer during the current season will be listed. The Web Site Output provides a file formatted for the Caritas web site's search engine. Lastly, it also provides a Course Catalog output; this output contains all the classes of the current term and will be used for the production of the course catalog. Depending on what type of output format the users have chosen, different files can be generated. These files will appear in a directory that is specified by the user. Now that the application has been equipped with all the required functions, a demonstration and initial training session were presented to Mrs. Catherine

Li, Mr. Anthony Leung and other potential users. During this demonstration, basic functionalities were shown and presented from a set of test data files that were created beforehand. The application successfully detected errors in the test files that contained errors which were deliberately put in (See Appendix I User Manual – Import Chapter for a list of known errors). After explaining to the user that a list of the errors could be viewed and printed out for ease of error correction, error free data files were imported into the database. These files were successfully imported and with the database filled with data, the database browsers were now applicable. Detailed explaining of the functionalities and the information provided on each different data browser allowed the users to understand the application and therefore resulting in suggestions and comments concerning improvements in the application and its functionalities which could significantly improve the efficiency of their work while using the database application. These suggestions were taken into serious consideration. Due to the fact that the users were not comprehensive on programming and what programming could accomplish, some of their suggestions were nearly impossible to implement in the amount of time that was available, but other comments could be accomplished in matters of just a few lines of code. The changes that were possible in terms of implementation and that would result in an improvement, (meaning that the tradeoff was worth the changes) for the application were implemented. The suggestions that were not possible were explained to the liaisons. There were also errors that were not realized before and these errors were also fixed.

The following is a list of suggested implementation improvements:

1. After sorting the database, formatted printouts can be produced.
2. The ability to print out the errors detected from import files.

3. Status bar showing percentage of records checked or imported.
4. Setup wizard allowed user to choose default directory for installation.
5. Welcome screen after the application is activated.
6. Application runs in maximized mode.
7. Application can be run on 800X600 desktop settings.
8. Exports text file (.txt) for CAHES website instead of an Excel file.
9. The ability to check if the correct type of data file is being imported into the respective database. (e.g. an internal file is being imported into the school database.)

A list of errors revealed:

1. Status bar calculations were incorrect.
2. Left out a required data field.
3. EMB export function operated improperly.
4. Class and Time tables in the database were not linked correctly for the Class Browser.
5. Internal Output format needed to be revised.
6. Print function outputted data in a list format instead of report format.

Considering the feedbacks and new ideas the application was improved and the errors were fixed. After the application's final version was released, the user manual (Appendix I) was published and additional training sessions were held at the Office of Academic Affairs.

5. Analysis

5.1 Breakdown of the Obstacles and Solutions

In an attempt to solve a large problem, smaller, unforeseen problems slowly appeared in the process. In order to overcome them, decisions were made after the analysis of the situation. These decisions were based upon what was best for the outcome of the project. The following sections will discuss some problematic occurrences during the design and implementation of the database application and decisions made to resolve them as well as other potential solutions.

A major distinction between an advanced computer user and a daily computer user is their confidence in using a computer. Daily users tend to be very cautious of pressing any buttons that are uncertain to them. It seems that they are afraid that they might break or delete something in the computer. Advanced users are more confident of their abilities and are more willing to explore unfamiliar parts of the computer. The users of the database application at the Office of Academic Affairs are confident in regard to the applications that they use regularly but are hesitant to try out new applications. Therefore adaptation to new software was a concern of the liaisons. Historically, there was a School Utility Program that CAHES provided for the schools to use. But due to the fact that many of the potential users were not proficient in computers and needed much more training than was provided to use the program properly, most of them were too afraid to use this new software and presently there are only a few users of this application.

To lower the amount of uncertainties that users of the database application might encounter, commands were made consistent, easily understandable and self-explanatory. The program was made with a step-by-step interface to do common tasks such as importing and exporting data. The future database application users are currently entering the course information in the form of Microsoft Excel worksheets. The ability of the application to import and export data in Excel formats allows the users to continue entering data in this manner. This functionality eliminates the amount of adapting the users will have to face, although the trade off is a less efficient application due to the loading time of the Microsoft Excel worksheets. If the user added courses within our program, then invalid course information would be recognized instantly after entry instead of having to look through the error report and then going to the Microsoft Excel worksheet to correct the errors. The ability to add, edit and delete records has been implemented into the database application to supply the increased efficiency if CAHES desires it. The staff of CAHES can begin by importing and exporting Microsoft Excel worksheets with our application. After they have been familiarized with the application, they can then use the data entry functions if they wish.

In any database, there are always many issues concerning security, although the security issues of this project are slightly different from those of a typical database. Generally database security is concerned about users who have control and access of certain data, however the liaisons are more concerned about security against invalid data. These two forms of security are actually the same because the solution to preventing entry of invalid data is to limit access to the database. CAHES does not want the schools to have the ability to enter unapproved courses or any faulty information in the files that are sent to the publishing company. This problem is solved by making each school send its course offerings to the Office of Academic Affairs where the data will be imported into the database application. Since the schools do not directly enter data into our application and instead the Office of Academic Affairs import an Excel file into it, the ability to check for invalid information on an Excel file has been implemented. There has been discussion of higher security issues regarding external access and a web-based database server. A web-based database will solve all of CAHES Office of Academic Affairs problems regarding invalid information with the most efficiency. Each school would go to the web and upload a file of all the courses they want to offer, the times for each course and other related information. The database server could instantly reject any file containing invalid information or instantly update the database if the checking of the file is successful. This web-based database would eliminate the need to pass Excel worksheets between the Office of Academic Affairs and each CAHES schools and the need to manually import Excel file at the Office of Academic Affairs. However, this solution could not be implemented due to security issues with the Caritas server. Presently they are running an older server with little security options. If such a database were to be put onto the Internet, hackers could easily get access to the database. The decision was made to keep the database on a local machine.

There have also been discussions of problems that might occur after the completion of this project. One possible problem is the required changing of the outputted Excel files. Currently the only solution to this problem is to change part of the source code of this database application. Anyone with an understanding of Microsoft Visual FoxPro has the ability to perform this modification. There are only minor additions to the source code, approximately four lines of code per extra field, and the Visual FoxPro database as well as modifications in the forms (a visual programming term for what appears as a window on the computer screen) to show the new data fields. One feature that will aid this process is to put many comments in the areas of code that need to be changed if such a problem should occur. The comments could guide the personnel in what needs to be added and modified. However, this process still requires a knowledgeable person in Visual FoxPro and recompilation of the entire application from the new source code. The implementation of the ability to choose which fields to output to Excel instead of having hard coded fields and to add personalized fields would be a solution to such a problem. This would require implementation of tools that could manipulate the FoxPro database and forms. Such a task, if accomplishable, seems to be out of the scope of the seven-week term as well as the present understanding of Visual FoxPro.

A user-friendly interface is very important in any application. Often powerful software without good interface designs are not utilized or used not to its full potential. The interface design is even more important when the target users are beginning computer users. The interface of the database application begins with a friendly welcome screen. This screen does not prompt the user for information instantly and the program does not start with many popup windows that could potentially scare the user. When users want to import or export data, a step-by-step prompt window guides them through only requesting one Excel file at a time. This interface is consistent with a Windows operating system installed wizard so many users will be familiar with it already. The database application windows used to view information and data all contain a self-explanatory navigation bar to browse through the information. This navigation bar resembles the buttons on many other items that are frequently used such as a compact disc player. Each button on this navigation bar will display its function when the mouse is placed over them in case the icons are not understood. Since the primary purpose of the database application is to eliminate invalid data, most of the implementation deals with the FoxPro relational database and functions used to modify the database, both of which are internal activities hidden from the user and not a tool that is visible. Therefore the interface can be kept simple for the ease of the users.

Currently CAHES still keeps an Excel file as a medium to transport course information due to the inability to put the database on a server. It is not such a burden to keep the file but the transport of the file between CAHES and the schools is not preferred. CAHES does not want to have a member of their staff fix the invalid data that the schools submit to them. Although the database application specifies the location of the error, stating the row and column of the error in the Excel file as well as the field name of this column, making it easier to correct invalid data, it should be the schools' responsibility to submit correct data. One solution that CAHES considered to this problem is to have the incorrect Excel file sent back to the schools for correction. However, this will lead to excessive transportation of the Excel files, which is an inconvenience and could lead to confusion among which Excel files have errors and which Excel files have been corrected. This process may also be time consuming. The CAHES staff members are very busy and the response to an error message could take a week or longer, potentially causing a delay in the publishing of the course catalog. After an analysis of the problem, a system to lower the amount of errors of the files submitted by the schools was proposed to CAHES. This new system would use the checking algorithm in the database application to be made into a minor application by itself. This checking program can be distributed to each individual school. The schools would be asked to use this checking program to check their class information file before submission. If errors are reported, they are to fix the errors before submitting the file. This is not a solution to the transporting Excel file problem because there is one error, 'invalid course code,' which cannot be recognized by the checking program due to the absence of the master database that contains a list of officially approved courses. However all other errors will be eliminated, this can significantly lower the amount of time the Office of Academic Affairs will need to use to fix the error on the class information files. This checking program is a very worthy addition because it is a very useful and easy to implement. Most of the checking program has already been implemented and it just needs to be extracted from the database program. A simple interface can be designed for it containing very little options so that the schools will have no problem utilizing it.

Our project will enhance the current system by eliminating redundancy and protecting the data. In analyzing the current data management system we found flaws that could be improved. As previously mentioned the data was not secured from invalid data, and it required time-consuming data entry. These problems were solved with the design of a database application that checks files for errors with the ability to output many different Excel formats. Other problems such as adapting users to a new application and the designing of a good user interface arose during the process of designing the application. These problems required analysis of the situation and the tradeoffs. After much consideration, decisions were made to best satisfy CAHES. On the completion of the database application, the new course management system with the inclusion of this application was then analyzed. Any remaining deficiencies of this new system were closely looked at to make further improvements if possible. At this point the liaisons realized that a member of the staff would be needed to correct errors found in the Excel files. This need resulted in the implementation of the checker program. Analyzing data and problems is a very important step in solving problems with the best possible solution. Other potential solutions to the problems were looked at to ensure that the decision made were the best choices for the satisfaction of the liaisons. The project directly eliminated the two major concerns of the liaisons therefore improving the current system. It does not provide the most efficient solution due to security issues, time and scope limitations exceeding this project. There are still changes and other possible upgrades with the course management system, some of these will be mentioned later in the recommendation section. The goal of the project has been successfully met with the implementation of a fully functional database application.

5.2 Analysis and Consideration of Human Response

Upon completion of this Interactive Qualifying Project there were many obstacles that were overcome, many decisions that were made to best satisfy CAHES and much information was gathered to strengthen the understanding of the situation so that the best choices were offered to CAHES. Interaction with our liaisons provided information for the project to best integrate into the CAHES' organization.

The interviews with Mrs. Catherine Lee and Mr. Anthony Leung supplied information on both the technical and organizational needs of CAHES. This information greatly helped the understanding of what CAHES wanted to accomplish and their priorities. Understanding of CAHES' previous course information management system, what they wanted in the new system, what they believed needed to be improved and which parts of their current system they would like to keep afforded the project the ability to improve CAHES' current system the way they wanted it to be improved. The primary deficiency of CAHES' current course management system was that every time CAHES wanted to distribute their information, a manual entry of the information was needed. They could not distribute a single, all-embracing file to the different users of the course information because different information was required by each of the different users.

Weekly meetings with the liaisons revealed that there have been problems with the course information handed to the course catalog publisher by the schools. Evidently, the schools sometimes compose their own courses or course information for their own convenience. The Office of Academic Affairs would like to stop this unregulated activity through the use of the database application. They would like all the information about the courses to be decided by their office and the times and sections of each class to be decided by the educational sites. The checking function in the database application was implemented for this purpose. The proposed new course management system required each school to hand in the class information in a certain Excel format. This format could be imported into the database. Before the data is imported, the database application would check all the data in the class data Excel file. Any inconsistency with the master database of the Office of Academic Affairs will be exposed and the file will be rejected from importation for corrections to be made.

As the implementation of the project was underway, the liaisons also brought up that the method of data entry was a factor to their staff. The future users of the application prefer to input data from Microsoft Excel worksheets because they are more familiar with using this application. Historically, there have been problems with these users' ability to adapt to new software. The application was then designed to accept Excel worksheets as input. A special Excel template containing all the information that the different outputs required was designed for the users to enter information, the application is able to read and extract data from this special template which would be the Excel form. However, the functions to input using the database application itself is kept in case CAHES would like to input directly from the application's user interface in the future. Entry of data from the application is more favorable in that it will save time to import Excel data and it will also be able to check for error while data is being entered. If the entry method were via importing Excel files, errors in the data would be displayed in a report processed by the application. The users would then have to go look for the errors in the Excel worksheet according to the row and column number or name listed in the report.

In consideration of the users preferences for the interface of an application, questions were asked to the liaison concerning what was preferable. It was decided that a simple interface utilizing a wizard would be most favorable. Very few choices are given to the user on the interface so that the design would not seem too complicated. The importation and exportation processes were done in the form of a wizard. These wizards will guide the user step-by-step through the functionalities of the database application, eliminating any uncertainties the user might have. The application does not require the users to supply a great deal of information at one time, avoiding the possibility of overwhelming them.

The training of the application to the users started with a demonstration of some of the functions that the application provided. Detailed explanation of the application to the users enabled them to understand the application. This hands-on experience sparked ideas of what would be an even more ideal database application. They suggested the ability to print out the search results in a formatted report. This will save them time, not having to search through all the data and not having to look at information from their monitors. The function to print search results was then implemented.

The users soon understood the use of the application and how to apply it to the new course management system. This new system and how to use the database application was also explained to Mr. Anthony Leung who is more technically informed so that he is able to help the users in the future if problems beyond the training arise. The user manual was then written to aid the users in their use of the application.

A primary goal of this project was to consider the needs of the users who would be affected by this project. Since the deficiency that CAHES wanted to eliminate from the current system was the redundancy in the need to reenter the data manually by the user and to error check data entered into their master database, these were the issues that were primarily considered. Some issues that could be improved were specifically not changed in consideration of the users' computer ability and their preference for their work environment. Further advances to the application, training and the write up of the user manual were to aid the user in the usage of the application.

