CSC340S ASSIGNMENT 3

Information Systems Analysis and Design

Names: Lie A Cheong, Ching Ching Lorna(990054203)
Kenneth Ho Kin Leung (980968140)
Ching Wah Janice Lo (990432858)

TA: Attila Barta
Prof: : John Mylopoulos
Date: Apr 12th, 2002

Table of Contents

Feasibility Study

Appendices

Introduction

Bestley Enterprises Co. Ltd (Bestley in short) is a company producing plastic bags, which located in Scarborough. Bestley mainly centralized in producing and printing logo on finished plastic bag.

The goal of this design report is to decide upon the proper hardware, software, database and interface design required to implement a new information system for Bestley. Decisions made in this report is based on the previously study of the functional and non-functional requirements of the company.

Company Development History

Bestley founded in 1993 by Mr. Allen.L.H Tang (currently CEO). The company was first a family owned plastic bag company, which had only two employees, Mr. and Mrs. Tang. At first, the company only responsible for production of more popular type of plastic bag and act as a middleman between clients and printing company. In 1995, the company expanded as a result of market pressure. The expansion included employed more workers and bought more up-to-date machinery.

Mr. Tang hopes to capture more of market share in the near future. Therefore, computerization is a must in the current business environment. The implementation of information system is essential in order to increase efficiency of data transfer and overall performance of company.

The current information system

To determine problems with the current system of Bestley, understanding the business routine of the company is important.

The daily routine:

- The customer representative meets the customer to figure out their requirement. Customer's contact information is recorded into the "Customers Contact Ledger"
- The owner designs a few patterns according to the customer's requirement. He then provides samples and materials for the customer to choose. The design and ideas are kept in the "Ideas & Design Ledger"
- The customer makes a final decision of the outlook of their products and sign up an order invoice. The invoice is then kept in the "Customers Invoice Ledger".
- The secretary orders raw materials from the supplier and records this transaction in the "Raw Material Order Ledger".
- Once the materials arrive, production process can be started. Workers put a proper amount of raw materials into the extruder machine and generate the plastic rolls. The amount of raw material used is recorded into the "Raw Material Order Ledger".
- Based on the description in the "Ideas & Design Ledger", workers start to print the specified pattern on the plastic rolls.

- The plastic rolls are then cut into pieces of plastic bags and packed into boxes.
- Driver delivers the finished products to customer.
- The accountant collects the account receivable from customers and makes a stamp on the customers invoice to indicate the transaction have been completed and paid.

The monthly routine:

- The accountant prepares the balance sheet, trial balance and income statement. These data are stored into the "Accounting Ledger"
- The secretary organizes the data in the "GST Record Ledger" and prepares monthly sales report/chart for the owner to determine future business plans.

Summary of the current information system:

From the description above, it is known that the 6-Ledgers play a very important role throughout the entire business operation. These 6-Ledgers are the fundamental information system for Bestley. Each ledger records different type of information and data. 4 of them record the information for all customers and suppliers. 2 of them are used for internal accounting purpose. Detail descriptions for each of them are list below:

Ledger for Customers and Suppliers:

1. Customers Contact Ledger – It contains records of all customers' contact information, such as company address, telephone number, email, etc. Each customer record consists of a list of invoice numbers, which represent all the previous invoices made by this customer.

2. Ideas & Design Ledger – It contains records of all plastic bag pattern design, as well as customers' special requirements on their products. The designs are produced using CorelDraw 9. Often, one hard copy is kept inside the ledger and one soft copy is kept on computer. Each record has a number, which is used in the invoice to identify the design of product.

3. Customers Invoice Ledger – It contains records of all transaction with customers. It keeps information such as the amount of customer's order (i.e. number of plastic bags), the date for shipment and the price for each transaction. Each transaction has a unique invoice number.

4. Raw Materials Order Ledger – It contains records of all transaction made with the suppliers. It also keeps track of the quantity of raw materials in inventory and the quantity used in production.

Ledger for internal accounting purpose:

5. Accounting Ledger – This ledger information is prepared monthly. At the end of each month, the accountant will gather all invoices from Customers Invoice Ledger and Raw Materials Order Ledger, and then prepares monthly balance sheet, income statement and trial balance for the company.

6. GST Ledger – This ledger contains all monthly income statements. They are used to calculate the GST of the company in order to report to the government every 3 months.

Many operations of Bestley depend highly on the flow of the information within these 6-ledgers. Failure in retrieving or transforming these data will result to inefficient, inconsistency and inaccuracy production processes. It could possibly provide misleading information for future forecast.

The Proposed Solutions

Among all the alternatives, this is the best alternative to correct problems in the current information system.

Proposed Alternative: E-Commerce + Computerization Analysis

Description

As the owner of Bestley intends to capture a largest market share. Therefore, the best way to attain this goal is to combine e-commerce and computerization.

Basically, the old 6-ledgers system will be replaced by a new computer system. The new system consists of:

A new PC and a customized software application that is designed specifically to suit the company business routine. Compare to the old system that required separating a diversity of data into 6 different ledgers, the new system can organize all the business data into a computer directly through a user-friendly interface.
Buy a new internal CD writer.
Buy two PDA.
The whole information system will protect by a password system. Each employee will be assigned a password to login the system.
Basic computer training is provided to some employees.
Internet and Intranet access for each terminal.
Each terminal is connected and work as a network.
Company's homepage and Online Processing system is provided.

Analysis of new system

1. Bestley's homepage on the web

This can attract new customer and attain the goal of increase market share.

2. Online Ordering

Customer can retrieve information on raw materials, design of plastic bags, payment methods and place an order on the web.

Now, the employees spend less time to explain and meet the clients. The clients can place their orders on the web. Therefore, more transaction can be made when compare to the past paper-based information system.

3. Retrieve data and print statements made simple

Employees can retrieve all the data they want in several minutes through newly developed software and centralized database system.

The efficieny and accuracy can be greatly increased with new system.

4. Reduce the time spend on backup the data

Everything in database will store in Compact Discs instead of making copies of 6 ledgers. This greatly increase efficiency and reduce the time spend on photocopying. Besides, a smaller space is need to store Compact Discs. With the help of computer, we can make several backup copies if we want and store them in different places. Therefore, this can safeguard against data loss.

Recommendation

Based on the above comparison table, obviously computerization and e-commerce combination is the best choice. The company's plan is to capture more market share in the long run and catch up with current technology level.

The four main steps in completing the described Computerization and e-commerce

- Set-up and design stage that maximize the end-user performance
Survey can be given out to end-user to find out the common operations and needs.

- Installation of the proposed hardware which meet the minimum requirement
Further price searching for new terminals are needed.

- Recruit programmers to develop the proposed information system
The details please refer to alternative solution of computerization and e-commerce.

- Introduction of new system to all the employees
Documentation of new system or courses can be given to employees.

Global Architecture

In this section, we will propose the design of the new information system for Bestley. We will particularly focusing in the hardware, networking and software platform used in the new system. Beside this, we will also have a clear description and justification on the software architecture that we will adopt.

Hardware Selection

The hardware required for the new information system includes an information storage server and 5 microcomputers. The 5 microcomputers are corresponded to different users. One for the director, one for the assistant director, one for the secretary, one for the accountant and one for the technician.

- Microcomputers
    - Intel 845 chipsets with Pentium IV processor.
    - 256 KB L2 Advanced Transfer Cache available with speeds 400 MHz.
    - 128 MB of Random Access Memory
    - 12 GB hard disk storage
    - Featuring the new Intel NetBurst™ microarchitecture.
    - Streaming SIMD Extensions 2 (SSE2) Instructions.
    - Intel MMX™ media enhancement technology.
- Information Storage Server
    - Intel 850 chipsets with Pentium IV processor.
    - 512 KB L2 Advanced Transfer Cache available with speeds 2.40 GHz.
    - 256 MB of Random Access Memory
    - 40 GB hard disk storage
    - Featuring the new Intel NetBurst™ microarchitecture.
    - Fully compatible with existing Intel Architecture-based software.
    - Streaming SIMD Extensions 2 (SSE2) Instructions.
    - Intel MMX™ media enhancement technology.
    - Support for uni-processor designs.

The reason for selecting a Pentium IV processor as the server rather than using a mainframe:
- The cost are much lower; Pentium IV cost only $2000 compare to $16,000 of the mainframe.
- Since Bestley is only a small business with only 10 workers. The use of mainframe is a waste of resources.

Network Selection

Networking is a major component of almost every modern computer system. Since Bestley's new information system allows E-commerce transactions, so our networking strategy must focus in two main areas, the internal information transfer and the external information transfer.

- Internal information transfer

- Since the previous information system is not computerized and has no E-commerce transaction, so there is no networking at all. For the new system, we have installed 5 microcomputers and a server. In order to connect all 5 microcomputers and the server together, we need to set up a Local Area Network (LAN) in the main office, so that those 5 microcomputers can access the data from the server and can sharing data between each other.
- The LAN is built by using a 10/100 kbps Ethernet (copper) connection to the server through shared hub.

- External information transfer

    - Since Bestley has a homepage for online ordering, so there is necessary to have a Wide Area Network (WAN) or Internet set up. This can be accomplished by setting up a cable connection with Rogers AT&T.

Software Platform Selection

The design of the software platform includes the selection of an operating system and any other commercial software that will be used for a company. To make selection on the operating system and the required software for Bestley, we must first understand the daily business routine for Bestley. (This has been previously mentioned.)

- The Operating System

    - Selecting a suitable operating system is very important in creating an efficient computer system. The most important factors that we need to concern are how well the operating system can perform in data management and how well the employees can accommodate to the new operating system interface. Since most of Bestley's employees are familiar with the Microsoft Operating System, so our analysis will focus on four MS operating system: Windows 2000, Windows ME, Windows NT 4.0, and Windows XP.

Operating System:	Windows 2000	Windows ME	Windows NT 4.0	Windows XP
System Requirements: (Recommended)	- 133 MHz + - 128 MB + Memory - 1.0 GB + free space	- 150 MHz + - 32 MB + Memory - 480 MB + free space	- Any Pentium Processor - 32 MB + Memory - 110 MB + free space	- 300 MHz + - 128 MB + Memory - 1.5 GB + free space
Built in networking tools:	None	None	Yes	Yes
Stability: (range: 1 – 4	3	4	2	1

1 means the least crash and 4 means crash very often)				
Security:	- normal	- normal	- very high	- high
Level of employees technical skills required:	- low level	- low level	- high level	- low level
Price:	$219 USD	$209 USD	$409 USD	$299 USD
Overall ranking: (Base on the suitability to Bestley)	**4**	**3**	**2**	**1**

- Based on the above analysis chart and the comparison pie chart[2], the most suitable operating system for Bestley is Windows XP.

- Others Required Software

   There are several software that are fitted to Bestley's daily business rountine:

   - Microsoft Office 2000
   - Corel Draw 9
   - Internet Explorer
   - Netscape
   - Norton virus

Software Architecture Selection

A software architecture defines the components of a software system and how they use each other's functionality and data. Indeed, the Bestley information system can be explored by 3 architecture styles: The layer, client-server, and MVC. Before exploring these architecture styles, it is better to divide the whole system into several subsystems first. Dividing a system into several subsystems can help the company to have a better-managed software development. It can also help to improved reuse potential, portability and easier maintenance.

- The subsystems

   The Bestley information system can divide into 4 subsystems or major components according to their daily business routine. They are the Database, Accounting, Administration and Backup. Below is the detail break down for the subsystems.

_____

[2] Pl...... ....lly D......
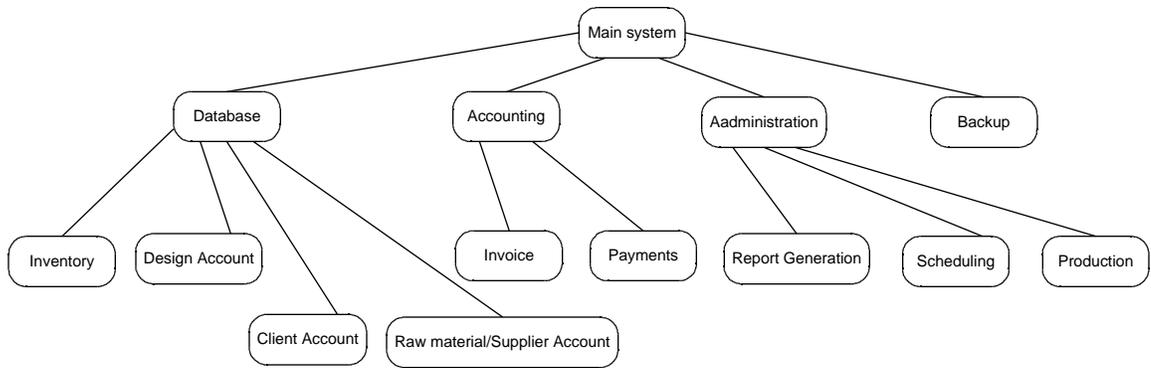
Figure – the subsystems for Bestley

- Open Four Layered Architecture

   Bestley subsystems can be classified as an open layered architecture. The main idea of this open layered architecture is that a layer can directly access any lower layer service without the need for extra program code to pass messages through each intervening layer.

Backup	layer 1
Administration	layer 2
Accounting	layer 3
Database	layer 4
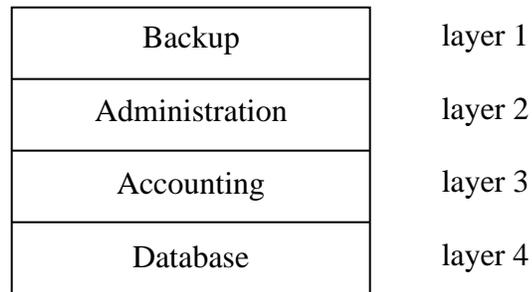
Figure -The open 4-layered architecture for Bestley

- The Client Server Architecture

   Since Bestley is having a web page that allows customer to order their products online, so we can explore this business process by the client server architecture.
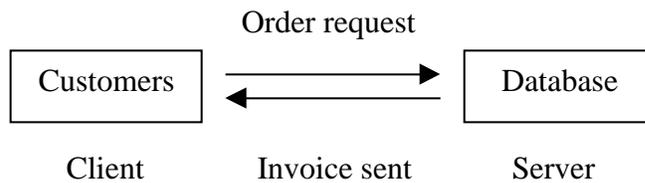
Figure- The client server architecture for Bestley

It actually work like this: when the customer want to order online, he first fills up the online order form. Then, the order request will send to the server of Bestley. The sever will automatically updated the database of the server indicating there is online request and employees will base on the information of the order and issue an invoice back to the customers.

- The Model View Controller (MVC) Architecture

To explore Bestley software system by using the MVC architecture, it must first classify its subsystems into Model, View, and Controller.

Model - the subsystem that classified as model should provide the main functionality of the application and is aware of each of its dependent view and controller components.

- in the Bestley case, the Database component is the Model.

View - each view corresponds to particular style and format of presentation of information to the use.

- in the Bestley case, the Administration component is the View.

Controller - accept user input in the form of events that trigger the execution of operation with the model.