The development of this project entailed more than designing and implementing of a database structure and application, consideration for CAHES and the users' was necessary. To fulfill all the needs of CAHES, it is critical to apply the users suggestions and liaisons preferences to this project.

6. Conclusions

1. The human factor was crucial for the successful fruition of this project.

2. The need to build off the staff's familiarities with the current course information management system was necessary in order to minimize their adjustment and to maximize the utilization of this project's results.

3. A straightforward user interface design of the database application promoted confidence for the user and provided a positive user experience.

4. The database application needs to provide data verification, storage, access and manipulation in order to play its role in the enhanced course information management system.

5. Training and instructional documentation facilitated the adoption of the enhanced course information management system.

6. This IQP allows the staff of CAHES to improve the fulfillment of the pledge to their students.

7. Recommendations

- Try the proposed enhanced course management system; this system can coexist with the current system not requiring a complete change for the staff of CAHES
- If CAHES feels that the proposed enhanced course management system improves their current situation and provides noticeable decrease in workload for the staff, they can switch totally to this new system.
- After the staff of CAHES is familiarized with the database application, they can make further use of it by entering course data with the application instead of importing and export course data.
- If CAHES is comfortable with their data stored in a database instead of Excel files, they should consider upgrading their server and put up a database server on the Internet. This move will potentially satisfy all the needs that have been discussed with our liaisons.
- Keep an updated Excel file with all fields and information in the database on a separate machine that can serve the purpose of a backup database.

8. Works Cited

Bazian, Menachem. Using Visual FoxPro 6. Indiana: Indianapolis, 1998.

Beynon-Davies, Paul. Relational Database Design. Great Britain, 1992.

Folkert, Nathan. Introduction to JDBC. Autumn 2000.
http://www-db.stanford.edu/~ullman/fcdb/oracle/or-jdbc.html

Lee, Chuo-Han. SQL Tutorial. September 29, 2000.
http://homepages.go.com/~chlee1h/sql.html

Stephens, Ryan K. and Plew, Ronald R.. Sams' Teach Yourself SQL. Indiana: Indianapolis, 1998.

Ullman, Jeffrey D. and Widom, Jennifer. A First Course In Database Systems. New Jersey: Upper Saddle River, 1997.

Widom, Jennifer. Getting Started With Oracle. Spring 2000.
http://www-db.stanford.edu/~ullman/fcdb/oracle/or-intro.html#creating%20a%20table

Access Article List. Microsoft Corporation. 2000.
http://officeupdate.microsoft.com/articlelist/o2kAccessArticles.htm

Caritas Adult & Higher Education Service. EPO-HK Ltd. Accessed November 8, 2000.
http://www.caritas.edu.hk

Microsoft Press. Microsoft Visual FoxPro 6.0 Programmer's Guide. Microsoft Corporation. Redmond Washington, 1998.

Visual Basic Technical Resources: Tutorials. Microsoft Corporation. September 7, 1999.
http://msdn.microsoft.com/vbasic/technical/training.asp

Appendix A

Background of Adult & Higher Education Service – Caritas, Hong Kong

Caritas Internationalis is an international confederation of Catholic organizations, run by their respective Episcopal conferences. Caritas Internationalis has 156 members stationed around the World to provide for people with needs. Caritas – Hong Kong is one of the 156 members.

Caritas – Hong Kong was established in 1953 and is dedicated to charity. The Latin word "Caritas" translates into English as "Charity". This truth implies that Caritas is a foundation dedicated to promote the public good (not for assistance to any particular individuals) since this is the definition for "Charity". Caritas is an official social service agency of the Roman Catholic Church in Hong Kong. The organization's purpose and task are to be at the service of individuals, families, groups and communities as they struggle in the society. The activities of Caritas are directed towards understanding, caring and serving people. Ever since the organization was recognized in Hong Kong, Caritas has developed itself into an enormous and well-known charity foundation. Caritas offers numerous services to people of all backgrounds, regardless of sex, race, religion, or ideology. Its purposes are to eliminate divisions, alter selfish and unjust behaviors, and develop the qualities of kindness, reconciliation, forgiveness, and friendship. At present date, it has over 300 services operating from many divisions and in over nearly 140 different locations. These divisions offer social work services, education services, medical services, community & hospitality services and many other services.

For this IQP project, we will be working for the Adult and Higher Education Service, which is a branch of the Education Division of Caritas – Hong Kong. This department started its service in 1963. Its goal is to create a learning society through promoting lifelong education. It offers programs to those in need of receiving education on their first or second opportunities in life. These programs will train them with better education, working skills, community participation and personal development. Programs offered carry a wide range of contents, from basic information to higher-level education. These programs include full-time career trainings, educational programs, part-time programs, summer study programs, extension programs, multimedia education programs, employees training programs and, in association with overseas educational institutions, higher education programs. These programs are offered in many of the twenty-six educational sites registered with the Education Department. The full time programs are offered in eight of the twenty-six educational sites. This program provides educational opportunities for those who wish to continue their studies or to receive vocational training. To guarantee an essential education, the schools emphasize not only technical knowledge and professional skills, but also ethical and social values. Also, there is the part-time program, which offers courses for the people who wish to spend their spare time meaningfully, to make up for their premature education, to improve their quality of life, or to improve their career prospects.

Every year, there are many new arrivals from Mainland China. A lot of these people have minimal education or perhaps no education at all and they do not have the working skills required to compete with the local people in Hong Kong for jobs. Therefore, the Education Department has started a subvention program offering courses for the new arrivals, parents, housewives, farmers and the retired elderly. These courses cover areas such as basic education, literacy, language, induction for new arrivals, job and life skills. Caritas also runs retraining courses for local employees displaced as a result of economic restructuring. This Employees Retraining Program enforces job-related and skill-specific training. In addition to that, the Extension program also provides in-house training courses to give opportunities for individual employees to enhance their personal development and professional competence. A Multi-Media Education Program was established in 1977 for people who are unable to attend regular classes. It includes Education for Parents, Weekend Courses, Secondary Education and Distance Learning Programs. The previous discussed programs are mostly for adults. But Caritas didn't leave out the children and teenagers. The Summer Study Program started in 1974 was designed for primary and secondary students. This program offers school subjects and subjects of recreation.

These programs attracted participants from all walks of life including students, housewives, working adults, the retired and new arrivals. Their age ranges anywhere from 15 to 70 and their educational level ranges from no schooling to secondary schools and to even some college bound. The distributions of the participants are shown in the Figure A.1.

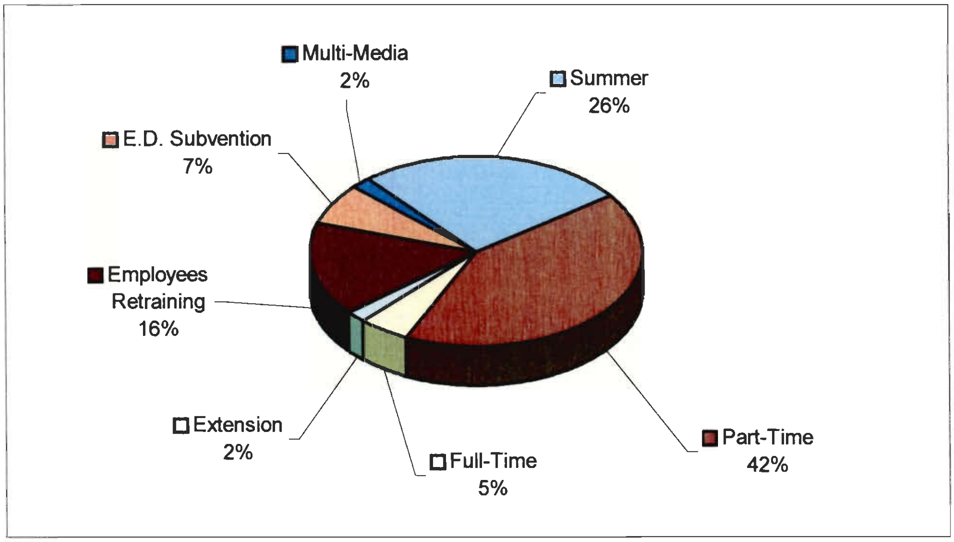


Fig A.1 Total Enrollment: 97626 (April 1996 – March 1997)

(Data obtain from a Caritas Brochure published in March 1998)

Timeline for Caritas' Adult & Higher Education Service:
(Caritas Adult & Higher Education Service, Important Milestones http://www.caritas.edu.hk/ehistory.html)

1963 First part-time courses in domestic science and secretarial training began.

1966 First adult education center set up.

1969 First full-time program started.

1974 Summer Study Program began.

1976 Functional Literacy and Adult Basic Education Program began.

1977 Multi-Media Education Program established.

1978 Tailor-made Extension Program for our business community began.

1980 Collaboration with various mainland institutions in organizing educational programs in Guangzhou, Shenzhen, Zhanjian, Kaiping and other parts of the Mainland began.

1987 City Center to provide human resource development and educational counseling service established.

1989 Linkage with various universities/colleges began.

1990 Caritas Francis Hsu College approved by the former Governor to offer sub-degree program. Affiliation with International Council of Adult Education (ICAE) established.

1991 Affiliation with Asian South Pacific Bureau of adult Education (ASPBAE) established.

1993 Employees Retraining Program to assist local employees affected by economic restructuring to get employment in another trade started. First graduates of Caritas Francis Hsu College since its approval to offer sub-degree program produced.

1994 Learning program through Cable Television launched.

1995 Overseas Studies Collaboration Scheme launched. The Federation for Continuing Education in Tertiary Institutions formed with CAHES being a founding member.

1996 Courses for new arrivals developed into a major undertaking.

1997 Center for Advanced and Professional Studies established. Education and Retraining Information Network for New Arrivals established. Mainland Educational Projects and Information Unit established.

Appendix B

In-depth Qualitative Interview Questions and Results With Catherine Li

1. What is your position at Caritas and what do you do?

Ms. Li is one of the six Assistant Organizers for Academic Affairs Office of Adult and Higher Education. Each Assistant Organizer is responsible for different subjects and different areas. Ms. Li is in charge of Ethics, staff development, student development, course development and the set up of non-formal activities for staff and students.

2. Can you give a brief overview of the Caritas Organization?

Caritas organization is a non-government and nonprofit charity organization that was established in 1953. The activities of Caritas are directed towards understanding, caring and serving people and to the encouragement of their development and co-existence.

3. Can you give us a brief description of the structure of the branches and so forth of Caritas?

Caritas organization has 4 divisions offering medical services, social services, hospitality services and education services. The medical division has clinics and hospitals that provide services to care for the needs of the sick inpatients and outpatients and to operate schools for the training of nurses, midwives and other medical assistants. The social work division offers services for the elderly, family, youth, rehabs, community development, preschool and daycare for those with special needs. The hospitality division has two lodges, a few restaurants, hostiles and family camp facilities. The education division also has four services: Adult and Higher Education, Pre-school Education, Special and Vocational Training and Vocational Training and Education.

4. What is the role of the Adult and Higher Education Services?

Adult and Higher Education service started to offer programs from literacy to higher education. Adult and Higher Education provides an alternate route for the students who do not follow the main stream for continuing their education after senior secondary school. They offer vocational training and practical job related courses. There are no age limits in the Adult and Higher Education programs; most programs are targeted at ages 15 and up. There are also programs for new arrivals from China and some programs are government subsidized.

5. How many schools are in the Adult and Higher Education Branch?

There are twenty-six registered schools and five to six new resource centers for the Adult and Higher Education Branch.

6. What are the most popular classes of the Adult and Higher Education Branch?

Part time classes are the most popular. They offer languages and practical courses: English, bookkeeping and mostly courses for job purposes.

7. Are daytime or nighttime classes more popular?

Nighttime classes are more popular.

8. How many students do you get annually?

The daytime enrollment is about 5000-6000 and the total enrollment is over 100,000.

9. Are there any required exams to get into certain programs such as your college program?

There are no entrance exams required for most courses. Caritas does not reject student unless the class is full. For some of the design and tourist programs, the students have to pass interviews. For the design programs, students must have artistic and creative talents and for the tourist programs, students must not be shy, speak fluent English and have a good appearance. There are optional level tests just to see which courses that students should take according to their basic knowledge about that particular subject. Some courses need prerequisites.

10. What types of degrees are available for each type of program?

Caritas Schools cannot grant degrees because they are not universities, but they do give out diplomas and certificates for their full time students and certificates for the part time students who are able fulfill the course requirements.

11. How helpful would a diploma and certificate in Caritas help a student in terms of future progress?

Full time program is a very good steppingstone. It is a good foundation for the students' career. The schools are very career oriented. The part-time program offers a wide range of courses for different purpose. Some are for interest, some are for practical uses and some are for job development and upgrading purpose. The schools do not offer students with job placement, but they do give information on recent job markets. The Employment Retraining program provides job placements.

12. How was the idea for this project brought up?

New people want the data from our database and right now the work to enter in the data is redundant. To reduce workload, a new database is needed to reduce replication and improve efficiency.

13. In what way do you think this project will help Caritas?

Ms. Li hopes that our project will reduce the staff's workload, so they do not have to manually enter the same course information into different formats.

14. Besides charity, does Caritas have any other income sources?

Adult and Higher Education Schools' income is mostly dependent upon the tuition fees for the classes and very little from charity. Social Work Division depends a lot on charity. Some programs are totally government funded.

Appendix C

In-depth Qualitative Interview Questions and Results With Anthony Leung

1. What is your position at Caritas and what do you do?

Mr. Leung is also an Assistant Organizer for the Office of Academic Affairs. He is responsible for development of science courses and selection of the curriculum. He also trains staff members on the use computers and some applications.

2. How does the current data storage system at Caritas work?

At the Office of Academic Affairs, after Mrs. Amy Chan has approved the course curriculum proposed by the Adult and Higher Education schools or Assistant Organizers at the Academic Affairs office, Rebecca enters the course information into an Excel file with course code, course name and other fields, except that she would leave the fee field blank. This course database file will be passed to the Coordination office where Rachel will put in the fee for each course and she will print it out and distribute the course list to the schools. Since the course information will be used for three purposes: Caritas' Adult and Higher Education Service (CAHES) website, Government's EMB (Education and Manpower Bureau) website and CAHES course catalog. Therefore, someone will have to manually enter the information again and also have to make sure that all the data has been entered error free, because different types of data fields are required for each purpose.

3. Can you give a brief description of what the project is to us again?

The database application will have to merge the course data from different Excel files into a master relational database file. This application outputs different fields for the four purposes: master database file for CAHES internal offices, the CAHES website, the government EMB website and the course catalog.

4. What functions do you hope to see in the database application?

The application should be able to search the database (e.g. by course code, by course date, by course type, etc…), and to export the required data fields in Excel format for each use. The application should take in Excel data files instead of manually entering data using our database application, because employees in the office are more comfortable with Excel and Excel provides spell check.

5. What improvements in the system do you hope to accomplish with a database?

The application would reduce reentry of data, error and workload. It will provide most up-to-date file.

6. Why did you choose to implement the database with Microsoft Visual FoxPro over any other software package?

Many people in Hong Kong are familiar with FoxPro programming, and it will be much easier for us to maintain the application.

7. Who will have access or need access to this database?

Rebecca and Chun – Office of Academic Affair staff will have access to this database.

8. Will databases be distributed or can the master database file be accessed from other computers?

It will be store in a local machine; the database file will be distributed through email or diskettes.

9. Who will be using this application?

Rebecca and Chun will be using this database application.

10. Where will the master database file be stored?

Just like the application, the master database file will be kept in a local machine.

11. To whom do you have to distribute the course information to?

Rachel from the Coordination Office will need the course information to decide the course fee. The course information will have to submit to the Government EMB website. The course catalog publisher will also need the course information. The staff in the Office of Academic Affairs will need the course information as well.

12. Can you tell us a little more on the government continuing education website?

We can hand in specially formatted Excel database files. The Government's EMB (Education and Manpower Bureau) website has a database of all the courses information from all the schools in Hong Kong that offers continuing programs or courses.

Appendix D

Sample Course Development Excel File

CODE	ENGLISH NAME	CHINESE NAME	CTYPE	TYPE	MAKE_CLASS	FEE	DURATION	ISSUE_DATE	MINIMUM
ACC2101	BOOKKEEPING (LCCI FIRST LEVEL) TOPIC REVIEW - PART A	初級簿記(倫敦商會)專題研習 - 甲部				400	16.5	1-Mar-99	15
ACC2102	BOOKKEEPING (LCCI FIRST LEVEL) TOPIC REVIEW - PART B	初級簿記(倫敦商會)專題研習 - 乙部				400	16.5	1-Mar-99	15
ACC2115	CHINESE BOOKKEEPING I	初級中文簿記				670	30.0	29-Aug-97	15
ACC2116	BASIC COST ACCOUNTING	成本會計初階				530	16.5	21-Dec-95	10
ACC2117	BOOKKEEPING	初級簿記				660	30.0	22-Oct-97	15
ACC2124	LCCI FIRST LEVEL BOOKKEEPING REVISION	LCCI初級簿記複習班				280	12.0	10-Sep-98	15
ACC2130	CHINESE BOOKKEEPING (SECONDARY 1)	中文簿記 (中一)					30.0	11-Aug-99	20
ACC2216	INTERMEDIATE COST ACCOUNTING	成本會計進階				520	16.5	13-Jun-96	10
ACC3114	AUDITING	審計實務				830	16.5	1-Mar-99	15

Appendix E

Sample Hong Kong Government Education and Manpower Bureau

Part A: Basic Questions 甲部：基本問題			
Part I: Basic Information of your Institution/Organization			
第一部份：貴機構的基本資料			
Name of Institution/ Organization (English) 院校 / 機構名稱 (英文)	Name of Institution/ Organization (Chinese) 院校 / 機構名稱 (中文)	Correspondence Address (English) 通訊地址 (英文)	Correspondence Address (Chinese) 通訊地址 (中文)
(A1)	(A2)	(B1)	(B2)
Caritas Bianchi College of Careers Evening School	明爱白英奇专业学校夜校	7-9/F 4 Cliff Road Kowloon	4?

Name of Contact Person (English) 聯絡人姓名(英文)	Name of Contact Person (Chinese) 聯絡人姓名(中文)	Contact Person Telephone Number 聯絡人電話號碼	Fax Number (if any) 傳真號碼(如有)	Web Site Address (if any) 網址（？？）
(C1)	(C2)	(D)	(E)	(F)
Mrs Celine Pang, Principal		27102628	27807636	cbccns@caritas.edu.hk

Part B: Information of Programmes/Courses 乙部：? ? ? ?

Please provide us with information on ***each*** of the continuing education programmes/courses conducted:
(a) for persons aged 15 or above; and (b) on or after 3 July 2000.

請就貴院校 / 機構
(a) 為15歲或以上人士；及 (b) 於二０００年七月三日或其後開課的持續教育課程，提供下列資料：

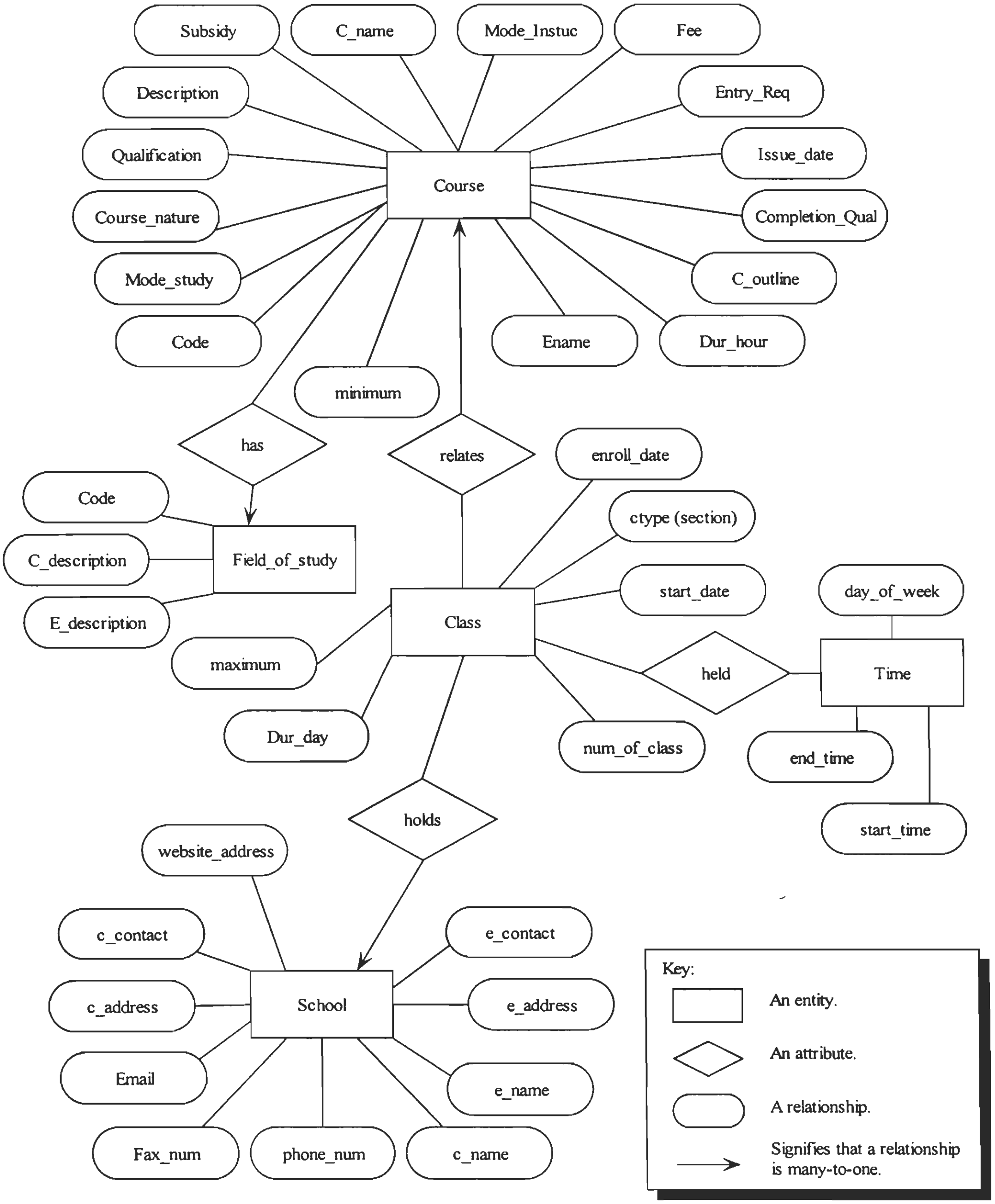
Programmes/courses jointly organized with other institution/organization should also be included.
貴院校 / 機構與其他機構合辦的每個課程，亦請包括在內。

Title of programme/ course (English) 課程名稱（英文）	Title of programme/ course (Chinese) 課程名稱（中文）	Institution code (if any) 機構編號（如有）	Contact telephone number 本課程之聯絡電話	Fax number (if any) 本課程之傳真號碼（如有）	E-mail address (if any) 本課程之電郵地址（如有）
Accounting (LCCI Third Level) Topic Review- Part B	高級會計(倫敦商會)專題研習 - 乙部		23593672	27807636	cbccns@caritas.edu.hk
Accounting (LCCI Third Level) Topic Review- Part C	高級會計(倫敦商會)專題研習 - 丙部		23593672	27807636	cbccns@caritas.edu.hk

Web-site address (if any) 本課程之網址(如有)	Does this programme/ course receive any government subsidy (Yes (Y)/No (N)) 本課程是否得到任何政府資助：是(Y) / 否(N)	Key area of study 主要學習範疇	Name of the co-organizer (English) 合辦機構的名稱（英文）	Name of the co-organizer (Chinese) 合辦機構的名稱（中文）	Nature of the course 課程性質
		(1)	(2B)	(2B)	(3)
www.caritas.edu.hk	2	Accounting			3
www.caritas.edu.hk	2	Accounting			3

Appendix F

Entity-Relationship Diagram of the Course Catalog Database



Appendix G

Object Description Language Model of the Course Catalog Database

```
interface Course {
        attribute string code;
        attribute string mode_study;
        attribute string course_nature;
        attribute enum E_qualifications {PrimaryLevel, JuniorSecondaryLevel,
                                SeniorSecondaryLevel,
                                SecondarySix/MatriculationLevel, Craft,
                                Technician, Sub-degree, Degree, Postgraduate,
                                NotSpecified} qualification;
        attribute string description;
        attribute enum YesNo {Yes, No} subsidy;
        attribute string c_name;
        attribute string mode_instruct;
        attribute currency fee;
        attribute string entry_req;
        attribute date issue_date;
        attribute enum C_qualifications {primaryLevel, SecondaryThreeLevel,
                                SecondaryFiveLevel,
                                SecondarySix/MatriculationLevel, Craft,
                                Technician, Sub-degree, Degree, Postgraduate,
                                AttendanceCertificate, ProfessionalQualification,
                                Certificate/Diploma, NotSpecified}
                                completion_qual;
        attribute string c_outline;
        attribute integer dur_hour;
        attribute string e_name;
        attribute integer minimum;
        relationship Set<Class> relates;
                inverse Class:: relatesto;
        relationship Field_of_Study has;
                inverse Field_of_Study:: belongsto;
};

interface Class {
        attribute integer num_of_class;
        attribute integer dur_day;
        attribute integer maximum;
        attribute string enroll_date;
        attribute string ctype;
        attribute string start_date;
        relationship Course relatesto;
                inverse Course relates;
        relationship Schools holds;
                inverse Schools heldby;
        relationship Set<Time> held;
                inverse Time heldin;
};

interface Time {
        attribute enum Days {Monday, Tuesday, Wednesday, Thursday, Friday}
        day_of_week;
        attribute string start_time;
        attribute string end_time;
        relationship Set<Class> heldin;
                inverse Class held;
};

interface Schools {
        attribute string e_name
        attribute cstring c_name;
        attribute string e_address
        attribute cstring c_address;
        attribute string email;
        attribute string fax_num;
        attribute string phone_number;
        attribute string e_contact
        attribute cstring c_contact
        attribute string website_address
        relationship Set<Class> heldby;
                inverse Class holds;
};
```

Appendix H

Relational Model of the Course Catalog Database

Course(code, mode_study, course_nature, qualification, description, subsidy, c_name, mode_instruct, fee, entry_req, issue_date, completion_qual, c_outline, dur_hour, e_name, minimum, field_code)

Field_of_study(code, c_description, e_description)

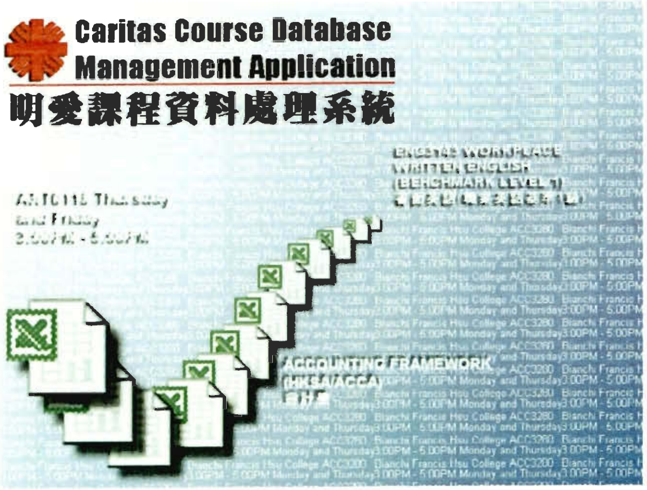
Class(course_code, ctype, school_e_name, maximum, dur_day, num_of_class, enroll_date, start_date)

Time(code, ctype, school, day_week, start_time, end_time)

Schools(e_name, c_name, e_address, c_address, email, fax_number, phone_number, e_contact, c_contact, website_address)

Appendix I

User's Manual


Caritas Course Database
Management Application
明愛課程資料處理系統
WORKPLACE WRITTEN ENGLISH (BENCHMARK LEVEL 1)
ACCOUNTING FRAMEWORK (HKSA/ACCA)

User's Manual

Contents 1

Overview

Welcome

Welcome to the Caritas Database Project. Thank you for trying this new database application and the new proposed course database management system. The usage of this project will help you manage a course database with less redundancy, more security and lighter workloads for the staff and its easy to use. You can still apply the previous course management system in parallel while trying out this newer way of keeping your course data organized. It offers the interface of the frequently seen windows browsers as well as step-by-step wizards, which can guide anyone through the toughest tasks.