- in the Bestley case, the Accounting component is the Controller.

Below is the responsibility of MVC components of Bestley:
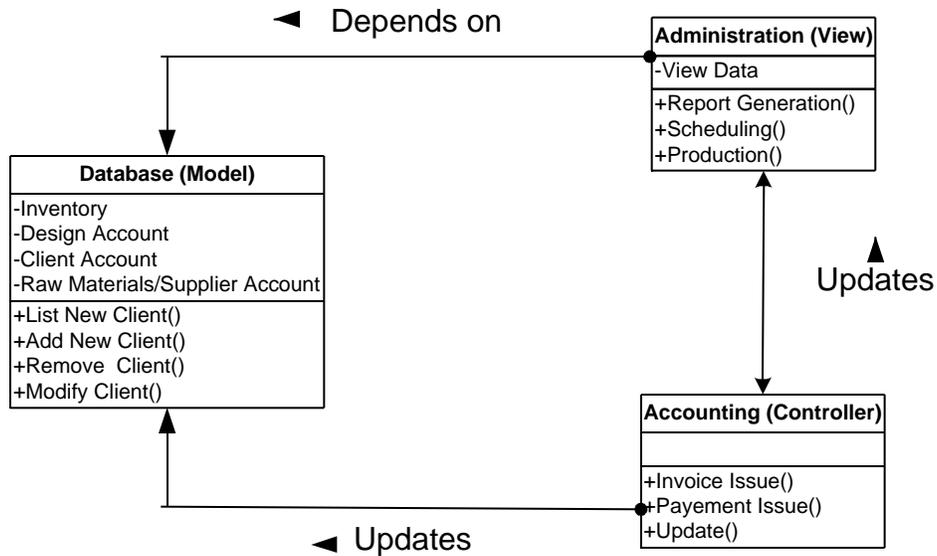
- As we can see, the Administration (View) totally depends on the Database (Model). It retrieves data from the model and updates its presentation when data has been changed. It also depends on the retrieves data to plan for production control and schedule planning. On the other hand, the Accounting (Controller) retrieves data from the Database (Model) and prepare for necessarily documents and sent to the Administration for admin purpose. Whenever a job has been finished (e.g: a product have been sold and money was received), then the Accounting (Controller) will collect all necessarily change and make update to the Database (Model) and the Administration (View).

Overall design meets all requirements

With the new software system, Bestley no longer uses 6 ledgers to keep records. Instead, it uses databases to store different kind of records. They are the client, raw material, design, and inventory databases. Secretary updates and add new record to these databases by simply logging into the database and perform such operations. In addition, data are backup monthly by writing them to Compact Disc. This saves a lot of time in photocopying the ledgers. Accountant accesses the transactions database monthly to generate sales report and balance report.

With the E-commerce facility, part-time web professional is hired to update the web site and information about products constantly. Bestley's employee logs on to the e-commerce system daily to check for new orders. If there is new order, the employee processes the order immediately by sending an invoice back to the customer and give all necessarily information to the administration for production planning and scheduling.

Program Design

Class Diagram

Description of class diagram

This class diagram shows the overall structure of Bestley's information system. Each class represents an object in the environment. There are main 3 different types of object in the class diagram, people, database and record. People in the system will facilitate the data transfer within the system. Each record contains certain kind of information, e.g. raw material or client information. Database acts like a container, which hold records.

Description derived from the data dictionary

Activity Record
This class is a specialization of Record class, which contains all the activities involved of each login session.

Backup Device
This is a superclass of CD Writer.

CD Writer
This class is a specialization of Backup Device, which contains functions for backup operations.

Client
Client class represents a customer in this system. This class contains all the information about a client.

Client Record
This class is a specialization of the Record class, which inherits the Client class.

Database
Database stores all the kinds of records.

Design Record

This class is a specialization of the Record class, which contains all the ideas and design of plastic bag.

Employee
Employee class represents an employee in this system. This class contains all the information about an employee.

Employee Record
This class is a specialization of the Record class, which inherits Employee class.

Inventory Record
This class is a specialization of the Record class, which stores all the information about inventory of each type of raw material.

Invoice

This class is a specialization of the Record class, which stores all the information about each transaction.

Login Record
This class is a specialization of the Record class, which contains login and logout information of each employee.

Online Order
This class contains all the information of online order.

Payment
This class is a specialization of the Record class, which contains all the information about payment made by each client.

Raw Material Record
This class is a specialization of the Record class, which contains all the information about each type of raw material.

Software Interface
This class represents the application or software that employees used to input data to database.

Supplier Record
This class is a specialization of the Record class, which contains information about supplier of raw materials.

Transaction Record
This class is a specialization of the Record class, which contains all the information of each transaction on invoice.

Justification that the design meets relevant requirements

The class diagram could meet the requirements since the class can be divided into 3 subsystems. The three subsystems are transaction, accounting and e-commerce. Transaction and accounting sub-system can be clearly shown on the class diagram, e.g. client and transaction record, etc. However, e-commerce sub systems can't be clearly shown in the class diagram because all the components of e-commerce can be found in the transaction and accounting sub-system.

Sequence diagram
Description of sequence diagram

The five sequence diagrams are corresponds with the major processes of our suggested system. Please see the Sequence diagrams attached in Appendix

1.Client upload order of product

When a client attempts to create an order online, an object of client is created. This object is then used to call function create() to create an order. The web

site provides a form and prompts the client to select product. After the client has selected the product, he/she proceeds to enter payment information and method. Then the web will display a confirmation to show the selected product, shipment time and total cost. The client selects submit and the order is uploaded.

The diagram shows the use of web order form as interface to purchase product.

2.Employee download online order of product

An Employee object is created. With the ID and password of this object, login function is called to get access to the E-commerce system for this employee. If the login is valid, the system will then automatically display all the new online orders. The employee object is then called the function storeRecord(invoiceid) to store various information of each new order.

The diagram shows the use of the back-end database of the web site to download information of orders.

3.Backup

An employee login to a particular database and get the last update time by the get_Last_Backup() function. If the time returned is longer than one month, it proceeds to access the records in the database and write them to CD. When the backup CD is written, the employee sets the current date to the database to record this backup process.

The diagram shows the use of back up device.

Create Sales Report

An object of employee is created for Accountant. Using the accountant's ID and password to login, it is then able to query the Transaction database and group required records together. Function createTable() is then called to form table for these grouped records. Lastly, the report containing the table is printed out.

4.Report usage

An employee object is created for the C.E.O Mr. Tang. This object is used to request viewing the sales, customer and product report.

The diagram shows the use of transaction and account applications subsystem to query the stored records and generate reports.

5.Store transaction to databases

The employee login to the Transaction database and create a new Transaction object. This object is used to call searchClient(clientid) to access the client database and obtain the client's information; call searchProduct(deisgnid) to receive the product information from the design database; call searchMaterial(rawid) to check if the required material is available and call searchShipmentdate() to get the date of shipment. After all of the information about this transaction is obtained, an invoice about this transaction can be printed.

The diagram shows the use of interfaces of various databases to create a transaction record and a print of invoice.

State Diagram
Description of state diagram

Please see the Sequence diagrams attached in Appendix

Class database
The database first recognize user identity, if password is correct then wait for user to select activities. For every activity, the database will check whether the file is already existed. If it is not existed, create a new record and fill up information. If file existed, modify the file and finally store the records into the database.

Class online-order
First, the client will paste a new order, then the system will send back a signal ask for product selection. The system will wait until client selects their product. Then the system will again send a signal to alert client to choose for a payment method. Until client selects a payment method, the online order is completed and a confirmation form is sent back to the client notify the transaction has been finished.

Class client record
When employee gets an order from client, they will first check whether this particular client record is in the database or not. If no, create new client record; if yes, modify the exist one. After all information has been filled up or gets modified, the process completed and stores the new information into the database.

Class Employee record
When employee wants to input some employees' information into the database, they will first check whether this particular employee record is in the database or not. If no, create new employee record; if yes, modify the exist one. After all information has been filled up or gets modified, the process completed and stores the new information into the database.

Class backup device
The backup system first verifies the employee's identity, if the password is correct then the employee can access into the database. The system will call the CD writer to

start copying data from the database into a CD. If the backup success, the process is completed.

Class design record
After employee created a design draft and wants to enter the information into the database, the system will first check whether this particular design record is in the database or not. If no, create new design record; if yes, modify the exist one. After all information has been filled up or gets modified, the process completed and stores the new information into the database.

Class material record
When employee buys raw material, they need to enter this information into the database. The system will first check whether this particular raw material record is in the database or not. If no, create a new raw material record; if yes, modify the exist one. After all information has been filled up or gets modified, the process completed and stores the new information into the database.

Class supplier record
When employee made order from the supplier, this information has to enter into the database. The system will first check whether this particular supplier record is in the database or not. If no, create new employee record; if yes, modify the exist one. After all information has been filled up or gets modified, the process completed and stores the new information into the database.

Class inventory record
This is a concurrent state diagram. The initiate state will be start from employee buys raw material or employee uses up raw materials. After both situations, the employee needs to perform a quantity count of inventory on hand. Then he will update the latest quantity into the database and finish the process.

Database Diagrams

DATABASE DESIGN

Data that required to stored in the system.
- Employee records
- Client records
- Inventory records
- Transaction records
- Design records
- Raw material records
- Supplier records
- Design records
- Activity records
- Order invoices
- Payment statements
- WebLogin record
- Online order

The Entity-Relationship model is a data model that describes the data requirements for a new information system. The database classes, relationships and the associated identifiers are listed below to prepare for the construction of the ER model.

Entity	
Object	Identifiers
Employee	employeeID
Client	clientID, name
Supplier	supplierID, name
Inventory	itemID, quantity
Raw Material	RawID
Design	DesignID
Online Order	OrderID
Transaction	tranID, paymentID, designID, rawID

Relationship	
Object	Identifier(s)
Activity	activityCode,
WebLogin	loginID, sessionID
Payment	paymentID, clientID, invoiceID
Invoice	invoiceID, invoiceDate, clientID

Workload Data
The following table represents the number of occurrences of each class in the database

Concept	Type	Volume	Description
Employee	E	100	Number of past and current full/part time workers
Client	E	500	Number of past and current clients
Inventory	E	30	Actual items in stock
Transaction	E	12000	Past and current
Online order	E	1000	New
Design	E	350	All design records
Raw material	E	50	All raw material records
Supplier	E	30	All past and current suppliers
Activity	R	250	All activity records for 3 months
Invoices	R	15000	Number of statements
Payment	R	12000	Number of sales
WebLogin	R	90	Number of logins for 3 months

The operations that are performed in the database are summarized below.

Type of Operations

#	Operation	Type	Frequency per month
1	Modify quantity for inventory item	B	20
2	Generate an online order	I	30
3	Generate payments statements	B	20
4	Generate invoice	B	50

5	Gather transactions for accountant	I	1
6	Add an entry log for an activity	I	30
7	Perform Backup	I	1

In order to restructure the ER schema, we need to optimize two parameters. They are Cost of an operation and Storage Requirements.

Storage Requirements
It evaluates in terms of the number of bytes necessary to store the data described by the schema

Cost Analysis
It evaluates in terms of the number of occurrences of entities and relationships that are visited during the execution of an operation on the database.

Operation 1 Cost Analysis: Modify quantity for inventory item
Sequence of visits: Employee accesses an inventory record and modify it.

Without Redundancy

Object	Type	Number of Accesses	Access Type
Manager	E	1	R
Access Record	R	1	R
Record	E	1	R
Inventory record	E	>1	RW

With Redundancy

Object	Type	Number of Accesses	Access Type
Manager	E	1	R
Access Record	R	1	R
Inventory record	E	>1	RW

Operation 2 Cost Analysis: Generate an online order
Sequence of visits:
Employee login to the e-commerce system, gather the online order information, create an order object then retrieve client, product, and raw material information and create a transaction.

Without Redundancy

Object	Type	Number of Accesses	Access Type
ITworker	E	1	R
Login	R	1	R
Login record	E	1	W
Receives order	R	1	R
Online order	E	>1	R
Access Records	R	>1	R
Record	E	1	R
Client record	E	1	R
Record	E	1	R
Raw material record	E	1	R

Object	Type	Number of Accesses	Access Type
Record	E	1	R
Design record	E	1	R
Insert new record	R	1	R
Record	E	1	R
Transaction	E	1	W

With Redundancy

Object	Type	Number of Accesses	Access Type
ITworker	E	1	R
Login	R	1	R
Login record	E	1	W
Receives order	R	1	R
Online order	E	>1	R
Access Records	R	>1	R
Client record	E	1	R
Raw material record	E	1	R
Design record	E	1	R
Insert new record	R	1	R
Transaction	R	1	W

Operation 3 Cost Analysis: Generate Payment statement

Sequence of visits: Retrieve client information and create a payment object to store all payment information.

Without Redundancy

Object	Type	Number of Accesses	Access Type
Manager	E	1	R
Access Record	R	1	R
Record	E	1	R
Client record	E	1	R
Insert new record	R	1	R
Record	E	1	R
Payment	E	1	W

With Redundancy

Object	Type	Number of Accesses	Access Type
Manager	E	1	R
Access Record	R	1	R
Client record	E	1	R
Insert new record	R	1	R
Payment	E	1	R

Operation 4 Cost Analysis: Generate invoice

Sequence of visits: Retrieve the payment and transaction information and create an invoice

Without Redundancy

Object	Type	Number of Accesses	Access Type
Manager	E	1	R

Access Record	R	1	R
Record	E	1	R
Transaction record	E	1	R
Record	E	1	R
Payment	E	1	R
Insert new record	R	1	R
Record	E	1	R
Invoice	E	1	W

With Redundancy

Object	Type	Number of Accesses	Access Type
Manager	E	1	R
Access Record	R	1	R
Transaction record	E	1	R
Makes record	R	1	R
Invoice	E	1	W

Operation 5 Cost Analysis: Create table of monthly transactions for accountant

Sequence of Visit: Searches all transaction records and gather those on the same current month and create a table.

Without Redundancy

Object	Type	Number of Accesses	Access Type
Worker	E	1	R
Access Record	R	1	R
Record	E	1	R
Transaction record	E	Approximately 12000	R
CreateTable	R	1	R
Report	E	1	W

With Redundancy (by adding an attribute to transaction record to indicate it is for the current month)

Object	Type	Number of Accesses	Access Type
Worker	E	1	R
Access Record	R	1	R
Record	E	1	R
Transaction record	E	Approximately 60	R
CreateTable	R	1	R
Report	E	1	W

Operation 6 Cost Analysis: Perform monthly Backup

Sequence of Visit: Employee retrieves the records within a certain range of time and make backup.