About this book

The Course Database Application Manual is a single reference to everything in the application. The booklet is intended to fully inform the users of each aspect of the database application and recommendations for a proposed course database management system, which this application is built upon.

This book is arranged as follows:

Overview

An overview of the database application, this manual and the requirements needed for using this application.

Understanding the Interface of the Application

Diagrams of the main interface and some frequently seen interfaces of the application will be displayed. Detailed definitions of the buttons and menu items contained in these interfaces give the users an idea of what each displayed function can accomplish.

Getting Started

Where do you begin? Anything from installation to starting up the application is outlined and explained here.

Understanding the Application and its Functionalities

Step-by-step instructions on how to use each functionality and explanation of what the purposes of each of these functionalities are. This section also would also give the user an idea of what the internal and external affects of these functionalities are.

Other Notes on the Usage of the Database Application

Some of the difficulties or limitations of the database application will be discussed. Notes on what are some things that should be done to prevent errors or could be done to enhance the usage of the database application.

Trouble Shooting

Possible solution to frequently occurring problems and answers to any frequently asked questions (FAQ) that users might have.

Requirements

System Requirements

- ✓ Microsoft Excel must be installed in order to use the Course Database Application.
- ✓ Chinese Windows must be installed to view all information on the database.
- ✓ The application is best viewed on computers with 800x600 or higher resolution.

Excel Formats

- ✓ Imported Excel files must be in a preset format.

Examples of the Excel Formats could be found in the directory:

C:\CAHES\Database Application\Excel Formats

In this folder, there should be two files. They are:

internaltemplate.XLS
schooltemplate.XLS

If the Application is not installed in the default directory then these formats are found in the folder Excel Formats of the directory you installed the application in. The Excel files in this folder are made only to be a template and are read only. If you wish to add data to it, it will prompt you to save it as something else. This is to ensure that this file will always exist for reference. After you have saved it under another file name, information can now be added to it but the titles for the rows cannot be changed. The Excel files have been locked in areas where changing it could affect the ability to import files. There has been a password set for the editing of these fields.

To edit the files:

1. Open the file you want to edit (internaltemplate.XLS or schooltemplate.XLS)
2. From the main menu Choose

    Tools ⟹ Protection
    Protection ⟹ Unprotect sheet

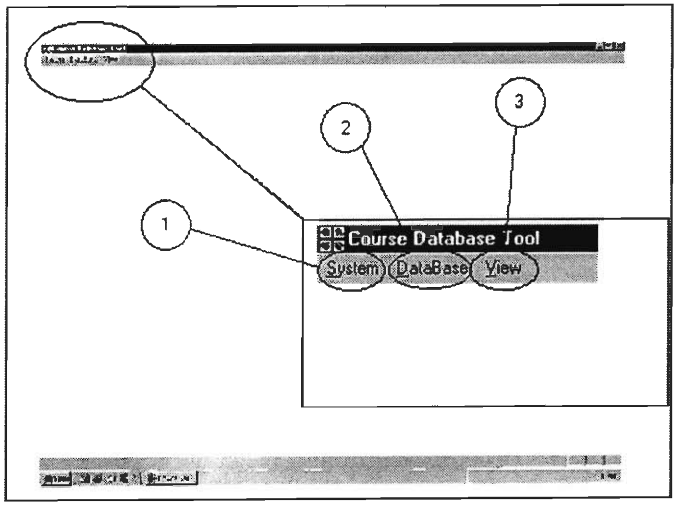
3. A window prompting for a password will pop up. Enter the password “cahes” and the sheet will be unlocked for editing.

Editing the sheet is highly not recommended. It is recommended that a back up copy be made if this is to be attempted.

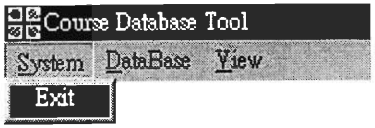
Understanding the Interface

Main Menu

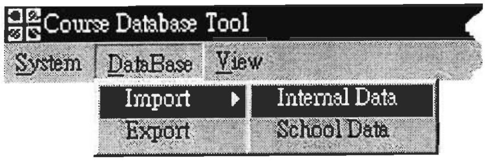
The main menu is the second interaction between the user and the application besides the welcome screen. You will see this menu throughout your entire usage of the application. If the user gets lost in the application, they can always cancel what they are previously doing and restart from the main menu.



The numbers above is for future reference



1. System – This menu selection allows the user to exit the application. *(Important note: Any other Windows exit tool does not apply in this database application)*



2. Database – This menu selection allows for importing and exporting of data. It provides seven main functions:

1. Import *(Important Note: Imported data must be in the Excel Formats noted in the requirements)*

a. Import Internal Data – Data from Office of Academic Affairs.

b. Import School Data – Data from each school.

c. Import All School Data– Continuously Import school data until user is done importing all desired schools data.

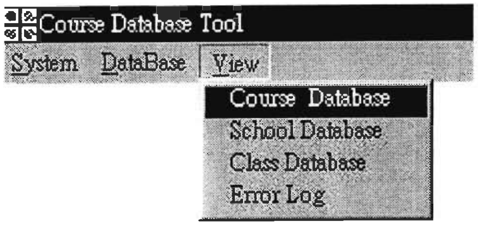
2. Export *(Important Note: Fields included in each type of export will be noted in the export section of this manual.)*

a. Export EMB – File for Education and Man Power Bureau Website.

b. Export Course Catalog – File for the course catalog.

c. Export Caritas Website – File for the Caritas website.

d. Export Internal – File for the Office of Academic Affairs.



3. View – Allows for views of different data. When viewing, you may search the data that you are currently viewing. The data may also be edited and the user can also add or delete data from the fields in the active view. There is no editing feature for the error log.

1. Course Database View – Data sorted by courses with helpful course information.

2. School Database View – Displays all the classes of what each school offers and other School information.

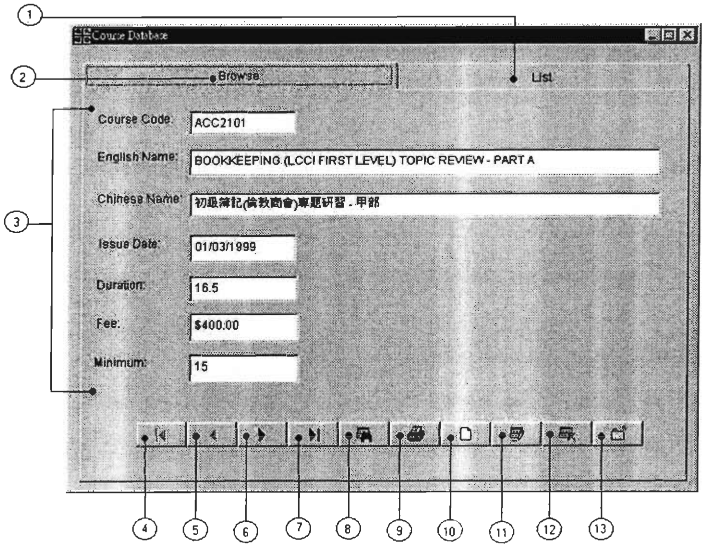
3. Class Database View – Display all the classes of all the schools and their relative information.

4. Error Log View – This will display all the errors that the application found in the check before the actual importation. It will only show the errors of the most current checked file.

Data Browsers

Course Browser

The course browser allows the user to view the information by course. The information can be seen one course at a time or as a list of all the courses available at Caritas. The following is the browse window for viewing one course at a time.



Course Code: ACC2101

English Name: BOOKKEEPING (LCCI FIRST LEVEL) TOPIC REVIEW - PART A

Chinese Name: 初級簿記(倫敦商會)專題研習 - 甲部

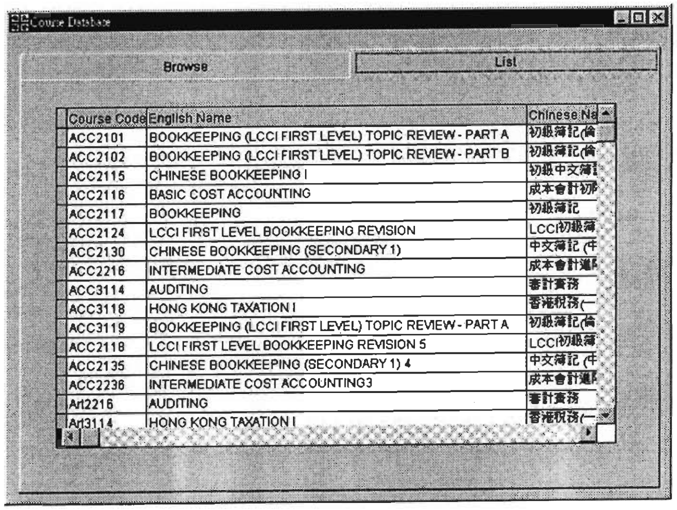
Issue Date: 01/03/1999

Duration: 16.5

Fee: $400.00

Minimum: 15

The following is a list of all the courses available at Caritas.



Course Code	English Name	Chinese Na
ACC2101	BOOKKEEPING (LCCI FIRST LEVEL) TOPIC REVIEW - PART A	初級簿記(倫
ACC2102	BOOKKEEPING (LCCI FIRST LEVEL) TOPIC REVIEW - PART B	初級簿記(倫
ACC2115	CHINESE BOOKKEEPING I	初級中文簿
ACC2116	BASIC COST ACCOUNTING	成本會計初
ACC2117	BOOKKEEPING	初級簿記
ACC2124	LCCI FIRST LEVEL BOOKKEEPING REVISION	LCCI初級簿
ACC2130	CHINESE BOOKKEEPING (SECONDARY 1)	中文簿記 (中
ACC2216	INTERMEDIATE COST ACCOUNTING	成本會計進
ACC3114	AUDITING	審計實務
ACC3118	HONG KONG TAXATION I	香港稅務(一
ACC3119	BOOKKEEPING (LCCI FIRST LEVEL) TOPIC REVIEW - PART A	初級簿記(倫
ACC2118	LCCI FIRST LEVEL BOOKKEEPING REVISION 5	LCCI初級簿
ACC2135	CHINESE BOOKKEEPING (SECONDARY 1) 4	中文簿記 (中
ACC2236	INTERMEDIATE COST ACCOUNTING3	成本會計進
Art2216	AUDITING	審計實務
Art3114	HONG KONG TAXATION I	香港稅務(一

1. The list tab – it leads to the window with a listing of all the courses available at Caritas. This list contain all the information in the browse window. Above you can only see the fields "Course Code, English Name, and Chinese Name," but the scroll bar on the bottom allows the user to see the rest of the information.

2. The browse window tab – it which helps navigation between the list window and the browse window. Information on this window will be described.

3. The information fields – these fields display all the information of the specified course. It includes Course Code, English Name, Chinese Name, Issue Date, Duration, Fee and Minimum.

1. Course Code – The code that uniquely identifies each course. The first three letters of the course code identifies the subject of the course and the last four digits relates to the level of difficulty of the course.
2. English Name – The English name of a course.
3. Chinese Name – The Chinese name of a course.
4. Issue Date – The date the course was officially approved by the Office of Academic Affairs.
5. Duration – The length of the course by hours.
6. Fee – The cost of the course per length of the course.
7. Minimum – The minimum amount of students needed for the course to be opened for teaching.

Items four through seven are used to navigate the data in the course browser tab.

4. To Beginning – this button is used to go all the way back to the first record of the database in the course browser tab. No matter which record the user is currently browsing, this button

will always bring the user back to the beginning. Pressing the button on the first record will have no affect.

5. Back – this button will go back one record. Pressing this button on the first record will have no affect.

6. Forward – this button will go forward one record. Pressing this button on the last record will have no affect.

7. To End – this button will skip to the last record from the current viewing record. Pressing this button on the last record will have no affect.

8. Search – this button will open the custom search window. This window will allow the user to search from any of the fields included in this database or all the fields mention is item three. This function will be discussed in more detailed in the search and sort section.

9. Print – This button will actually do more then just print the data; it also allows the export of data and previews of data.

10. Edit – Allows the user to edit the current record. All fields are originally read only, but in edit mode, it will be readable and editable.

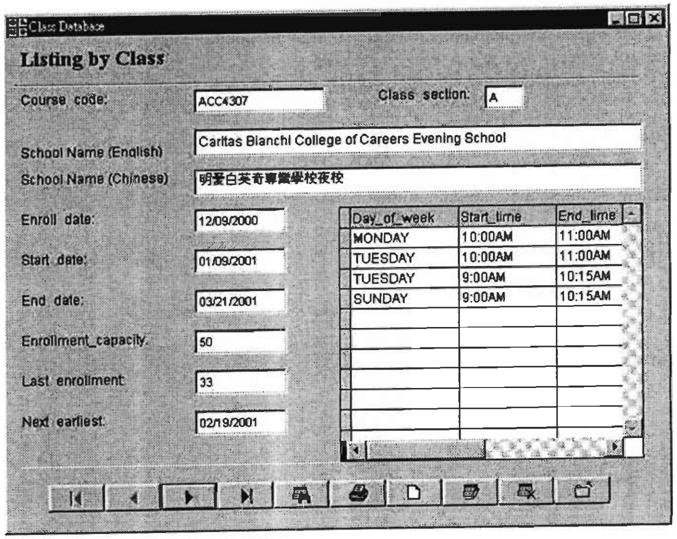
11. Add – This allows user to add approved courses into the course database directly. All the course information will need to be filled out in fields identical to the ones of the browse windows. Invalid data will be found instantly after entry if entered.