Without Redundancy

Object	Type	Number of Accesses	Access Type
ITworker	E	1	R

Retrieve Records	R	> 1	R
Record	E	1	R
Transaction record	E	12000	R
Record	E	1	R
Client record	E	500	R
Record	E	1	R
Supplier record	E	30	R
Record	E	1	R
Raw material record	E	50	R
Record	E	1	R
Design record	E	350	R
Record	E	1	R
Payment record	E	12000	R
Record	E	1	R
Invoice record	E	15000	R
Record	E	1	R
Online order record	E	1000	R
Record	E	1	R
Employee record	E	100	R
Login record	E	90	R
Activity Record	E	250	R
Backup	R	1	R
CD-Writer	E	1	W

With Redundancy

Object	Type	Number of Accesses	Access Type
ITworker	E	1	R
Retrieve Records	R	> 1	R
Transaction record	E	60	R
Client record	E	20	R
Supplier record	E	5	R
Raw material record	E	10	R
Design record	E	20	R
Payment record	E	30	R
Invoice record	E	30	R
Online order record	E	5	R
Employee record	E	10	R
Login record	E	30	R
Activity Record	E	80	R
Do Backup	R	1	R
CD-Writer	E	1	W

Operation 7 Cost Analysis: Add an entry log for an activity

Sequence of Visit: Employee logins to the system and perform any activity.

Without Redundancy

Object	Type	Number of Accesses	Access Type

Worker	E	1	R
Access Record	R	1	R
Record	E	1	R
Worker record	E	1	R
Do web Login	R	1	R
Record	E	1	R
Activity record	E	1	R
Weblogin record	E	1	W

With Redundancy

Object	Type	Number of Accesses	Access Type
Worker	E	1	R
Access Record	R	1	R
Worker record	E	1	R
Do web Login	R	1	R
Record	E	1	R
Activity record	E	1	R
Weblogin record	E	1	W

Decision on Redundancy

Eliminating redundancies have some major advantages. It decreases the complexity of the code and the total storage space, which result in increase in the access time.

The following redundancies are considered as disadvantages. They will be removed from the restructured class diagrams.
1. Adding a date attribute to all the record classes.
2. Generalization: Employee, Client and Supplier classes share common attributes for contact information.

Generalization

For generalization on Employees:
- The relevant operations make no distinctions between the child classes and these classes have no specific attributes;
- We can therefore delete child classes (Manager, ITworker, Worker) and an attribute "Position" to the parent Employee class.

Partitioning on Records:
- The operations are distinct between child classes (e.g. Employee and Payment). These classes have specific attributes;
- We can therefore remove the parent Record class and let the child classes stand individually. This saves storage place for having irrelevant parent operation for some child classes

Merging of concepts:
- Since the Worker, Manager and ITworker have some distinct operations, these 3 classes can be merged into class Empolyee with the union of all their operations.

Employee class:

- The UserID can be served as the primary identifies.

Transformation into Relational Model
- Employee (<u>UserID</u>, Position, joinDate, Name, Fax, Phone, password, Salary, dob, recordDate)
- Payment (remark, clientID, invoiceID, paymentAmount, paymentDate, paymentMethod, <u>paymentID</u>, recordDate)
- RawMaterialRecord (<u>rawID</u>, supplierID, quantity)
- DesignRecord (designFile, <u>designID</u>, instruction, recordDate)
- InventoryRecord (comment, quantity, rawID, type, recordDate, <u>inventoryID</u>)
- Transaction (designID, paymentID, quantity, rawID, <u>tranID</u>, recordDate)
- SupplierRecord (supplyRawRecordID, supplierContact, <u>supplierID</u>, supplierfax, supplierphone, <u>supplierName</u>, recordDate)
- ClientRecord (address, comment, fax, clientID, name, phone, pic, recordDate)
- Invoice(client, <u>invoiceID</u>, trainID, invoiceDate)
- ActivityRecord (activityCode, desciption, recordDate)
- OnlineOrder (<u>orderID</u>, designID, orderStatus, paymentMethod, shipmentDate, recordDate)
- WebLoginRecord (activityLog, employeeID, loginTime, logoutTime, sessionID, recordDate)

<u>Normalization</u>
Now we normalize the relational schema to reduce the number of functional dependencies.

Case 1: Inventory Record

InventoryRecord(comment, quantity, rawID, type, recordDate)

Normalization Table:

1NF	Yes
2NF	Yes
3NF	Yes

Case 2: Supplier Record

SupplierRecord (supplyRawRecordID, supplierContact, <u>supplierID</u>, supplierfax, supplierphone, supplierName, recordDate)

Normalization Table:

1NF	Yes
2NF	No SupplierRecord (supplyRawRecordID, <u>supplierID</u>, contactInfoName, recordDate) ContactInfo (supplierName, supplierphone, supplierfax,

	supplierContact)
3NF	No SupplierRecord (supplyRawRecordID, <u>supplierID</u>, contactInfoName, recordDate) ContactInfo (<u>supplierName</u>, supplierphone, supplierfax, supplierContact)

Case 3: Employee Record

Employee (<u>UserID</u>, Position, joinDate, Name, Fax, Phone, password, Salary, dob, recordDate)

Normalization Table:

1NF	Yes
2NF	No Employee (<u>UserID</u>, contactInfoName, privateInfoName, recordDate) ContactInfo (<u>Name</u>, Fax, Phone) PrivateInfo (<u>Name</u>, joinDate, password, Salary, dob, Position)
3NF	No Employee (<u>UserID</u>, contactInfoName, privateInfoName, recordDate) ContactInfo (<u>Name</u>, Fax, Phone) PrivateInfo (<u>Name</u>, joinDate, password, Salary, dob, Position)

Case 4: Client Record

ClientRecord (address, comment, email, fax, clientID, name, phone, pic, recordDate)

Normalization Table:

1NF	Yes
2NF	No ClientRecord (<u>clientID</u>, contactInfoName, recordDate) ContactInfo (Name, Fax, Phone, address, email, pic, comment)
3NF	No ClientRecord (<u>clientID</u>, contactInfoName, recordDate) ContactInfo (Name, Fax, Phone, address, email, pic, comment)

Case 5: Raw Material Record

RawMaterialRecord (<u>rawID</u>, supplierID, quantity)

Normalization Table:

1NF	Yes
2NF	Yes
3NF	Yes

Case 6: Payment

Payment (clientID, invoiceID, paymentAmount, paymentDate, paymentMethod, paymentID, recordDate, remark)

Normalization Table:

1NF	Yes
2NF	Yes
3NF	Yes

Case 7: Design Record

DesignRecord (designFile, designID, instruction, recordDate)

Normalization Table:

1NF	Yes
2NF	Yes
3NF	Yes

Case 9: Online Order

OnlineOrder (orderID, designID, orderStatus, paymentMethod, shipmentDate, recordDate)

Normalization Table:

1NF	Yes
2NF	Yes
3NF	Yes

Case 9: Transaction Record

Transaction (designID, paymentID, quantity, rawID, tranID, recordDate)

Normalization Table:

1NF	Yes
2NF	Yes
3NF	Yes

Case 10: Activity

ActivityRecord (activityCode, desciption, recordDate)

Normalization Table:

1NF	Yes

2NF	Yes
3NF	Yes

Case 11: Login

WebLoginRecord (activityLog, employeeID, loginTime, logoutTime, sessionID, recordDate)

Normalization Table:

1NF	Yes
2NF	No WebLoginRecord (activityLog, employeeID, loginInfo, recordDate) LoginInfo(loginTime, logoutTime, sessionID)
3NF	No WebLoginRecord (activityLog, employeeID, loginInfo, recordDate) LoginInfo (loginTime, logoutTime, sessionID)

Case 12: Invoice

Invoice (clientID, invoiceID, tranID, invoiceDate)

Normalization Table:

1NF	Yes
2NF	Yes
3NF	Yes

Justification that the design meets relevant requirements

A well-designed database allows operations to be performed efficiently with a conservative storage space. Object-oriented database are chosen for Bestley to handle all the transactions, records, and accounting information. This is suitable since the data Model for Bestley has a very strong object-oriented structure. After observing the redundancies of the initial class diagram, it is discovered that the 3 types of employees, ITworker, Manager and Worker could be generalized into one Employee class. This saves storage space and allows a simpler database structure. A date attribute can also be added to various kinds of records. Doing this can greatly reduce the search time for records, the time for backup and generating report. After normalization, the schema does not suffer from anomalies.

User Interface Design

Overview of the current user interface design

Bestley currently do not have any user interface design in information system because it had not yet computerize. Therefore, the main motivation of the new interface design is to produce a simple, direct and user friendly interface. Nowadays, almost everyone

got a computer, therefore, our interface design mainly based on Windows environment.

User group

Bestley is a small-sized company with a simple employment structure and production processes. The user groups can be classified into 5 major categories.

Non-technical User

Non-technical users refer to people that only responsible for using the system. The interface mainly divided into 2 parts. One part of the interface only available for employees in the company. Another part of the interface is available to all the potential customers on web.

Management level

This is the highest management level refers to the director and assistant director. They are responsible for company's future planning. In order to make sensible and sound policy, the operation involved include access sales, employee record and retrieve all kind of data in database.

Secretary

Secretary is responsible for all the data input, e.g., raw material, supplier and sales data
Besides data input, access sales in order to prepare the entire legal document.

Accountant

This user group is responsible for retrieve information in order to prepare invoice, balance sheet, trial balance and income statement.

Client

With the innovation of the new information system, all potential customers can search all the products catalog, payment methods and place order online through internet.

Technical User

Information System Developer

Information System Developer is a part-time employee who responsible for setup, monthly maintenance of system and backup of database. Only the information system developer can access Backup submenu. This is because changing any settings in Backup may cause great damage to database or system.

State Diagrams describing the dialogues supported by the interface
Please refer to Appendix

Mockups of Windows
Please refer to Appendix

Website Design

The main purpose of the web site is to provide a channel for the public and potential customer to learn about this company. Therefore, more people can order Bestley's online in a few minutes. As a result, Bestley can capture even the smaller customers through online ordering.

Please refer to Appendix for site map and html document.

Input / Output Design

Input Design

There is no special input design since the data is mainly input by users. Keyboard and mouse are main tools to assist the user to facilitate the data input.
User group
   1. Clients will input data through web site.
   2. Employees will input data through local intranet with a series of interface.
The users just need to follow the link and the instruction provided on the interface to input the data.

Output Design

Bestley's employees need to retrieve the data frequently in order to facilitate the transaction. This can be in two forms, one is display the data on the screen and the other one is to produce a report using the data.

Data like client and raw material information can display on the screen after using a relevant field to search. Then list out all the information in the account on the screen. The user has a choice to print the account information for further process.
Besides, only the director can retrieve statistical report which is a summary of specific information. The purpose of restriction of access to statistical report because to prevent the misuse of company's and client's privacy. This can be in the form of bar chart, histogram or pie chart which server the purpose of future planning.

Justification that the interface design meets relevant requirements

A success interface design must have user-friendly environment. Nowadays, a lot of people have a personal computer at home and most of the computer use Windows as operating system. The gaining popularity of Windows operating system over the rest (e.g. UNIX, DOS), this is because of user friendly features of interface design. In order to reduce the stress of computerization put on employees and the time for them to get familiar with the new system, therefore, the new interface design will try to follow the format of Windows interface. As a result, user-friendly and ease to use are achieved.

The potential customers should feel comfortable to use internet since internet is very common and every website is more or less the same. Therefore, in the external web-based online purchasing system, a simple homepage design can already achieve the properties mention above.

Everyone can order products online through internet and settle the transaction by credit card. In order to protect the customers' and Bestley's against any kind of technology crime. Bestley sets up a PayPal system in order to prevent online security fraud. I.e. the credit card information is entering to the Bestley's database only when customer sets up the account. From the mockups of windows in appendix, the employees cannot access the credit card information. The credit card information is lock up and encrypted in the Bestley's database. Security measures is strictly enforced and match with security requirement.

The design of the interface follows the characteristics of a well-designed interface.
<u>Characteristics of a Well-Designed Interface</u>
Affordances
Clear and obvious buttons provide for user on each menu which describes what the user can do with it.
Forcing Functions
Disable Backup button in main menu for general employees. This will prevent user from taking certain actions.
Automatic Learning
The repetitive pattern of the user actions and screen displays which force learning on the user.

Appendix

Appendix A : Comparison between on performance between difference OS
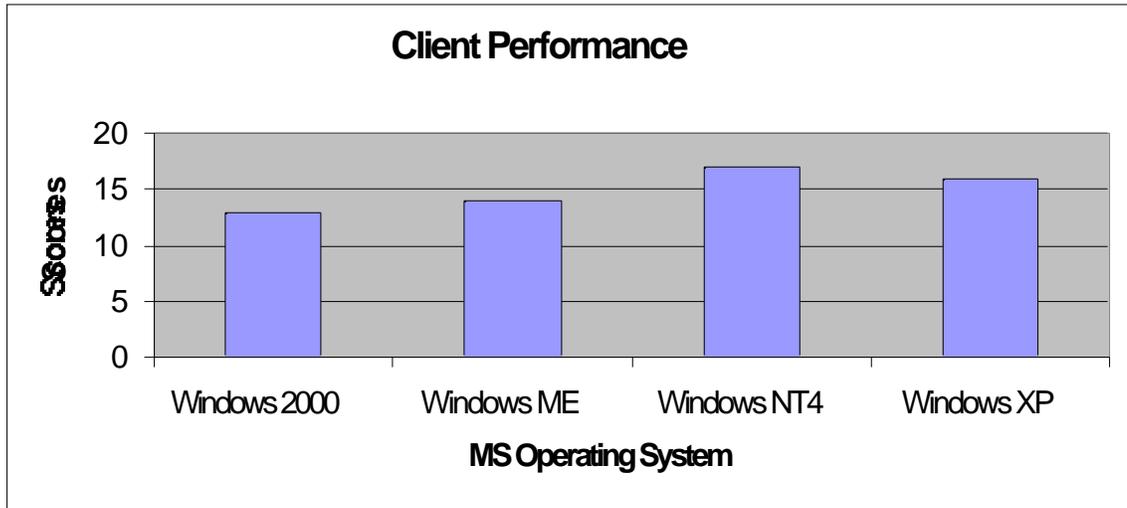
Chart 1. – Comparing averages of 5 desktop and two notebook systems running three different memory configuration (Higher scores represent better performance.)
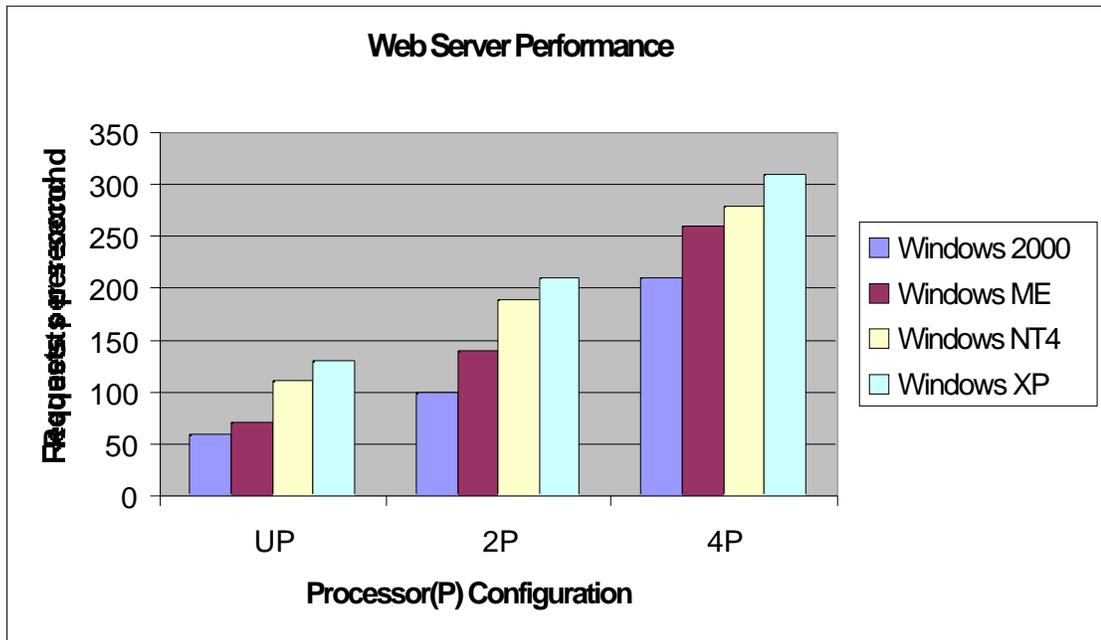
Chart 2. – Comparison of in-process performance of different OS on different processor (P) configurations, measured in requests per second. (Higher scores represent better performance.)