12. Delete – The ability to delete records directly off the database. This is the only way to delete data from the database. *(Incorrect deletions can cause errors in the database.)*

13. Exit – Exit the course browser window.

Class Browser

The class browser allows the user to view information by class. This means that it can potential contain thousands of records if all classes have been imported. The times for each class, which school it is held at and other information is included.



Listing by Class

Course code: ACC4307

Class section: A

School Name (English) Caritas Bianchi College of Careers Evening School

School Name (Chinese) 明愛白英奇專業學校夜校

Enroll date: 12/09/2000

Start date: 01/09/2001

End date: 03/21/2001

Enrollment_capacity: 50

Last enrollment: 33

Next earliest: 02/19/2001

Day_of_week	Start_time	End_time
MONDAY	10:00AM	11:00AM
TUESDAY	10:00AM	11:00AM
TUESDAY	9:00AM	10:15AM
SUNDAY	9:00AM	10:15AM

As you can see, this window is similar to the one of the course browser. Information fields and a navigational bar make up this class browser. The information fields include Course Code, Class Section, School English and Chinese name, Enroll date, Start Date, End Date Enrollment Capacity, Last Enrollment and Next Enrollment. The Chart on the left of the window clearly lists which days the class is held and at what time on each of the days.

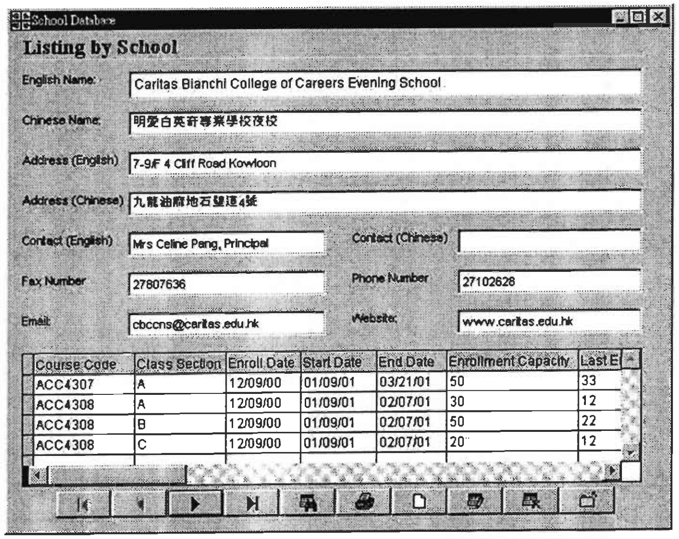
Clearer explanation of the information is as follows:

1. Course Code – the code that uniquely identifies what course is this class teaching.
2. Class Section – the section of the class to this school.
3. School English Name – the English name of the school the class is held at.
4. School Chinese Name – the Chinese name of the school the class is held at.
5. Enroll Date – the date of the last day in which the student can enroll into this class.
6. Start Date – the begin date of the class.
7. End Date – the date of the last day of class.
8. Enrollment Capacity – the maximum number of students allow to enroll in this class.
9. Last Enrollment – the number of students enrolled in this class the last time this class was taught in this school.
10. Next Earliest – the next time this class will be held in this school.
11. Time Chart – information of time this class will be held.

The navigation bar serve the same purpose as before, except this time the edit, add and delete buttons will be used to edit the class information. Again, deleted information cannot be retrieved unless the data has been backed up and this is the only place where school records can be deleted.

School Browser

This browser lets the user see what each courses and classes each school is offering. The schools information will be shown in the text fields of this browser.



Listing by School

English Name: Caritas Bianchi College of Careers Evening School

Chinese Name: 明愛白英奇專業學校夜校

Address (English) 7-9/F 4 Cliff Road Kowloon

Address (Chinese) 九龍油麻地石壁道4號

Contact (English) Mrs Celine Pang, Principal

Contact (Chinese)

Fax Number 27807636

Phone Number 27102628

Email: cbccns@caritas.edu.hk

Website: www.caritas.edu.hk

Course Code	Class Section	Enroll Date	Start Date	End Date	Enrollment Capacity	Last E
ACC4307	A	12/09/00	01/09/01	03/21/01	50	33
ACC4308	A	12/09/00	01/09/01	02/07/01	30	12
ACC4308	B	12/09/00	01/09/01	02/07/01	50	22
ACC4308	C	12/09/00	01/09/01	02/07/01	20	12

Above is the course view for Caritas Bianchi College of Careers Evening School. All the schools information such as address and contact information are listed. The courses that are held at this school currently will be shown on the chart below all the information. The chart will clearly state the course code and dates this class will be held. More information about the class can then be looked up in the class view.

1. English Name – English name of the school.
2. Chinese Name – Chinese name of the school.
3. English Address – Address of the school in English.
4. Chinese Address – Address of the school in Chinese.
5. Contact (English) – English name of contact person for the particular school, often the principal.
6. Contact (Chinese) – Chinese name of contact person for the particular school, often the principal.
7. Fax Number – The schools fax number.
8. Phone Number – The schools telephone phone number.
9. Email – The email address of the school.
10. Website – The schools website address.
11. Class information chart – chart-containing information on the class.

Getting Started

Installation

An installation wizard has been built to guide the user through the installation process.

1. Locate the Course Database Application installation CD.
2. Put the CD in the CD-ROM drive
3. From the Windows desktop, open My Computer
4. Open the CD-ROM drive usually labeled D:
5. Run setup.exe

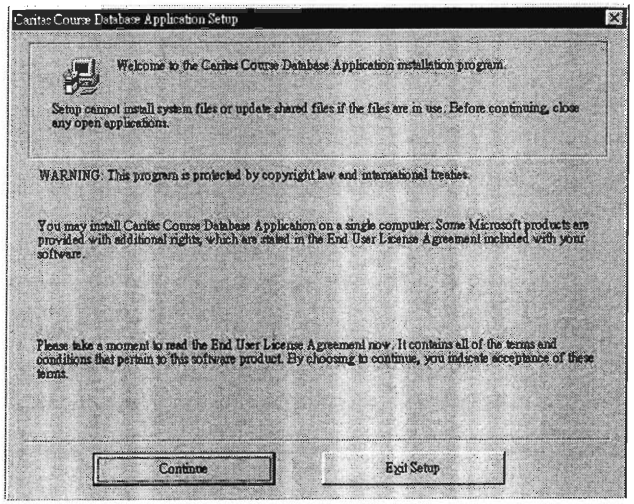
Or

1. Locate the Course Database Application installation CD.
2. Put the CD in the CD-ROM drive
3. From the Start Menu, choose Run
4. Key in: D:\setup.exe

(Important Note: CD-ROM can often be labeled as E: or any letter the administrator sets it to.)

After the above steps have been accomplished, a setup wizard should pop up on the screen. Follow the screen instructions and choose your preferences when asked and the installation should take no longer then five minutes depending on the speed of your computer.

1. The beginning welcome screen states licensing facts about the application. Click 'Continue.'



Welcome to the Caritas Course Database Application installation program.

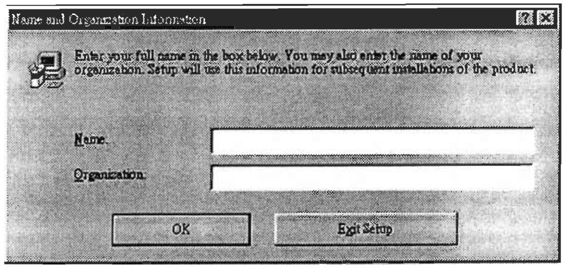
Setup cannot install system files or update shared files if the files are in use. Before continuing, close any open applications.

WARNING: This program is protected by copyright law and international treaties.

You may install Caritas Course Database Application on a single computer. Some Microsoft products are provided with additional rights, which are stated in the End User License Agreement included with your software.

Please take a moment to read the End User License Agreement now. It contains all of the terms and conditions that pertain to this software product. By choosing to continue, you indicate acceptance of these terms.

2. Enter your name and the name of the Organization this application will be used for.



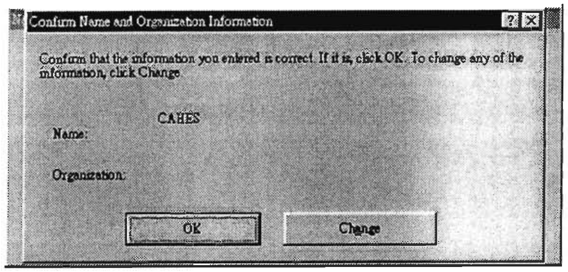
Name and Organization Information

Enter your full name in the box below. You may also enter the name of your organization. Setup will use this information for subsequent installations of the product.

Name:

Organization:

3. Verify the information you have entered. If the information is correct, then click 'ok.' You can change the information by clicking 'change.'



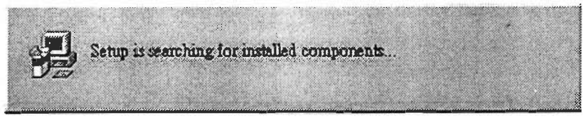
Confirm Name and Organization Information

Confirm that the information you entered is correct. If it is, click OK. To change any of the information, click Change.

Name: CAHES

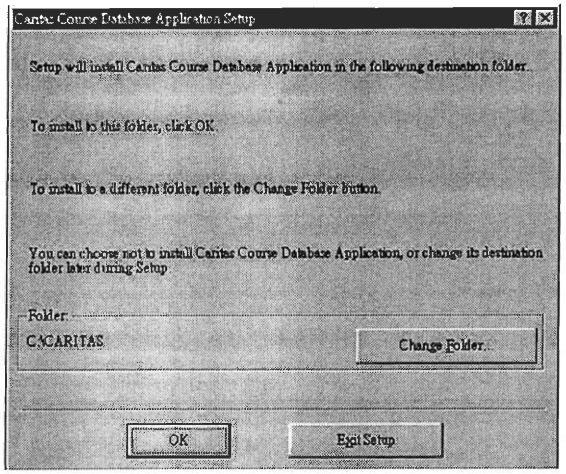
Organization:

4. The install wizard will then search for components installed into your computer.



Setup is searching for installed components...

5. Choose the directory where you would like to install the application into. Click 'Change Folder' to choose a new directory.



Caritas Course Database Application Setup

Setup will install Caritas Course Database Application in the following destination folder.

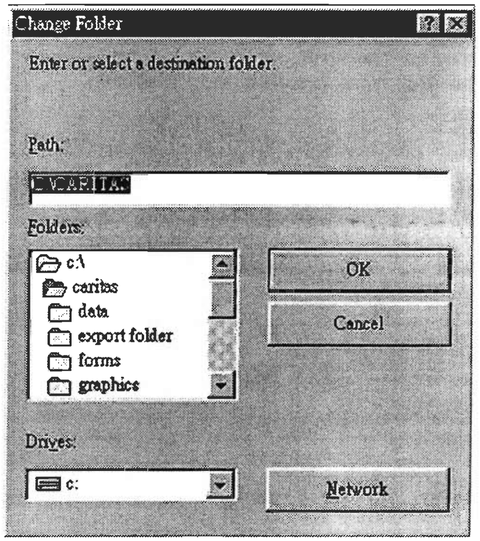
To install to this folder, click OK.

To install to a different folder, click the Change Folder button.

You can choose not to install Caritas Course Database Application, or change its destination folder later during Setup.

Folder:
C:\CARITAS

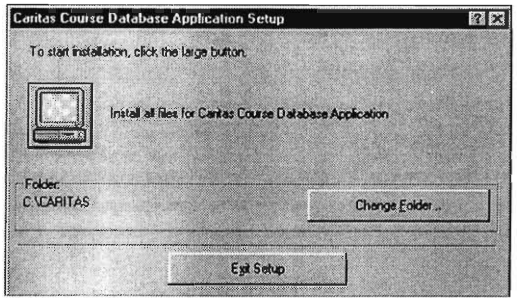
6. If you choose to change directories, the following window will be displayed. Choose your desired directory.


Change Folder

Enter or select a destination folder.

Path: C:\CARITAS

7. To verify the directory you have chosen, the following screen will be displayed next. Click on the Computer button to install.

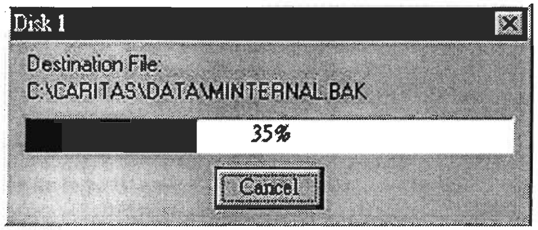

Caritas Course Database Application Setup

To start installation, click the large button.

Install all files for Caritas Course Database Application

Folder: C:\CARITAS

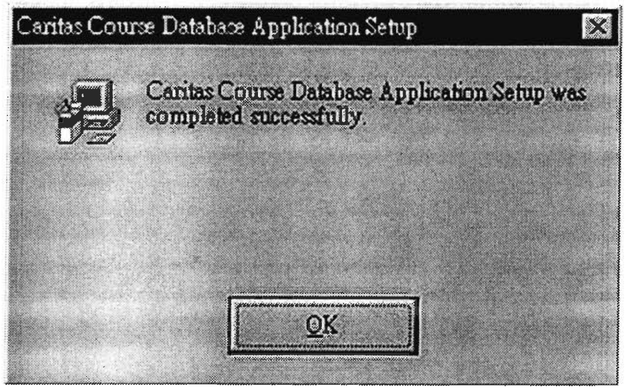
8. The meter to indicated percentage of installation done.


Disk 1

Destination File:
C:\CARITAS\DATA\MINTERNAL.BAK

35%

9. The installing process is done. Click ‘Ok’ to begin.



Caritas Course Database Application Setup

Caritas Course Database Application Setup was completed successfully.

Start Up

Now you are ready to use the application. The installation wizard will automatically place a desktop icon and a start menu shortcut to the application. Access the application by either the icon on the desktop or through your start menu and you are now ready to take advantage of the Caritas Course Database Management Application. You will be managing your own courses in a more organized way and be able to add and remove data in the matter of a few clicks.

Tools to help Manage the Database

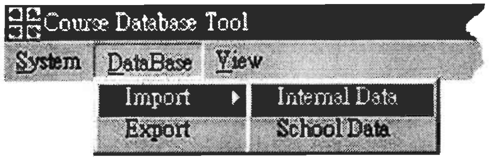
Import

Data from Microsoft Excel is imported into the database application and the data is stored in a database with table structures. These tables are related together so that no redundant and invalid information can be stored.
Importation of data will fill the database with information from Excel files. These data will be check to meet certain criteria before any actual writing to the database will occur eliminating data that can be false. There are two types of data that will be imported, internal data or data from the Office of Academic Affairs and School data or data from each of the schools of Caritas. *(Internal Data must be in the database before any school data is imported)*

Importing Internal Data

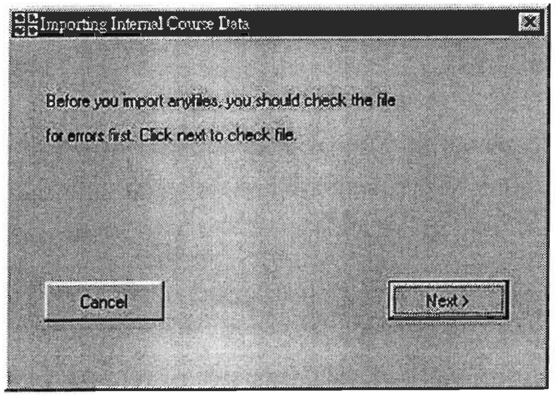
From the main menu Choose

Database ⟹ Import
Import ⟹ Internal Data or School Data



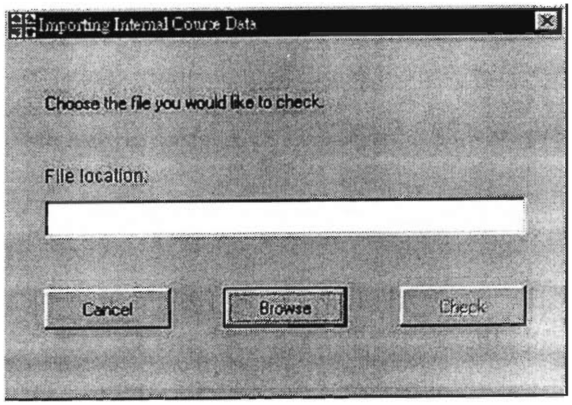
An importation wizard should appear and onscreen instructions will guide you through the process.

1. Before importing any file, the file should first be checked for errors, click 'Next' to choose a file to check and import.



Before you import anyfiles, you should check the file for errors first. Click next to check file.

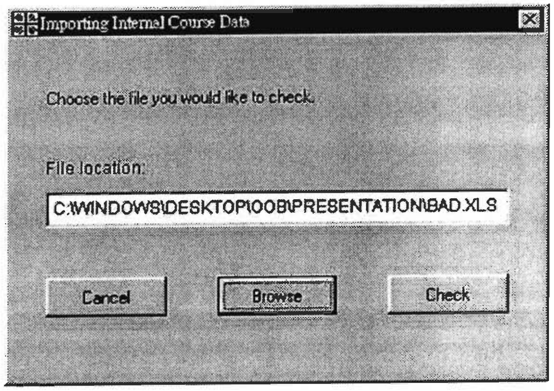
2. What file would you like to import, click 'Browse' to look through your computer for the file or key in the path manually.



Choose the file you would like to check.

File location:

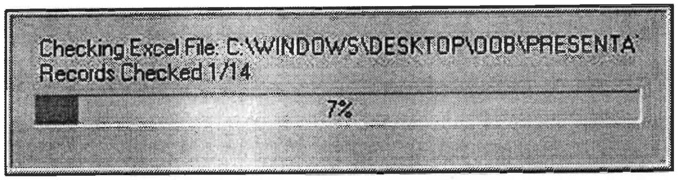
3. When the path has been chosen, the Check button should be enable, click 'Check' to continue on with the checking.



Choose the file you would like to check.

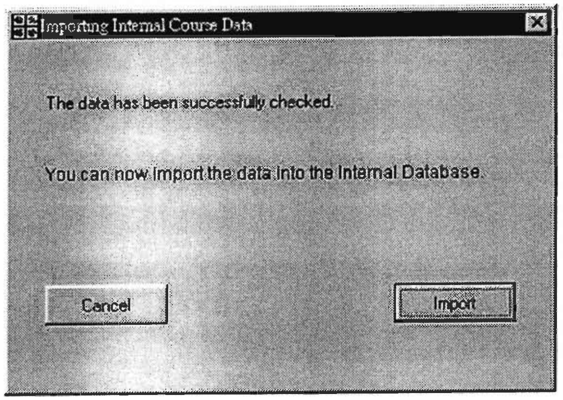
File location: C:\WINDOWS\DESKTOP\OOB\PRESENTATION\BAD.XLS

4. A status bar of the checking process will show up. Which file is being checked is displayed on 'Checking Excel File' and 'Records Check' show how many records have been checked out of the total number of records.



Checking Excel File: C:\WINDOWS\DESKTOP\OOB\PRESENTA
Records Checked 1/14
7%

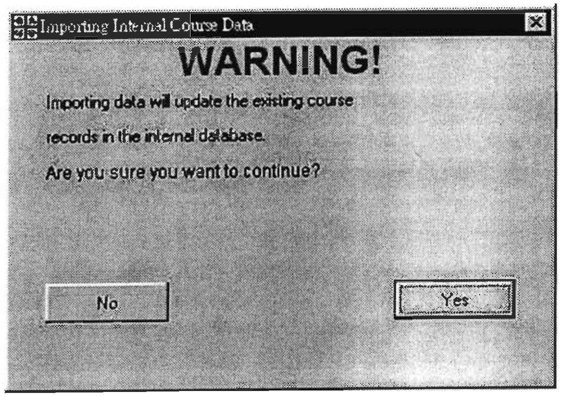
5. If no errors are found, a successful checking screen should pop up.



The data has been successfully checked.

You can now import the data into the Internal Database.

6. This warning states that importing new data will mean that old data will be over written. If you would like to preserve the old data, then you should export all of your current data first.

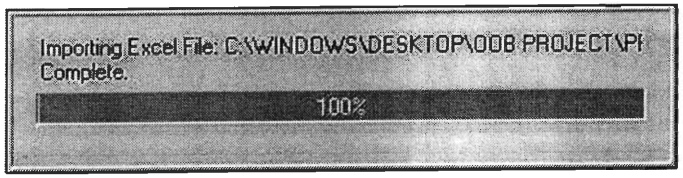

Importing Internal Course Data

WARNING!

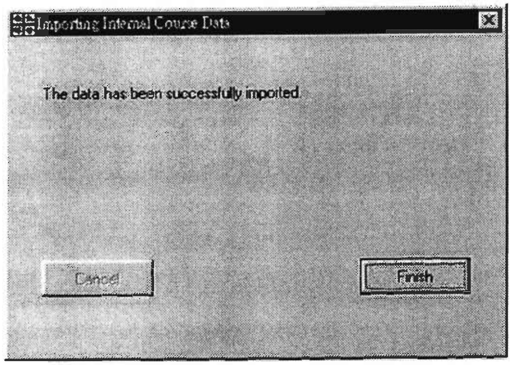
Importing data will update the existing course records in the internal database.

Are you sure you want to continue?

7. Clicking 'Yes' on the warning screen will lead to the actual importing of data. Like the Checking, A status bar of how many records have been successfully imported will show up.


Importing Excel File: C:\WINDOWS\DESKTOP\OOB PROJECT\PI
Complete.

8. After completion of the importing process, the following screen states that the process is done.


Importing Internal Course Data

The data has been successfully imported.

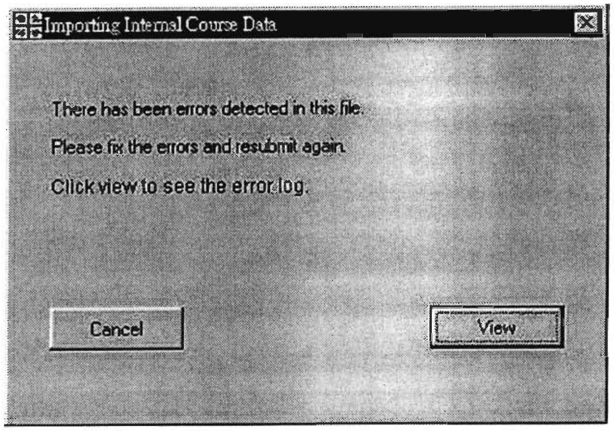
What if an error has been detected?

During the Checking process, error could be detected, not enabling you to import the data. This is because there is an error or errors in the file that you would like to import. The application allows you to view a list of these errors so that you can correct them and then import the data.

The following screen shots and explanations will give you an idea of an error-founded process. Errors in Excel files are common due to flaws in data entry or inconsistent data. This is not a big problem and could be easily fix.

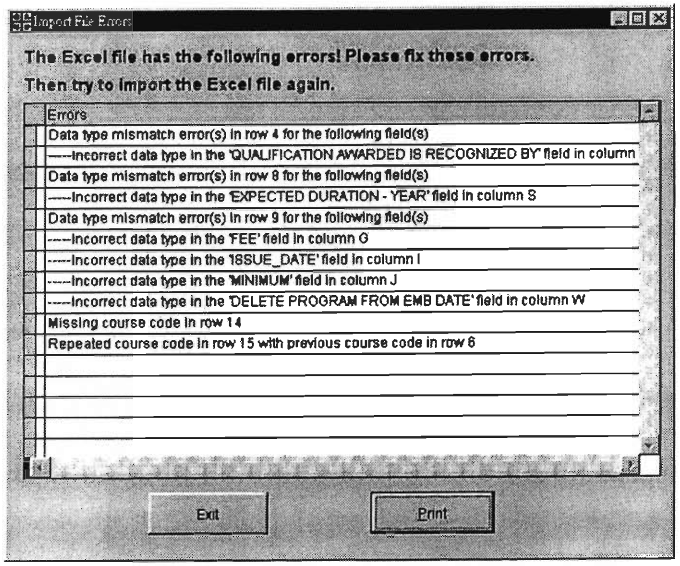
After the checking process from step 4 in Importing Internal Data, if errors were founded then the following screens will follow. We will also start the numbering process from 5 in continuation of step 4 from above.

5. When errors are found, the following screen will show to inform you that errors are detected.



There has been errors detected in this file.
Please fix the errors and resubmit again.
Click view to see the error log.

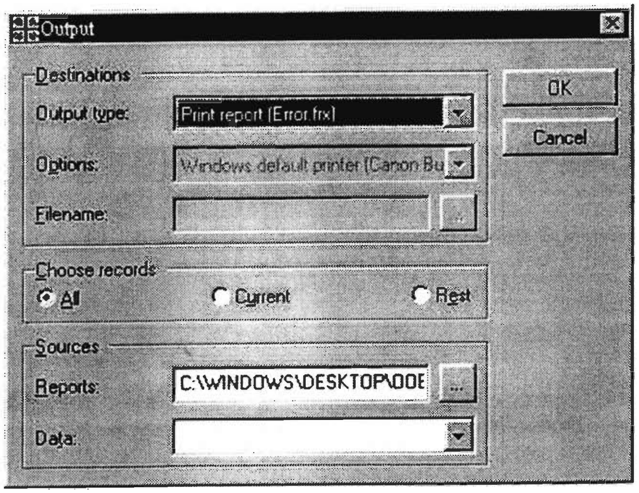
6. Clicking the 'View' will allow you to see a list of the errors. *(if you accidentally hit cancel or you may not want to view the errors now, these errors are saved in a table until the next set of importation errors overwrite it. You may come back to this full list of errors from the main menu -> view -> error log.*



The Excel file has the following errors! Please fix these errors.
Then try to import the Excel file again.

Errors
Data type mismatch error(s) in row 4 for the following field(s)
-----Incorrect data type in the 'QUALIFICATION AWARDED IS RECOGNIZED BY' field in column
Data type mismatch error(s) in row 8 for the following field(s)
-----Incorrect data type in the 'EXPECTED DURATION - YEAR' field in column S
Data type mismatch error(s) in row 9 for the following field(s)
-----Incorrect data type in the 'FEE' field in column G
-----Incorrect data type in the 'ISSUE_DATE' field in column I
-----Incorrect data type in the 'MINIMUM' field in column J
-----Incorrect data type in the 'DELETE PROGRAM FROM EMB DATE' field in column W
Missing course code in row 14
Repeated course code in row 15 with previous course code in row 6

7. This list of errors can also be printed out for ease of reference when correcting the Excel file. If the correcting of the Excel file is not made on the computer with this application installed, a print out of the error log may prove useful. Hitting print on the above screen will lead to the following screen. After choosing the 'Output type,' just click ok and a report of all the errors will be printed.



Output type: Print report (Error.frx)

Options: Windows default printer (Canon Bu

Filename:

● All ○ Current ○ Rest

Reports: C:\WINDOWS\DESKTOP\DOE

Data:

The Error log is used to correct all the errors in the Excel file that you wanted to import. For the internal import, an explanation of the possible errors and what each mean is listed below:

(# denotes a number in the following list)

1. Missing course code in row #

A course code is required for all courses. These codes are used to reference each class in the database. A course is not known in the database unless a course code is included. Also courses without course codes could be non-approved courses.

2. Data type mismatch error(s) in row # for the following field(s)

This means that the type of data you have entered in the field is not what the database is looking for. Area's which expect a number for data such as 'maximum students' would not make sense if the letter A were in this field. Expected data types are explain in the comments of each Excel column.

3. Repeated course code in row # with previous course code in row #

One course can only be entered into the database once with one set of data that goes with it. If there has been a mistake in the process of approval of a course, and a course code for an already existing course was given to a new approved course, this will cause an error.

4. Unknown error in row #

This is an error that has not been anticipated. The row number is given so that the user can take a look at the data in that row to find the error.

Simply open up Excel and the file that you want to import and fix the errors according to the instructions in the error log and above. The rows and columns of the errors are clearly stated in the error log.