Networking Performance
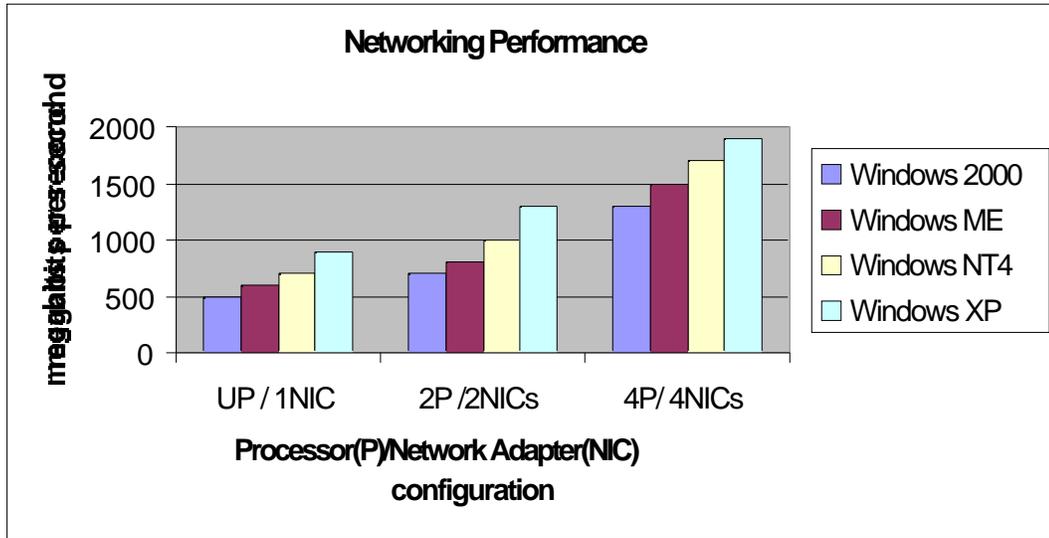
Chart 3. – Comparison of network throughput of different OS on systems measured in megabits per second (Mbps) with different processor (P) and network adapter (NIC) configurations. (Higher scores represent better performance.)

File & Print Server Performance
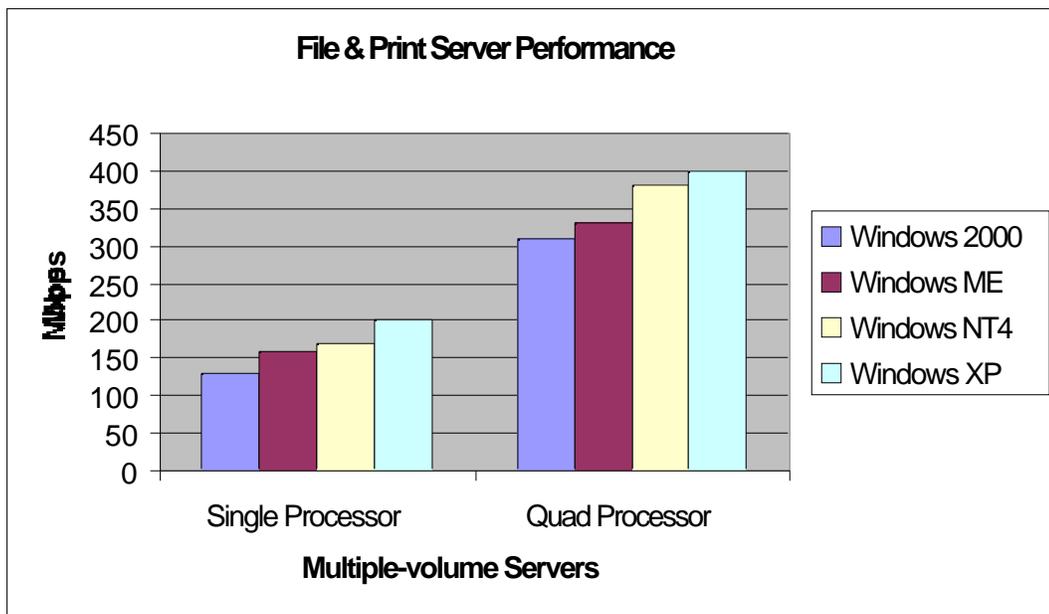
Chart 4. – Comparison of different OS in terms of file server performance on multiple-volume servers, measured in Mbps. (Higher scores represent better performance.)

Application Server Performance

Scores

4
3
2
1
0

Windows 2000   Windows ME   Windows NT4   Windows XP

MS Operating System
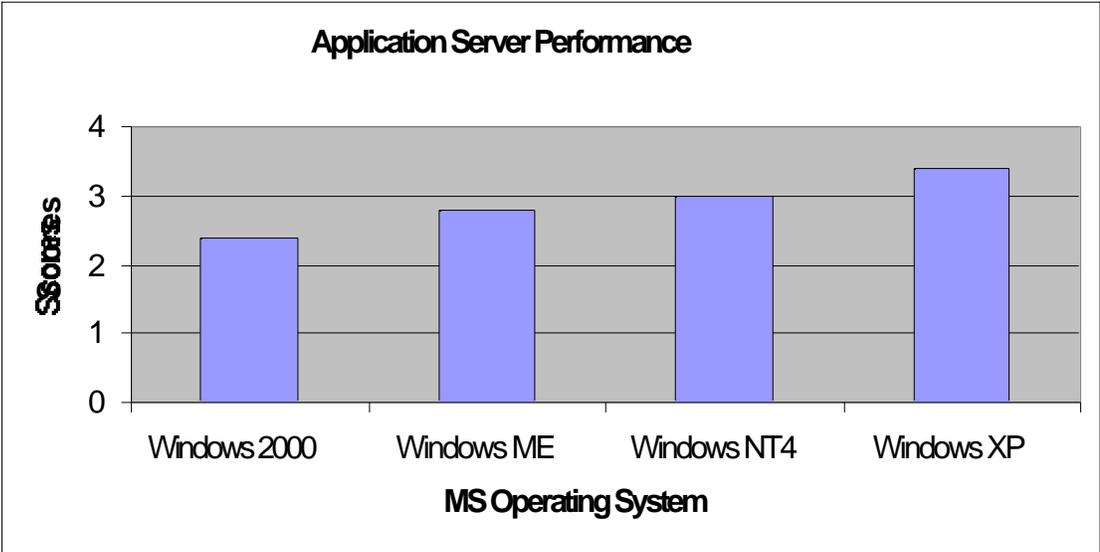
Chart 5. - Windows XP outperforms the other system. (measured in millions of items processed per hour).

Appendix B : **Class Diagrams**
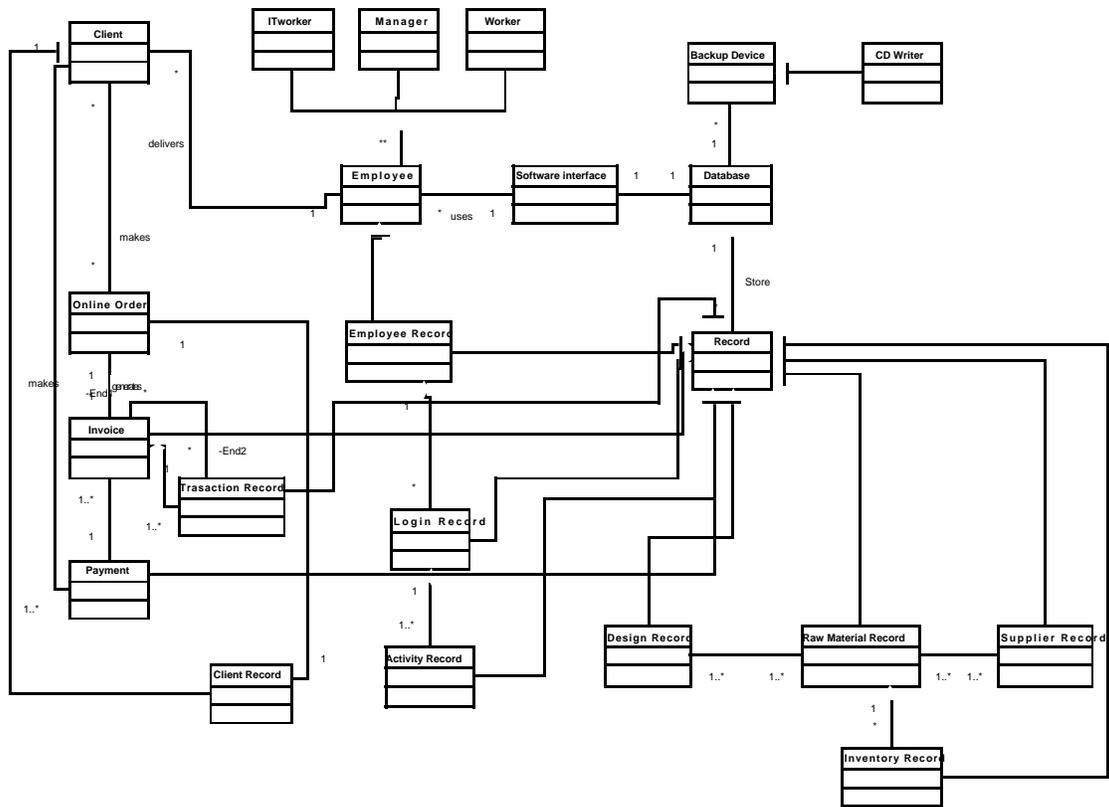
Appendix C : Sequence Diagram

Appendix C.1 : Client Upload Order

Sequence Diagram - Client Upload Order
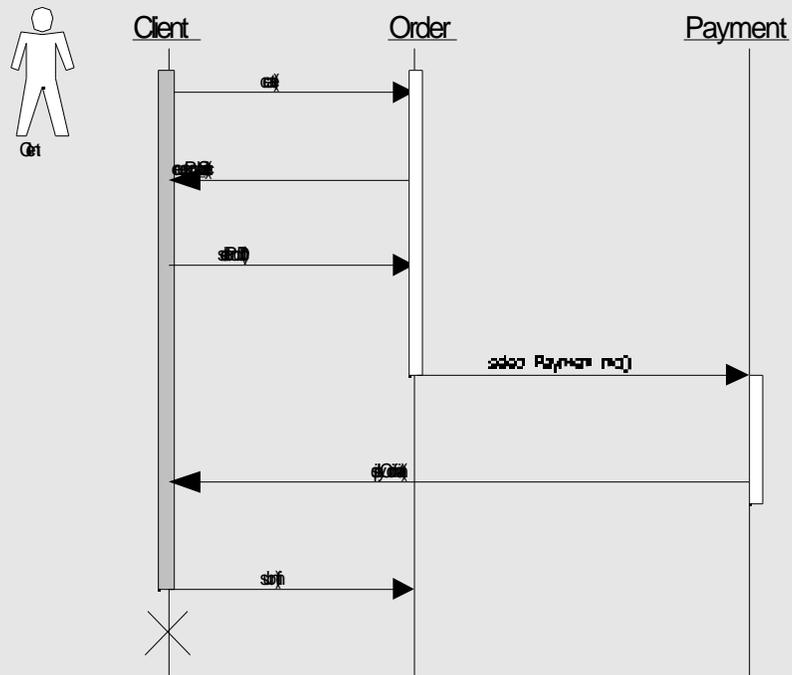
Appendix C.2 : Employee download online order

Sequence Diagram - Employee download online order

Employee   Web Site   Database

Employee
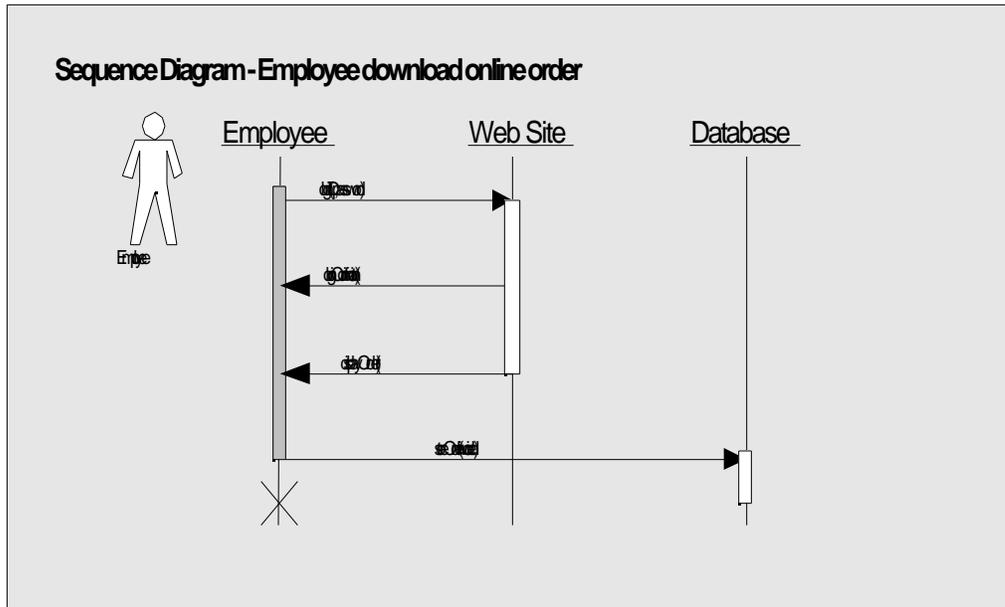
Appendix C.3 : Create Sales Report

Sequence Diagram - Create Sales Report
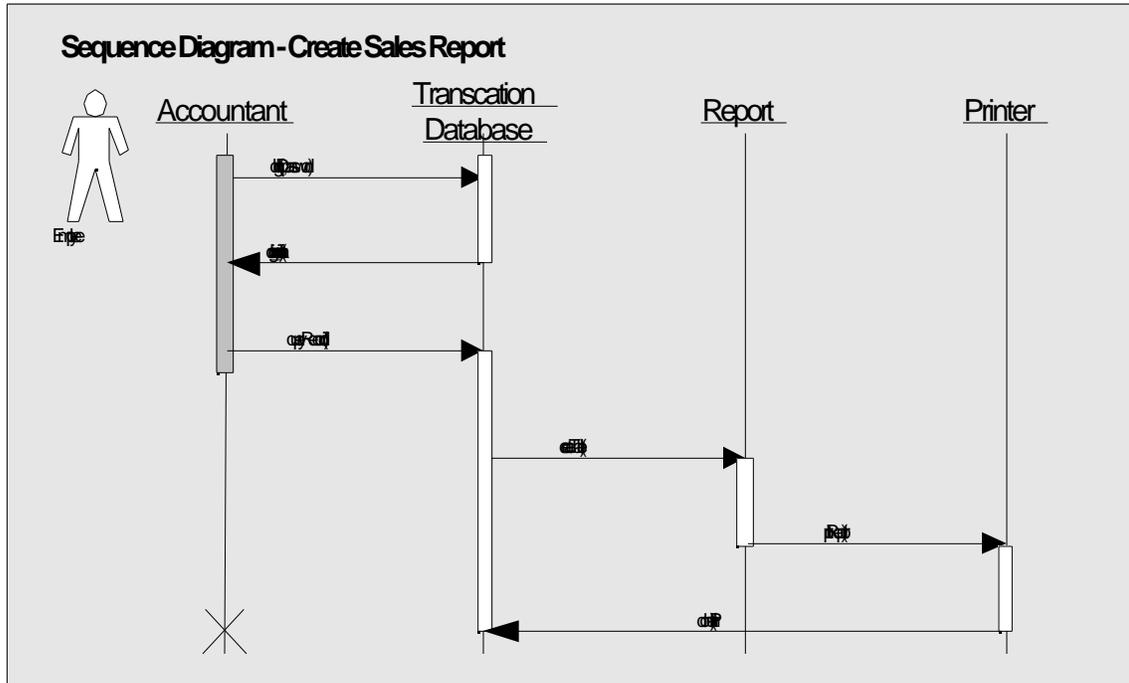
Appendix C.4 : Report

Sequence Diagram - Report

Employee   Order   Client   Product

CEO

login(user)

view sales report

display sales report

get client report

display client report

view product report

display product report
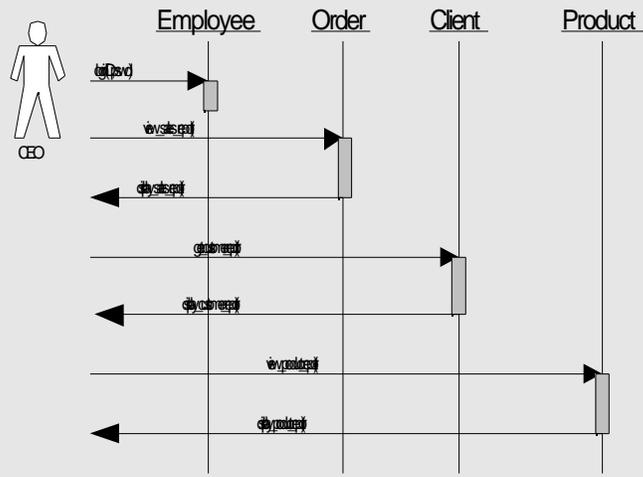
Appendix C.5 : Store Transaction to Database

Sequence Diagram - Store Transaction to Database

Appendix C.6 : Backup

Sequence Diagarm - Backup

Employee

Database

CD-Writer
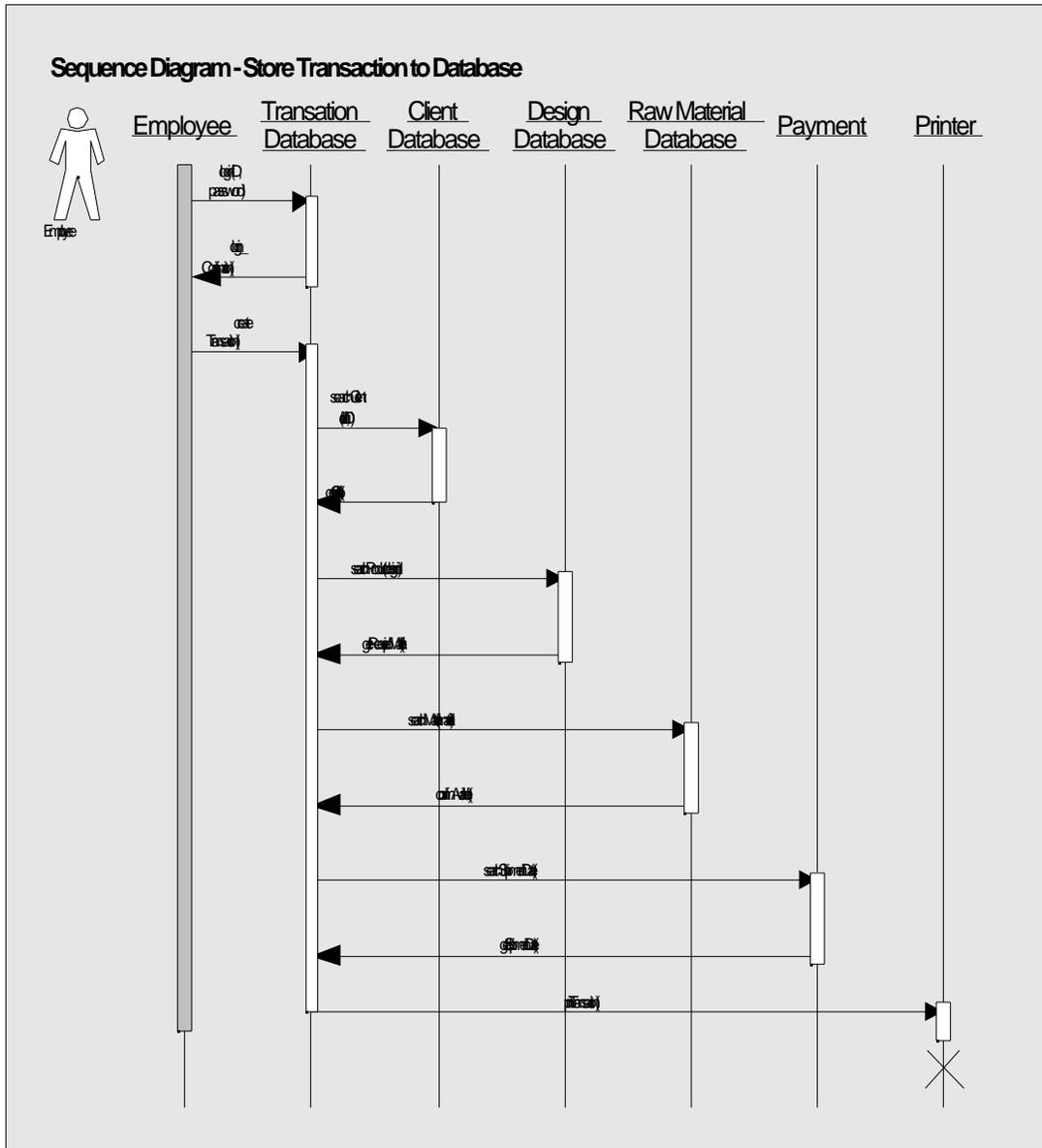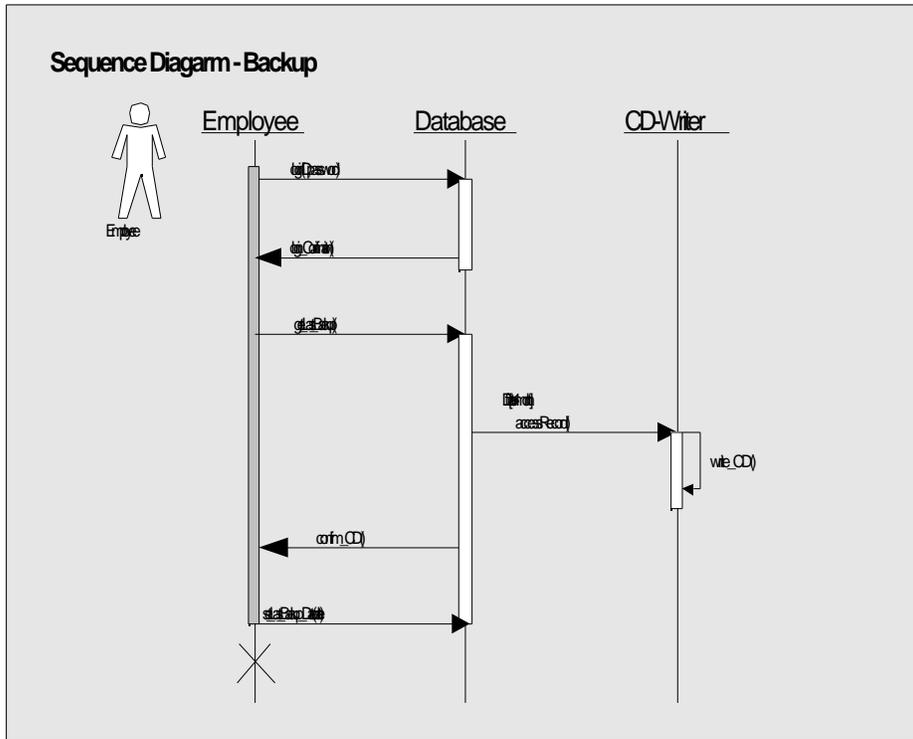
Appendix D : State Diagram

Appendix D.1 : State-diagram of class Database
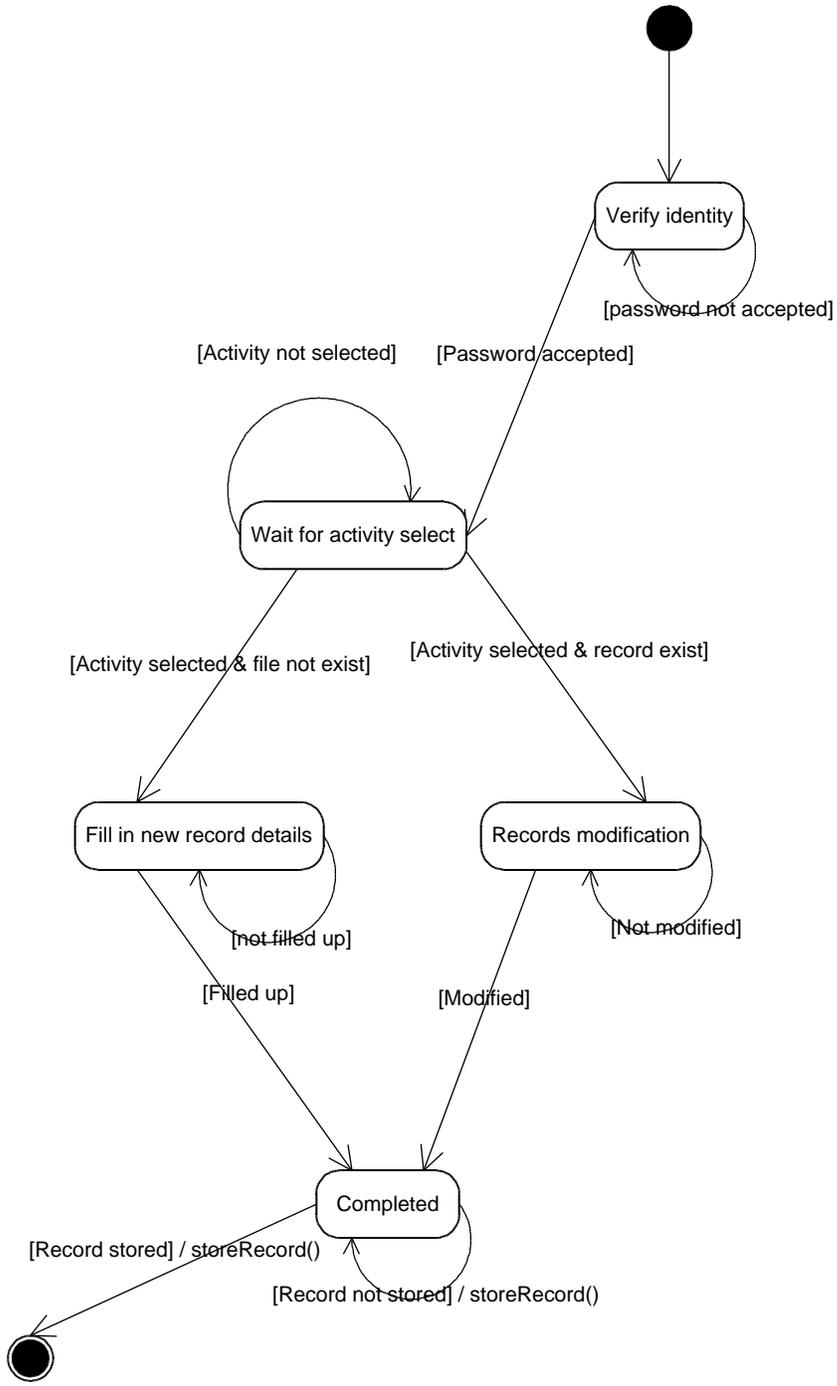
Appendix D.2 : State-diagram of class Online-Order

/ NewOrder()

Ordered

/ request_Product_Selection()

Wait for product selction

[Product Selected]

Payment

[Product Not Selected]

Alert user to select payment method

Wait for payment selection

[Payment Selected]

Completed

/ displayConfirmation()

[Payment Not Selected]
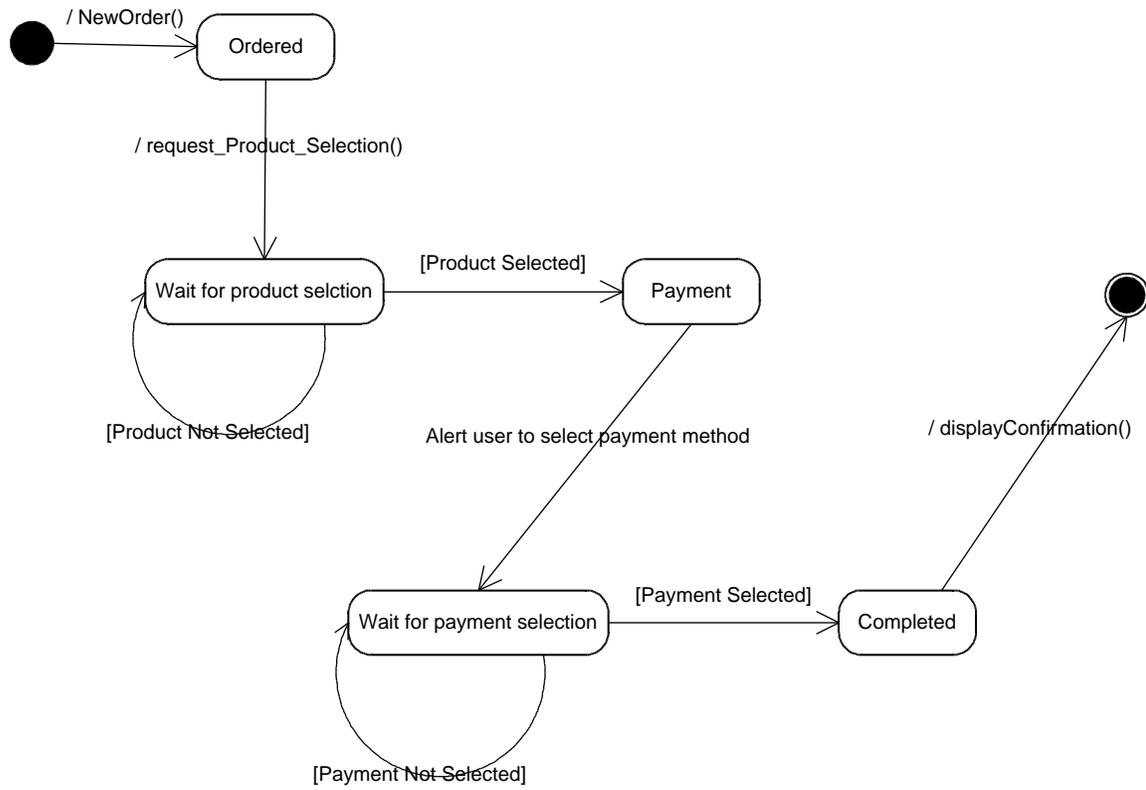
Appendix D.3 : State-diagram of class Client Record

Check

[record not exist] / getCustomerInfo()

[record exist] / modCustomerInfo()

[not finish]

Fill up new client info.