After all the errors have been fix, then attempt to import the file again. If errors still exist then not all the errors has been fixed or fixed correctly yet.

Importing School Data

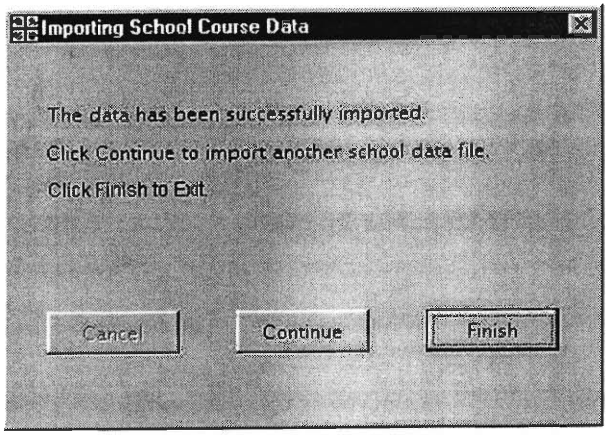
The Importing of the school data is much like the importing of the internal data. *(Before you import School Data, make sure internal data has already been imported.)*

From the main menu Choose

Database ⇨ Import
Import ⇨ School Data

Follow steps 1 – 7 in 'Importing Internal Data,' however due to the fact that there will be multiple school data to be imported per season unlike the one set of internal data, the final screen gives you a choice to continue importing. The 8th step will be a little different in that it will include a 'Continue' button. Clicking 'Continue' will bring you back to step 2 where you are prompt for another file name to import. This can be done until all the schools information has been imported into the database.

The screen will look like this:



The data has been successfully imported.
Click Continue to import another school data file.
Click Finish to Exit.

What if Errors are detected in the School Data?

Fixing errors for the school data will be the same process as fixing errors for the internal data. *(Refer to 'What if Errors are detected in the Internal Data?')* The same format of the error log as the one for the internal data will show the errors for the school data. However, there are different errors that are checked for the school file and the internal file, therefore different errors could appear in the error log. These errors is explained and listed in the following:
Numbers are denoted with '#.'
The name of an Excel worksheet is denoted with '(worksheet name).'

1. Missing email address

    Each school must have an email address for contact and referencing purposes in the database.

2. Improper email address: must include '@'

    An email address without the symbol '@' is detected as a faulty email address.

3. Missing course code in row #

    (Explained in 'Internal Data Errors')

4. Missing section letter in row #

    The section letter is also a required field.

5. Invalid course code detected in row #

    A course code has been detected that has not approved yet. (Also explained in 'Internal Data Errors')

6. Data type mismatch error in row # of the (worksheet name)

    (Explained in 'Internal Data Errors')

7. Repeated class in row # with previous class in row # of the (worksheet name)

    There cannot be two of the same class.

8. Unknown error in row # of the (worksheet name)

    (Explained in 'Internal Data Errors')

Export

One of the major advantages of using the database application is its ability to do multiple outputs saving much data entry time. The export feature will allow exports of data in multiple formats to make use in multiple ways. Exporting the data will ensure that the data is exactly the same way as it was when it was inputted into the database. Hand key entry can cause many typos.

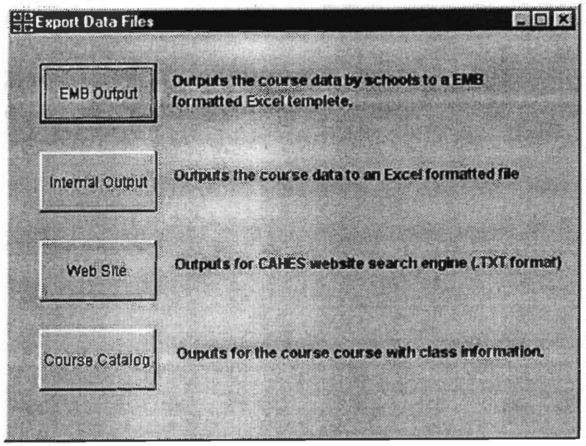
How to Export Data?

From the main menu Choose

Database ⟹ Export

Then the exporting wizard will take over and filling an Excel file up with the data you desire is just a matter of clicks away.

1. Choose a format to export. There are four formats for you to choose from. EMB Output will give you a format for the EMB website. This is one of the more detailed outputs consisting of many fields. The Internal Output has the fields most used for the Office of Academic Affairs. All the courses that Caritas has offered and will offer the current term will be listed. The Web Site outputs a file formatted for the Caritas web site's search engine. And the Course Catalog output, this output contains all the classes of the current term.



Outputs the course data by schools to a EMB formatted Excel templete.

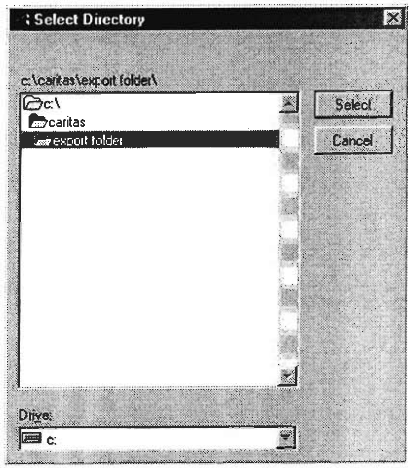
Outputs the course data to an Excel formatted file

Outputs for CAHES website search engine (.TXT format)

Ouputs for the course course with class information.

2. After you have chosen the type of output you prefer, there will be a prompt for the directory you prefer to output to. After the directory has been chosen, click 'Select' and the application will start to write the file. This can take a bit of time depending on the size of the file. There can be an expected wait for both the EMB website file because there is more then one file that needs to be written to. There will be a file written for each school. The Internal file should take a bit of time also because

of the large amount of classes offered and the large amount of data for each class.



3. After successfully outputting the data, a screen will be display to ensure that this has been done.

Depending on what type of format you have chosen, different files will be generated in the directory you have specified. The files names of each output is different and has a preset output file name, they are the following.

File names for Application outputs:
EMB Output – (Email Address of School).XLS

An EMB output will be generated for each school, therefore there could be multiple files outputted each time you want to output. Each of these files will be named accordingly to there school email address. For Example for Caritas Bianchi College of Careers Evening School, their email address is CBCCNS@Caritas.edu.hk. The name of the EMB output file for this school will be CBCCNS.XLS.

Internal Output – Course File.XLS

Website Output – weboutput.XLS

Course Catalog Output – course catalog output.XLS

Just remember the directory you have chosen to output the files and when you click into the directory the file of your choice should appear in the above naming form depending on what your output choice was.

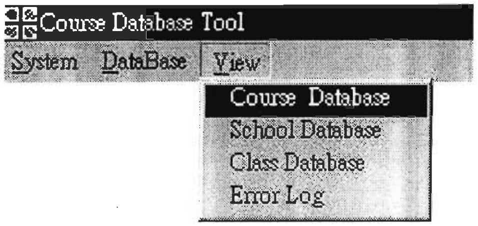
View Databases

Although the different view of the database has already been discussed. What are the advantages and disadvantage of each view over each other and when is each view adequate to use.

Usage of Different Views

From the main menu Choose

View ⟹ Course Database,
School Database,
Class Database,
Error Log



Class View

The class view is the only view with the days and times of each class. To find specific information regarding the class, you can search it in this view. To uniquely identify each class however, you would need to know which school it is held at, what section it is and what is course code does this class has. So if you would like to locate a single class, you would need all this information. The search from the class view makes it easy in that it will narrow your choices one by one

Course View

This will show all the courses that Caritas has in their entire program. This view makes it easy to see what type of courses Caritas has and specific information such as the cost of each course. To uniquely find a course, all that is needed is the course code.

School View

The school view obviously contains much information about the particular school. Contact information, the school's address and website, but what is the most useful is that it also contains all the classes each school offers.

Error Log View

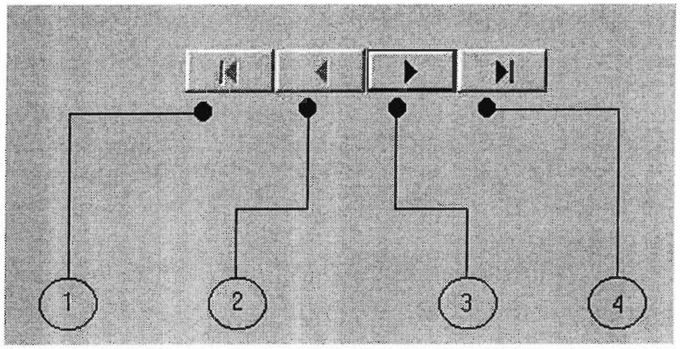
As discussed before the error log contains information that has been detected in the previously checked excel file. This view is just for correction purposes.

Additional Tools and the Navigation Bar

As you might have notice, there is a navigation bar on the bottom of each view. This navigation bar not only serves to navigate through data but it includes additional tools to modify the data in the current viewing window.

Navigation Tools

There are actually only four buttons primarily used for navigation of the data.



1. Back to the beginning – this button will skip to the beginning of the list of data. Clicking this button on the first record will have no affect because you are already at the beginning.

2. Back – this button goes back one record.

3. Forward – this button goes forward one record

4. To the end – this button skips to the end of the list of data.

Search and Sort

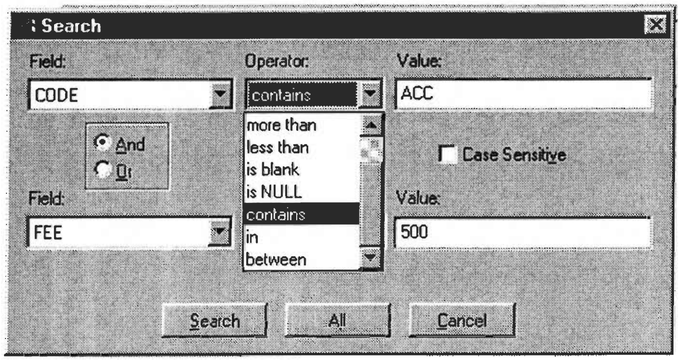
The application enables the user to search for any set of information with similar attributes. This means that for a set of classes that is offered at the Francis Hsu College, then the information for these classes could be found in one search. This is because they have one similar attribute. The application can almost search for anything from any information included in the database.

How do I search for what I want?

From the main menu choose any of the following

View ⇨ Course Database,
School Database,
Class Database

The navigation bars on the bottom of these three views have a search button. When the button is clicked you will see a window that allows you to choose a searching criteria and search the data.



Search

Field: CODE

Operator: contains

Value: ACC

● And ○ Or

Field: FEE

Value: 500

□ Case Sensitive

This form helps you make a sentence like structure that basically tells the application what you are looking for. The above picture is the search for the course view. The field sections lets you choose what items you can search, such as course code and the fee of a course. The operator is the relationship between the field and the value. The value can be whatever you enter. So that when you have finished choosing the field, operator and have entered a value, you have made a sentence that describes what you want to search for. For example: You have chosen English Name for the field, equals for the operator and you enter Accounting for the value. This could be read as "Search for a course with the English Name equal to Accounting," the application will then look through all the courses and pick out the ones with the word accounting in the name. Notice that there are two sets of these fields that you can choose; this is so that you can search with even more detail and limit your results to more accurately reflect what you are looking for. The above picture would read something like this, "Search for a course where the course code contains the letters ACC and the fee is less then 500 dollars." Obviously the user is looking for a cheaper accounting course. The possible searching fields are different for each type of view. The course view search contains the ability to search by course code, by the cost of the course, duration of the course and much more information relative to the course, but the class view and school view would contain searchable fields that are related to them. However, the operators or relationships between the field and value will stay the same. These operators will cover any type of search that you can probably think of. If one operator doesn't cover it, most likely a combination of the first and the second will with the inclusion of the ability to choose 'and' or 'or.'

Available operators:

Equal – the value is the same as the field.

Not Equal – the value is not the same as the field.

More then – the field is greater then the value. *(Important Note: In dealing with characters, B is greater then A and A is greater then 99, also 99 is greater then 98. However 99 is also greater then 888 because only the one character of the string is compared at a time, hence 9 is greater then 8. Numbers are compared normally where 2 is greater then 1 and 10 is greater then 9)*

Less then – the field is less then the value, characters are deal with the same way as the more then operator.

Is blank – looks for a blank in the field.

Is NULL – looks for a null value in the field.

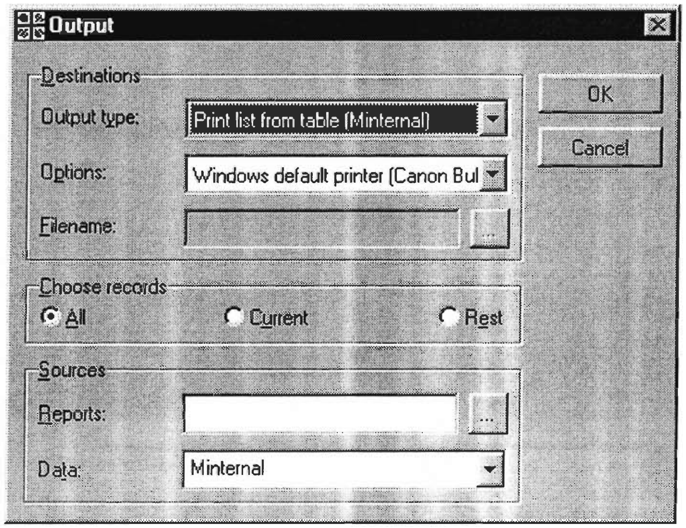
Contains – as long as the value is somewhere in the field. Ex: looking for a English name containing a 'A' will result you with all the English names that have a 'A' in the spelling of the entire name.

In – Field is in the value.

Between – Field is between the values.

Print

Also on the navigation bar is the print button. The printing function gives you the option to print the list of records out for easier reference. The print ability has a couple types of output that you can choose from. Like the search ability, you will print the current data you are dealing with.



Destinations

Output type: Print list from table (Minternal)

Options: Windows default printer (Canon Bul

Filename:

Choose records

● All ○ Current ○ Rest

Sources

Reports:

Data: Minternal

Just click 'OK' on the above screen and the print process is up to the printer.

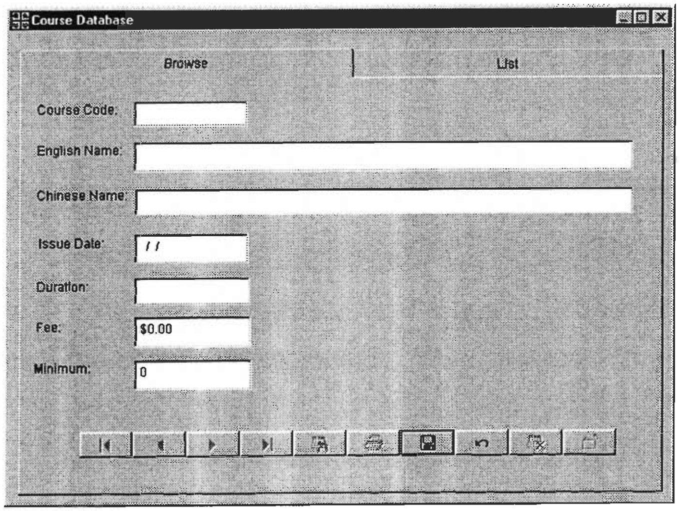
Record Modification Tools

The seventh, eighth and ninth buttons on the navigation bar give you the ability to modify the database and the records inside them. These are powerful tools in that they can change the database in an instant. These tools are not recommended for regular use, however there maybe times when these tools can come handy saving time and effort to import and export Excel files. *(Important Note: The delete record tool is the only way to delete records from the database. The database will not be deleted if an empty Excel file is imported because importation does not delete record, it only append new record on and update older records.)*

Add Record

Adding records will disable the read only function of the browser window and changes it to an editor. Each field would be empty out, after you fill in the information, you may click add record again and the record would then be saved into the database. You may also not want to add the record after you have chosen to go into the Add Record screen. Simply click the revert button next to the Add Record button and you will be brought back to the browser.

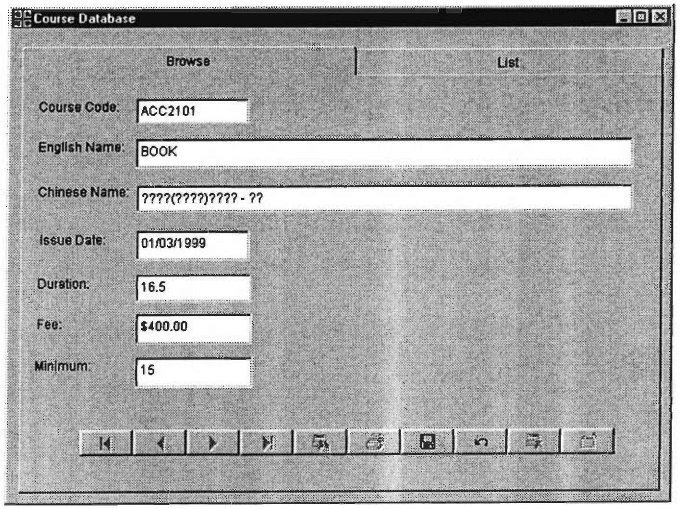
Add Record Window:



Course Code:
English Name:
Chinese Name:
Issue Date: / /
Duration:
Fee: $0.00
Minimum: 0

Edit Record

Edit Record allows for the editing of the current record. All fields will no longer be read only. After editing, make sure you save the record. If you do not want to save the record you have edited, once again hit the revert button.

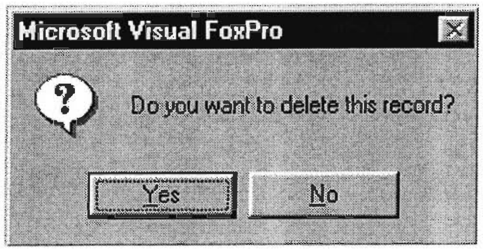


Course Code: ACC2101
English Name: BOOK
Chinese Name: ????(????)????- ??
Issue Date: 01/03/1999
Duration: 16.5
Fee: $400.00
Minimum: 15

As you might notice in this edit screen that the English name has been edited and erase to only having the work book instead of bookkeeping.

Delete Record

It is very important not to delete the wrong records. The data delete cannot be retrieved from the database anymore. It is highly recommended that the record be printed out or exported before any actual deletion is made. Unless you are sure that the record will no long be of any use to you.



Do you want to delete this record?

After you have decided to delete the record, a prompt asking if you are sure will pop up. Think this over and make sure the record is no longer needed and click 'Yes' to delete a record.

Other Notes on Usage of the Database Application

Backup Database

1. It is highly recommended that the database be backed up frequently. Back ups should be made periodically as well as before any deletion and editing of the database. Before any importation of new files, the older records should be exported to have something for future reference. The new data will over write all of the older data so the older data will no longer be existent in the database. Back ups of data can easily be made with the export function so that there will be an Excel file of all the data to keep track of things. The database can also be easily printed out to keep a paper copy of all records. A print out of the three views should include all the data available in the database. If you not only want to back up information but would also like the entire database to be backed up in case the system crashes then it you would need to copy the entire data folder into another directory or to be safer, to another computer.

In the directory that the application is installed, there is a folder named 'data' that contains all the tables that form the database. The default location of this folder is:
C:\CAHES\Database Application\data
To back up the entire database, this folder would need to be saved onto a safe location.

1. Right click on the folder 'data' and choose 'copy'
2. Find the location you would like to save this folder either on your computer or on a networked computer.
3. Right click on the destination and choose 'paste.' A copy of the data should then be transferred to the destined location.

This should be done periodically to ensure that a copy of the database exists. If for any reason an uninstall of the application is needed then you can replace the data folder with a back up copy and the material in the database should be preserved with no problems.

Uninstall

If the application is no longer needed, it needs to be uninstalled through the Add and Remove section of the control panel. The folder cannot be deleted with out an official uninstall because system files have been changed in the process of installation.

Contact

If there is any technical difficulties or questions about the application, please email:

Email Address:
Cuong Diep – cdiep@wpi.edu
De Ming Guan – deguan@wpi.edu
Victor Chen – vchen@wpi.edu

Address:

Worcester Polytechnic Institute
100 Institute Road
Worcester, MA 01609
U.S.A.

Appendix J **Internal Excel Template Guideline**

Field Name	Description	Options	Sample Data
Course Code	This code can consist of letters and numbers. Academic Office assigns course Code. It is a unique code that defines every course.	N/A	ACC2101
English Name	To specify Course's English Name	N/A	BOOKKEEPING (LCCI FIRST LEVEL) TOPIC REVIEW
Chinese Name	To specify Course's Chinese Name	N/A	□□□□(□□□□)□□□□
Course Type	N/A	N/A	N/A
Type	N/A	N/A	N/A
Make Class	N/A	N/A	N/A
Course Fee	To specify the fee of this course in Hong Kong Dollars	N/A	400
Duration (hours)	To specify the total hours for this course	N/A	16.5
Issue Date	To specify the date when course was developed. Date is in DD/MM/YYYY format. (D stands for Day, M stands for Month and Y stands for Year)	N/A	21/7/2000
Minimum	To specify the number of enrolled students required before this course can be offered.	N/A	20

Government Subsidy	To specify whether this course receives government subsidy. Enter 1 for Yes and 2 for No. Default Value: 2 for No.	1; 2	1
Key Area of Study	To specify the field of study of this course according to the system of the EMB.	See EMB User Manual Appendix III	AD02-Introduction and Fundamental System Operation
Mode of Study	To specify the mode of study of this course. Default Value: 2 (part-time)	(Please use the code given below and give only one answer) 1. Full-time 2. Part-time 3. Distance Learning with classroom tutorial/lecturing 4. Pure Correspondence course 5. On-line Internet/Video course 6. Self-study 7. Seminar/Workshop 8. Others (Please specify)	1
Primary Mode of Instruction	To specify the mode of instruction primarily used in this course. Default Value: 1 (Lecturing)	(Please use the code given below and give only one answer) 1. Lecturing 2. Tutorial 3. Internet 4. Outdoor Activity 5. Group Project 6. Correspondence 7. Video/Cassette/CD-ROM 8. Practice 9. Placement 10. Others (Please specify)	1

Other Mode of Instruction	To specify the other mode of instruction used in this course (if any).	(Please use the code given below and give only one answer) 1. Lecturing 2. Tutorial 3. Internet 4. Outdoor Activity 5. Group Project 6. Correspondence 7. Video/Cassette/CD-ROM 8. Practice 9. Placement 10. Others (Please specify)	2
Major Medium of Instruction	To specify the major medium of instruction of this course.	(Please use the code given below and give only one answer) 1. Cantonese 2. Putonghua 3. English 4. Japanese 5. French 6. German 7. Others (Please specify)	1
Other Medium of Instruction	To specify the other medium of instruction of this course.	(Please use the code given below and give only one answer) 1. Cantonese 2. Putonghua 3. English 4. Japanese 5. French 6. German 7. Others (Please specify)	2

Caritas – Hong Kong Database

Nature of The Course	To specify the nature of the course with respect to academic course.	(Please use the code given below and give only one answer) 1. Academic 2. Professional 3. Vocational/Technical 4. Other continuing education courses	1
Expected Duration (Year)	Expected duration for the completion of the course in number of years.	N/A	1
Expected Duration (Month)	Expected duration for the completion of the course in number of months	N/A	3
Expected Duration (Week)	Expected duration for the completion of the course in number of weeks	N/A	4
Expected Duration (Day)	Expected duration for the completion of the course in number of days	N/A	16
Delete Date for the EMB Site	To specify the expiry date of this course in DD/MM/YYYY format. (D stands for Day, M stands for Month and Y stands for Year)	N/A	15/9/01
Entry Qualification	To specify the entry qualification of this course.	(Please use the code given below and give only one answer) 1. Primary (P1 - P6) 2. Junior Secondary (S1 - S3) 3. Senior Secondary (S4 - S5) 4. Sixth Form (or Matriculation)	3

		5. Craft 6. Technician 7. Sub-degree 8. First degree 9. Postgraduate 10. No specific requirement	
Registered/Exempted	To specify whether this course is registered or exempted under Non-Local Higher and Professional Education (Regulation) Ordinance	(Please use the code given below and give only one answer) 1. Registered 2. Exempted 3. Not applicable	1
Award-bearing	To specify whether the course is award-bearing	(Please use the code given below and give only one answer) 1. Yes 2. No	1

Qualification Conferred	To specify the qualification to be conferred upon completion of this course.
	Default Value: Certificate
		Primary level;
		Secondary three level;
		Secondary five level;
		Secondary Six;
		Craft;
		Technician;
		Sub-Degree;
		Postgraduate;
		Attendance certificate;
		Professional qualification;
		Certificate;
		Diploma;
		Not Specified
			Certificate

Institution Code	To specify the reference number assigned to the institution by the EMB.	N/A	0066
Name of Co-Organizer (Eng)	To specify the English name of the co-organizer of this course.	N/A	ABC Training Center
Name of Co-Organizer (Chi)	To specify the Chinese name of the co-organizer of this course.	N/A	ABC□□
Institute Awarding Qualification (Eng)	Please give the English name of institute/organization.	N/A	123 Center
Institute Awarding Qualification (Chi)	Please give the Chinese name of institute/organization.	N/A	123 □□
Qualification Recognized By	Please give the name of institute/organization.	(Please use the code given below and) 1. Hong Kong SAR Government 2. Professional Bodies 3. Local Tertiary Institutions 4. Non-local Tertiary Institutions 5. Accreditation Bodies (e.g. Hong Kong Council for Academic Accreditation (HKCAA)) 6. Employers 7. Others (Please specify) 8. None	1

Appendix K

School Excel Template Guideline

Part A: School Info

Field Name	Description	Options	Simple Data
English Name	Specify the school's English Name.	N/A	Caritas Francis Hsu Evening College
Chinese Name	Specify the school's Chinese Name.	N/A	□□□□□□□□□
Address (English)	Specify the school's address in English.	N/A	11 Caine Road Hong Kong
Address (Chinese)	Specify the school's address in English.	N/A	□□□□11□
Contact Person (Eng)	Specify contact person's English name.	N/A	Mr K.K. Lo
Contact Person (Chin)	Specify contact person's Chinese name.	N/A	K. K. □□□
Contact Telephone #	Specify contact person's telephone Number	N/A	28434651
Fax Number	Specify contact person's fax Number	N/A	25240659
Email Address	Specify the school's email address. (Required, can not leave this field blank)	N/A	caps@caritas.edu.hk
Web-Site address	Specify the school's web site address. Default Value: www.caritas.edu.hk	N/A	www.cfhc.caritas.edu.hk

Part B: Class Info

Field Name	Description	Option	Sample Data
Course Code	To specify the course code.	N/A	ACC3070
Class Section	To specify the section for a particular course.	N/A	A
Enrollment Date	To specify the enrollment date in DD/MM/YYYY format. D stands for Day. M stands for Month. Y stands for year.	N/A	28/8/2002
Start Date	To specify the start date for this course in DD/MM/YYYY format. D stands for Day. M stands for Month. Y stands for year.	N/A	01/09/2002
End Date	To specify the start date for this course in DD/MM/YYYY format. D stands for Day. M stands for Month. Y stands for year.	N/A	22/10/2002
Enrollment Capacity	To specify the maximum number of students can enroll for this class	N/A	30
Day/Time	To specify the day and time when the class meets	Day and time when class meets. M - Monday T - Tuesday W - Wednesday R - Thursday F - Friday S - Saturday N - Sunday e.g. MT/9:00am - 12:00pm	MW/1:00PM-2:00PM

		means class meets Mondays and Tuesday from 9am to 12 pm MR/10:00am-11:00am, W/1:00pm-3:00pm means class meets Mondays and Thursday from 10 to 11 am, also Wednesdays from 1 to 3pm.	
Last Round	To specify the number of students enrolled last round when this course was offered	N/A	21
Next start Date	The earliest start date for the next round of this course in DD/MM/YYYY format.	N/A	24/5/2002
Number Of Classes	Number of class meetings through out the course.	N/A	16