Record modification

[not finish]

[finished]

[finished]

Complete

/ storeCustomerRec()
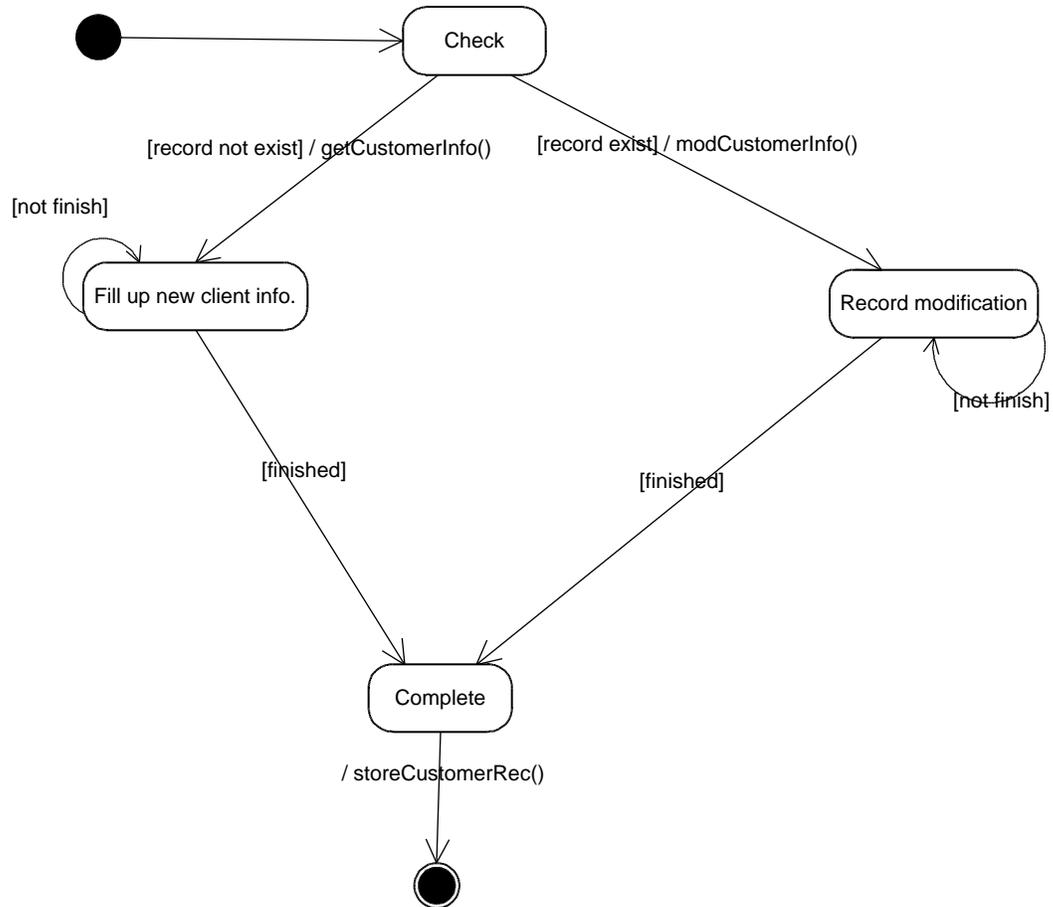
Appendix D.4: State-diagram of class Employee Record

Check

[Record not exist] / getEmployeeInfo()

[Record exist] / modEmployeeInfo()

[not finish]

Fill up info

State3

[Not finish]

[finished]

[finished]

Complete

/ storeEmployeeRec()
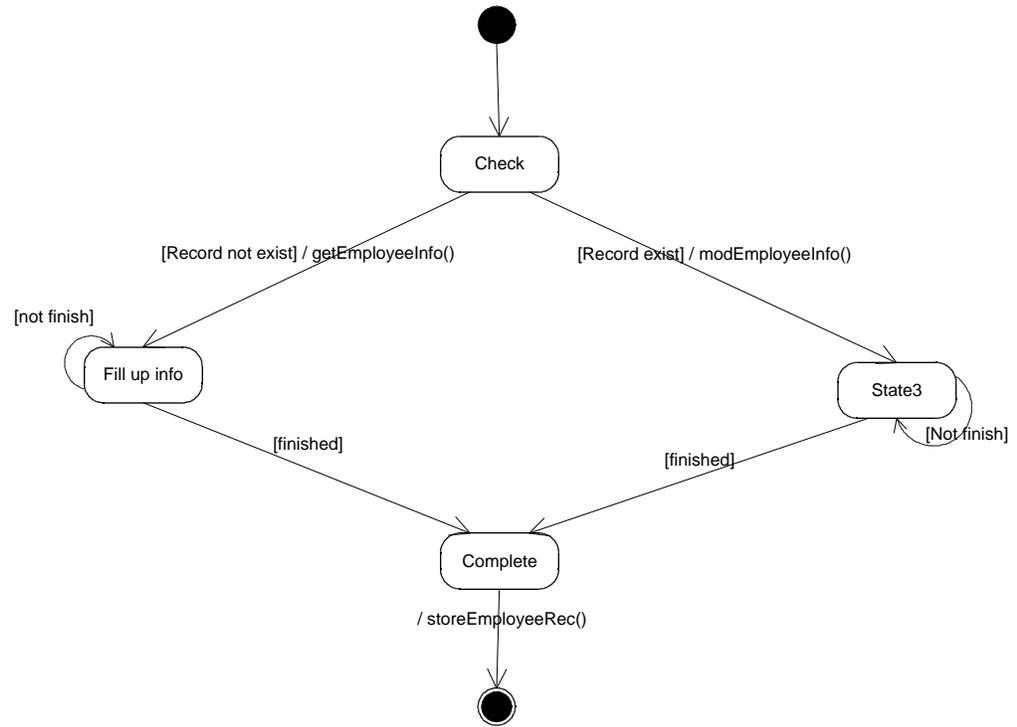
Appendix D.5 : State-diagram of class Backup Device

Identity verification

[password not accepted]

[password accepted] / accessRecord()

Accessing database

/ CDwriter.Write_CD()

[backup not success]

Copies Created

[backup success]

Complete
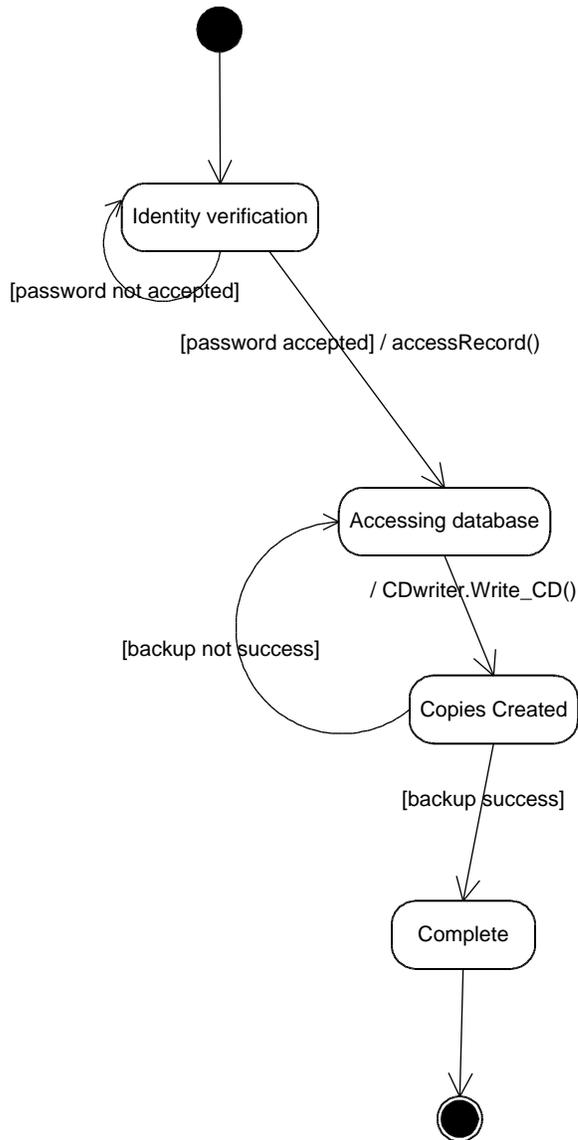
Appendix D.6 : State-diagram of class Design Record

/ employee.design()

Design Draft

[Design record not exist] / getDesignInfo()　　　[Design record existed] / modDesignInfo()

[not finish]　　　　　　　　　　　　　　　　　　　　　　　　　　[not finish]

Creating new design record　　　　　　　　　　　　　Update design record info.

[finished]　　　　　　　　　　　　　　　　　[finished]

Complete
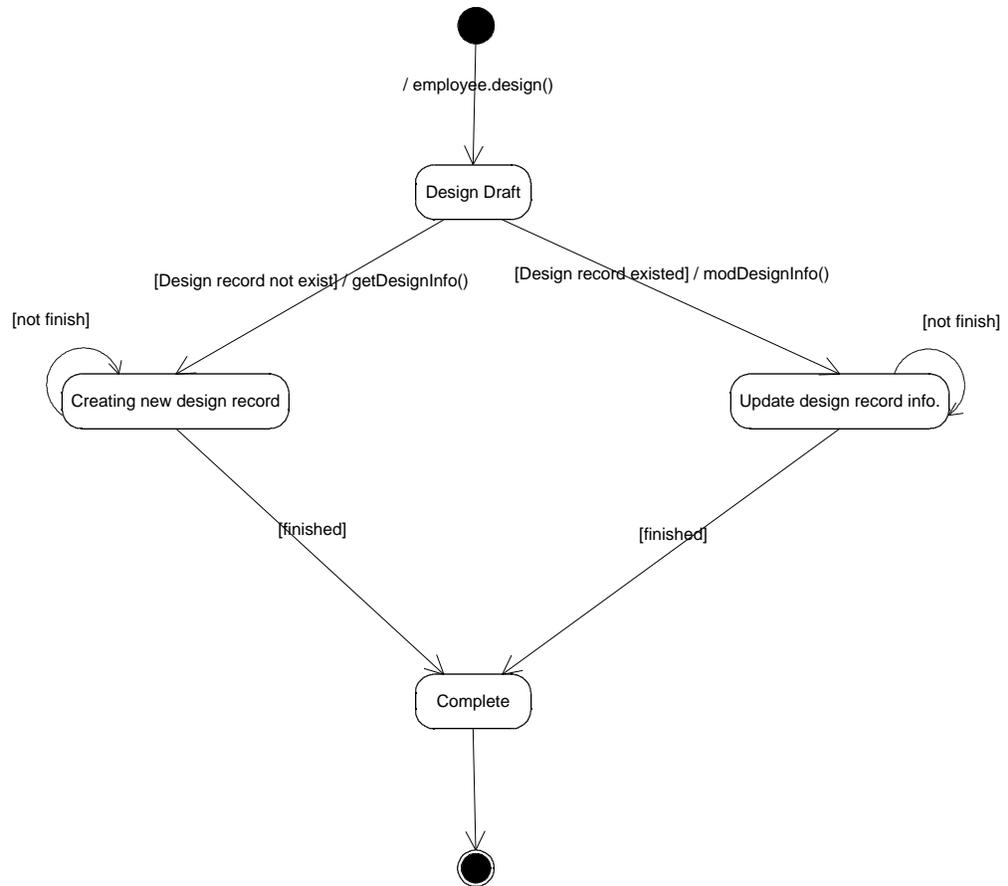
Appendix E.7 : State-diagram of class Raw Materials Record

/ employee.buyMaterial()

Check

[Raw Material not existed] / getMaterialInfo()

[Raw Material existed] / modMaterialInfo()

[not finished]

Create new Raw Materials record

Modify Raw Materials record

[not finished]

[finished]

[finished]

Complete

Appendix E.8 : State-diagram of class Supplier Record

Check

[Supplier not existed] / getSupplierInfo()

[Supplier existed] / modSupplierInfo()

Create new supplier record

Update supplier record

[not finished]

[not finished]

[finished]

[finished]

Complete

Appendix E.9 : State-diagram of class Inventory Record

/ employee.usedMaterial()

/ employee.buyMaterial()

Material used up

New Material received

/ employee.getMaterialQuantity()

/ employee.getMaterialQuantity()

/ modInvQuantity()

Completed

Appendix E : State Dialogue

Login State Dialogue
This State dialogue happens at the beginning of the system.

Start / Open
window

[valid password]/
go to Main Menu

| Authenication Window | Main Menu Window |

[Invalid password]/
clear text box

Click Cancel /
Close window

Main Menu State Dialogue
If the user login successfully, then main menu will display to the user.

| Client Acct Window

Click Raw
Material, Supplier
Accounts /
change interface

| Raw Material / Supplier Window

Click Design
Accounts/
change interface

| Main Menu Window |

| Design Window

Click Statement /
change interface

| Statement Window

Click Reports/
change interface

Click Delivery
Schedule /
change interface

| Reports |

[System developer]
click Backup / change
interface

| Backup | | Delievery Schedule

Client Account Menu State Dialogue

Enter client
name / change
interface

Client Account select
/ go to client account
window

Enter invoice
No. / change
interface

Click logout /
close window

Enter client ID /
change interface

| Client Account Window

| Client detail information window

Click on back /
return to the
previous menu

Click main
menu/ change
interface

Click Add a
client / change
interface

Click back / return
to previous
window

Select an
invoice / change
menu

Click back / return
to previous
window

| Main Menu |

Add a client account window

| Invoice Menu|

Raw Material / Supplier Menu State Dialogue

Supplier detail information window

Enter supplier ID /
change interface

Click back / return
to previous
window

print

Raw Material and
Supplier Account
select / go to Raw
Material and supplier
window

Enter Raw Material
ID / change
interface

Raw Material / Supplier detail window

| Raw Material / Supplier Window

Click back / return
to previous
window

Click logout /
close window

Click Add a supplier
/ change interface

Click on main
menu / change
interface

| Main Menu |

Add a supplier account window

Click back / return
to previous
window

Design Menu State Dialogue

●

Design Account
select / go to Design
window

Click OK / return   Click back / return
to previous   to previous
window   window

| Design Account Window | Design detail information window

Enter Design ID /   print
change interface

Click logout /
close window

Click on main | main menu |
menu / change
interface

●
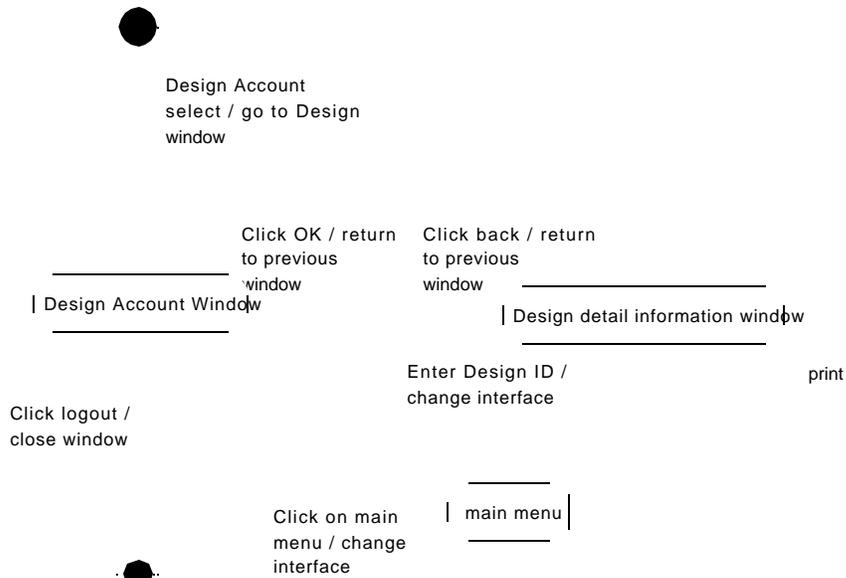
Invoice Menu State Dialogue

●

Invoice Account   Click back / return
select / go to invoice   to previous
window   window | invoice detail information window

Enter invoice ID /   print
change interface

Click Add a   Click Payment/
supplier /   change interface
change interface

| Invoice menu | generate invoice window | Payment subsystem

Click Add a
supplier /
change interface

Click logout /   Click OK / go to   print
close window   main menu

Click on main | main menu |
menu / change
interface

●

Payment Menu State Dialogue

Appendix F: Mockups windows

Authentication Window

LoginID and password is entering by user to check for authentication in order to gain further access of the system.

Main Menu (for employees)

This main menu is use by general employees e.g. accountant, secretary. General employee except information system developer and director can't access backup submenu. (Disable Backup button) This is to ensure no one will mistakenly change the database or setting of the system.

```
┌──────────────────────────────────────────────────────┐
│ Bestley Main Menu                                      │
│                                                        │
│                                                        │
│         ┌──────────────────────────┐                   │
│         │      Client Accounts     │                   │
│         ├──────────────────────────┤                   │
│         │ Raw Material / Supplier Accounts             │
│         ├──────────────────────────┤                   │
│         │      Design Accounts     │   ┌─────────────┐ │
│         └──────────────────────────┘   │  Payments   │ │
│                                        ├─────────────┤ │
│                                        │  Invoices   │ │
│                                        └─────────────┘ │
│         ┌──────────────────────────┐                   │
│         │        Reports           │                   │
│         ├──────────────────────────┤                   │
│         │     Delivery Schedule    │                   │
│         └──────────────────────────┘                   │
│                                                        │
│   ┌──────────┐                         ┌──────────┐    │
│   │  Backup  │                         │  Logout  │    │
│   └──────────┘                         └──────────┘    │
│ 04/15/0209:11 AM                                       │
└──────────────────────────────────────────────────────┘
```
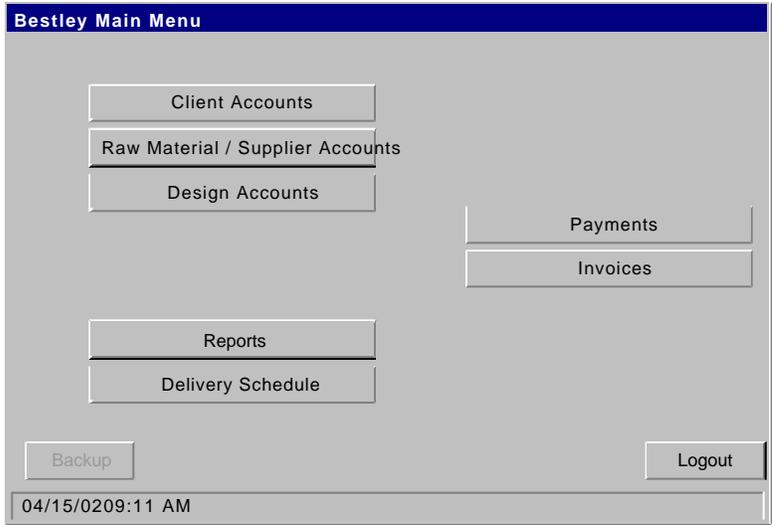
Main Menu (for Information System Developer and Director)

This main menu is different from the above since the Backup button is enabling.

```
┌──────────────────────────────────────────────────────┐
│ Bestley Main Menu                                      │
│                                                        │
│                                                        │
│         ┌──────────────────────────┐                   │
│         │      Client Accounts     │                   │
│         ├──────────────────────────┤                   │
│         │ Raw Material / Supplier Accounts             │
│         ├──────────────────────────┤                   │
│         │      Design Accounts     │   ┌─────────────┐ │
│         └──────────────────────────┘   │  Payments   │ │
│                                        ├─────────────┤ │
│                                        │  Invoices   │ │
│                                        └─────────────┘ │
│         ┌──────────────────────────┐                   │
│         │        Reports           │                   │
│         ├──────────────────────────┤                   │
│         │     Delivery Schedule    │                   │
│         └──────────────────────────┘                   │
│                                                        │
│   ┌──────────┐                         ┌──────────┐    │
│   │  Backup  │                         │  Logout  │    │
│   └──────────┘                         └──────────┘    │
│ 04/15/0209:26 AM                                       │
└──────────────────────────────────────────────────────┘
```
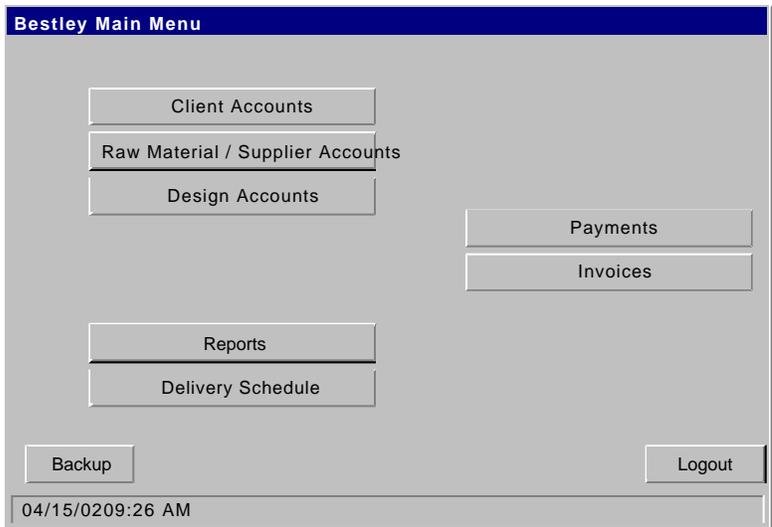
Client Accounts Submenu

```
┌─────────────────────────────────────────────┐
│ Bestley - Client Account Menu               │
├─────────────────────────────────────────────┤
│         Search a client                     │
│                                             │
│   ClientID      [                ]          │
│                                             │
│   Invoice No.   [                ]          │
│                                             │
│   Client Name   [                ]  ┌──────────┐ │
│                                     │  Search  │ │
│                                     └──────────┘ │
│       ┌─────────────────────────┐           │
│       │      Add a client       │           │
│       └─────────────────────────┘           │
│     ┌──────────────┐                        │
│     │  Main Menu   │                        │
│     └──────────────┘                        │
│     ┌──────────────┐                        │
│     │   Logout     │                        │
│     └──────────────┘                        │
│                                             │
├─────────────────────────────────────────────┤
│ 04/15/0209:42 AM                            │
└─────────────────────────────────────────────┘
```

Client Accounts Detail Information Window

This is the window display on the screen after search a client is performed. Any correction of existing client's information can perform in this window too. (I.e. just select a particular field and type the new data to overwrite the old one).

```
┌─────────────────────────────────────────────────────┐
│ Bestley - Chan, Peterson 456213                     │
├─────────────────────────────────────────────────────┤
│ Account Creation Date : 5 March, 2001               │
│ Name : Mr. Chan, PetersonPhone No. : 416-507-6922   Fax No. : 416-123-8966 │
│ Company Name : The one Ltd.   Gender : M      Date of Birth : 5 May, 1970 │
│ Email : peterson@boo.com                            │
│ Address : 4 mile Cr., Markham, Toronto L4M 3k5      │
│ Comments :  Colour of plastic bags  - purple with logo │
│   Date        Invoice ID    Active/Inactive  Total Amount  Amount Paid │
│  ┌──────────────────────────────────────────────────┐  │
│  │03/15/2001    12365          I          1236.02▲│  1236.02 │
│  │04/15/2001    23654          I           256.00 │   256.00 │
│  │03/05/2002    56563          A           365.00 │     0.00 │
│  │                                                │  │
│  │                                              ▼│  │
│  └──────────────────────────────────────────────────┘  │
│                                       ┌─────────────┐   │
│                                       │ Goto Invoice│   │
│                                       └─────────────┘   │
│ ┌──────────┐  ┌──────────┐                          │
│ │ < Back   │  │  Print   │                          │
│ └──────────┘  └──────────┘                          │
├─────────────────────────────────────────────────────┤
│ 04/15/0209:53 AM                                    │
└─────────────────────────────────────────────────────┘
```

Raw Material / Supplier Accounts Submenu

Bestley - Raw Material / Supplier Menu

Search for Supplier Accounts

Supplier ID   [_____]

Raw Material ID   [_____]   [Search]

[Add a supplier]

[Main Menu]

[Logout]

04/15/0209:46 AM

Invoice Accounts Submenu

Bestley - Invoice Menu

Search an invoice

Invoice ID   [_____]   [Search]

[Generate an invoice]

[Main Menu]

[Logout]

04/15/0209:48 AM

Generate an Invoice Window

```
Bestley - Generate an invoice

Generate an invoice                    Date : 10 April 2002

Invoice ID      243482          Quantity     [          ]

Client ID       [          ]    Handle Employee ID  [    ]

Raw Material ID [          ]    Delivery Date and Tim [    ]

Design ID       [          ]         Special Intruction

Order Method   ⊘ Online  ⊛ By Phone   [              ▲]
                                      [               ]
                                      [               ]
                                      [              ▼]
                                      [◄        ►    ]

[  < Back  ]  [  OK  ]  [ Payment ]

04/15/0209:51 AM
```

Payment Window

```
Bestley - Payment Menu

         Search an invoice

Invoice ID      [ 55445 ]   [ Search ]

          [  < Back  ]

          [  Logout  ]

04/15/0209:56 AM
```

**Payment Details Information Window**

This is the window display on the screen after the Search button on the above screen
pressed.

Bestley - Payment

Receipt

Date : 13 April 2002

Invoice ID : 55445   Receipt No. : 56598

Client ID : 23625

Delivery Date and Time :   15 April, 2002

Payment Method ⊘ Cash   ⊙ Cheque

Cheque No. | 5464-5626-5466596 |

Amount paid | 560.00 |

Payment handle employee ID | I |

| < Back | | OK | | Print |

04/15/0209:58 AM

Delivery Schedule Window

Bestley - Delievery Menu

Date : 14 April, 2002

Date	Invoice ID	Schedule Time	Total Amount	Amount Paid
04/14/2001	12365	12:00	1236.02	1236.02
04/15/2001	23654	13:00	256.00	256.00
04/19/2002	56563	17:00	365.00	0.00

| < Back | | Print |

04/15/0210:20 AM

Appendix G: Website design

<u>**Site Map**</u>
This site map is to show the overall structure of Bestley's homepage. Those lines represent the linkage of the different page.

```
                    Main page
                 (introduction.html)
                         |
          ┌──────────────┴──────────────┐
   Products and Design          History and contact page
     (catalog.html)                  (contact.html)



                 Client login page
                   (login.html)
                         |
          ┌──────────────┴──────────────┐
   Registration form              Order form
    (register.html)              (order.html)
                                      |
                                Confirmation page
                                  (confirm.html)
                                      |
                                 Thank you page
                                   (final.html)
```
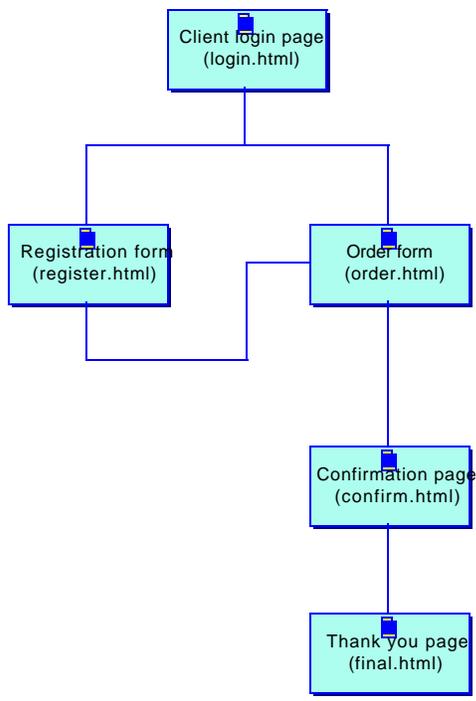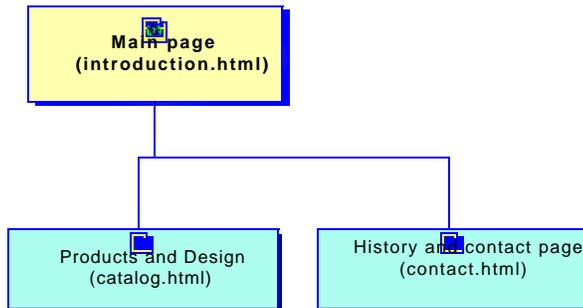
<u>**Sample Bestley's homepage**</u>

Introduction.html
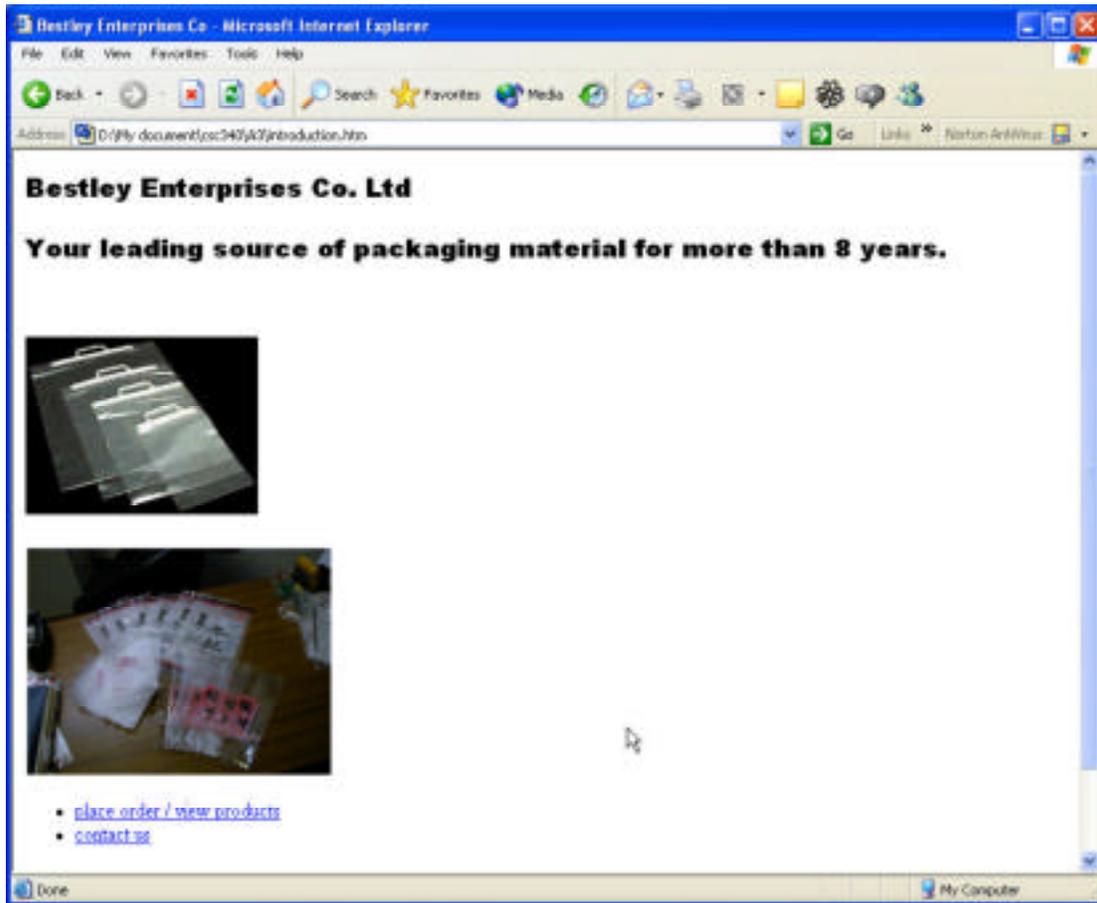
login.html
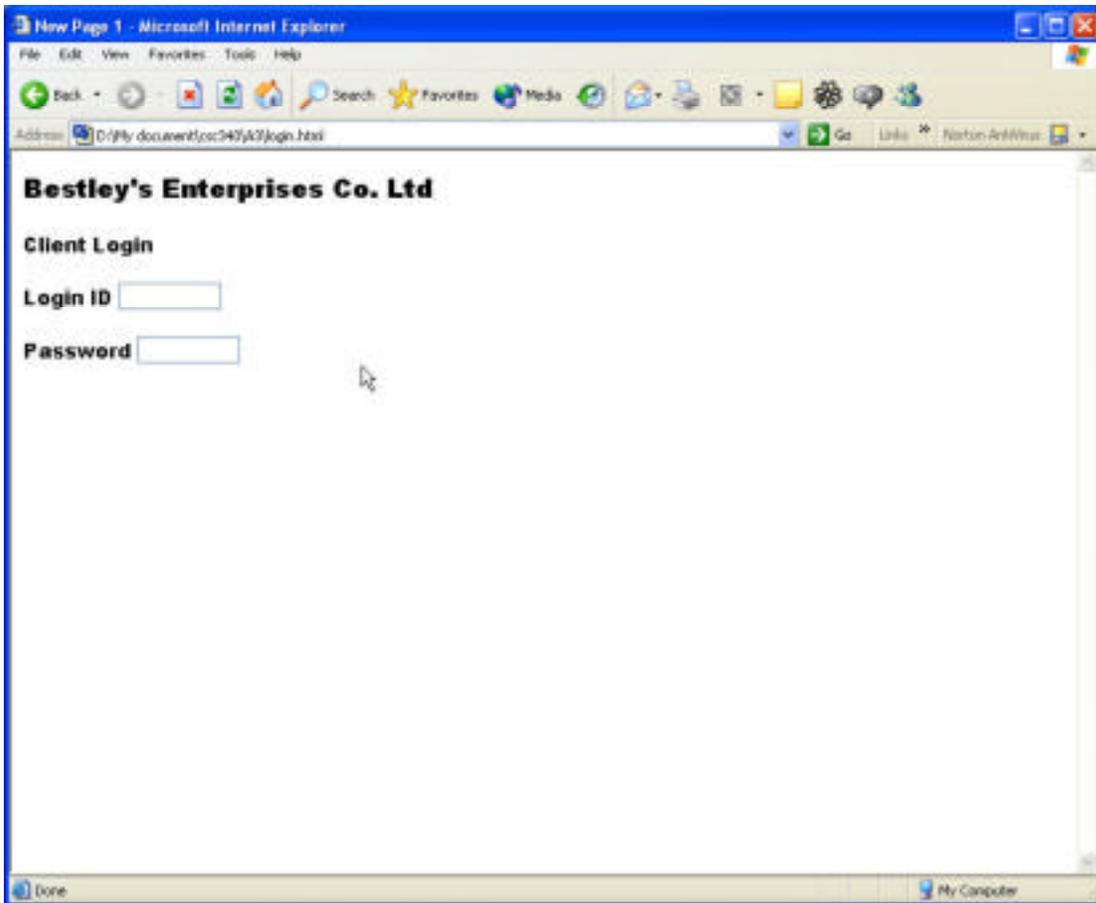
order.html

Bestley Enterprises Co. Ltd

Date : 14 April, 2002

Raw Material ID	12354	145285	14575	4567
Design ID	486534	425335	476552	945944
Quantity	0	0	0	0
Price	$5/1	$3.2/9	$0.5/9	$1.0/9
Description	8 x 2 Nylon	16 x 8 Poly	45 x 2 Nylon	16 x 4.5 Poly

Submit Query

Activitiy Record

Private Attributes:

Attributes	Description
activityCode : String	Unique code of this particular activity
description : String	Description of activity

Public Operations:

Operations	Description
getCode() : String	Return unique code of this activity
getDescription() : String	Return the description of this activity
modCode(code : String)	Modify the activity code
modDescription(description : String)	Modify the description
newActivity(activityCode : String)	Create a new instance of activity record
printActivity()	Print the activity record

Backup Device

Private Attributes:

Attributes	Description
lastBackup : Date	The date of last backup operation

Public Operations:

Operations	Description
get_Last_backup() : Date	Return last backup date
set_backup_date();	Set the current time for last backup date
accessRecord()	Access record to be back up

CD Writer

Public Operations:

Operations	Description
write_CD()	Perform backup operation

Client

Private Attributes:

Attributes	Description
address : String	Client's address
comment: String	Special comment about client
email : String	Client's email address
Fax : String	Client's fax number
Id: String	Unique client's id
name : String	Name of company
phone: String	Client's phone number
pic : String	Client's side person-in-charge
Date: int	Date of record

Public Operations:

Operations	Description
getCustomerInfo()	Get customer information
modCustomerInfo()	Modify customer information
storeCustomerRec()	Store customer record
printClient()	print the client record
selectProduct(designID: String)	Select product

Database

Private Attributes:

Attributes	Description
record : record	The set of record objects

Public Operations:

Operations	Description
backup()	Backup all the information in backup device
insert(record : Record)	Insert a new record instance into database
newDatabase()	Create a new database instance
queryRecord(recordID : String)	Query operation with recordID as parameter
remove(recordID : String)	Remove record from database with recordID as parameter
update(record : Record)	Update the record in database
createTable(record: Record)	Group Records into a table

Design Record

Private Attributes:

Attributes	Description
designFile : File	File contain the design with associate invoice Id
designId : String	Unique DesignId
instruction : String	Special instruction on design
Date: int	Date of record

Public Operations:

Operations	Description
getDesignInfo () : File	Return the file contain the design of plastic bag
modDesignInfo(Design : File)	Modify design file
printDesign()	Print the design record

Employee Record

Private Attributes:

Attributes	Description
dob : Date	Date of birth of employee
employeeID : String	Unique employee's ID
joinDate : Date	Employee's join date
name : String	Name of employee
password : String	Employee's login ID
position : String	Position of employee in company
salary : Real	Employee's monthly salary
Date: int	Date of record

Public Operations:

Operations	Description
getOrder()	Get order from client
getEmployeeInfo()	Get the information of new employee
modEmployeeInfo()	Modify employee information
storeEmployeeRec()	Store employee record

Manager Record

Private Attributes:

Attributes	Description
dob : Date	Date of birth of Manager

ManagerID : String	Unique Manager's ID
joinDate : Date	Manager's join date
name : String	Name of Manager
password : String	Manager's login ID
position : String	Position of Manager in company
salary : Real	Manager's monthly salary
Date: int	Date of record

Public Operations:

Operations	Description
getOrder()	Get order from client
getManagerInfo()	Get the information of new Manager
modManagerInfo()	Modify Manager information
storeManagerRec()	Store Manager record
design()	Design the product draft
buyMaterials()	Buy raw materials
usedMaterial()	Use up materials
getMaterialQuantity()	Count the quantity of material
printManager()	Print the Manager record
login(ID:String, password:String)	Login to database
storeRecord(invoiceid:String)	Keep order in record

ITworker Record

Private Attributes:

Attributes	Description
dob : Date	Date of birth of ITworker
ITworkerID: String	Unique ITworker's ID
joinDate : Date	ITworker's join date
name : String	Name of ITworker
password : String	ITworker's login ID
salary : Real	ITworker's monthly salary
Date: int	Date of record

Public Operations:

Operations	Description
getOrder()	Get order from client
getITworker Info()	Get the information of new ITworker
modITworker Info()	Modify ITworker information
storeITworkerRec()	Store ITworker record
backup()	Perform backup to the database
printITworker()	Print the ITworker record

login(ID:String, password:String)	Login to database
storeRecord(invoiceid:String)	Keep order in record

Worker Record

Private Attributes:

Attributes	Description
dob : Date	Date of birth of worker
workerID : String	Unique worker's ID
joinDate : Date	worker's join date
name : String	Name of worker
password : String	worker's login ID
salary : Real	worker's monthly salary
Date: int	Date of record

Public Operations:

Operations	Description
getOrder()	Get order from client
getWorkerInfo()	Get the information of new Worker
modWorkerInfo()	Modify Worker information
storeWorkerRec()	Store Worker record
buyMaterials()	Buy raw materials
usedMaterial()	Use up materials
getMaterialQuantity()	Count the quantity of material
printWorker()	Print the Worker record
login(ID:String, password:String)	Login to database
storeRecord(invoiceid:String)	Keep order in record

Inventory Record

Private Attributes:

Attributes	Description
comment : String	Special comment about inventory
quantity : Integer	The quantity of stock
rawID : String	The raw material's ID
type : String	The type of raw material
Date: int	Date of record

Public Operations:

Operations	Description

inverntoryUpdated()	Update the inventory
newInventory(rawID : String)	Create a new inventory instance with raw material ID
printInventory()	Print the inventory record

Invoice Record

Private Attributes:

Attributes	Description
clientID : Client	All neccssary information about client to facilitate this transaction
invoiceDate : Date	Invoice's creation date
invoiceID : String	The id of this invoice
tranID: Transaction Record	The set of transaction records for this invoice
Date: int	Date of record

Public Operations:

Operations	Description
getClientInfo() : Client	Return the information of client associated with this invoice
getDate() : Date	Return this invoice's creation date
getInvoiceID() : String	Return this invoice's ID
getpaidStatus() : Boolean	Return paid status
getShipmentdate() : Date	Return the date of shipment
getTotal() : Integer	Return the total of this invoice
getTransactionRecord() : Transaction Record	Return the transaction record
modClientInfo(clientRecord : Client)	Modify Client's information
modInvoiceDate (invoiceDate : Date)	Modify Invoice's creation date
modInvoiceID (inoviceID : String)	Modify Invoice's Id
modpaidStatus (status : Boolean)	Modify paid status of this invoice
modShipmentDate (shipmentDate : Date)	Modify the date of shipment
modTotal (total : Real)	Modify the total amount of invoice
modTransaction (transRecord : Transaction Record)	Modify the transaction record
newInvoice(invoiceID : String)	Create a new invoice instance with invoice ID
printInvoice()	Print invoice
displayConfirmation()	Display confirmation details

WebLogin Record

Private Attributes:

Attributes	Description
activityLog : Acitivity Record	A set of activities involved in this session
employeeID : String	Employee ID of this session
WebloginTime : Date	The Weblogin date and time
WeblogoutTime: Date	The Weblogout date and time
sessionID : String	Unique number assign to each session
Date: int	Date of record

Public Operations:

Operations	Description
getActivityLog() : Activity Record	Return the activity record
getEmployeeID() : String	Return EmployeeID of this session
getWebLogintTime() : Date	Return Weblogin time of this session
getWebLogoutTime() : Date	Return Weblogout time of this session
getSessionID() : String	Return session ID
modActivityLog(activitylog : Activity Record)	Modify the activity record
modEmployeeID(employeeID : String)	Modify the employee
modWebLogin(loginTime: Date)	Modify the login time
modWebLogout(logoutTime : Date)	Modify the logout time
modSessionId (sessionID : String)	Modify session ID
newWebLogin (sessionID : String)	Create a new instance of WebLogin Record
printWebLogin()	Print the login record

Online Order

Private Attributes:

Attributes	Description
designID : String	Product ID
orderID : String	Unique order ID
orderStatus : String	Payment status of this online order
paymentMethod : String	The method of payment
shipmentDate : Date	The date of delivery
Date: int	Date of record

Public Operations:

Operations	Description
display_Confirmation()	Display all the data associated with this order on the screen for further confirmation

login(id:String, password:String)	Login to the online system
login_Confirmation()	Return true if login is valid
request_product_selection()	Request the client to select product
displayProduct()	Display design of all products available
displayPaymentinfo()	Display information about payment
getOrderID() : String	Return the order ID
getStatus() : String	Return the status of this order
getPaymentMethod() : String	Return the method of payment
newOrder()	Create a new instance of online order
printOrder()	Print the online order
selectProduct(DesignID : String , RawID : String)	Enter selection with Design ID and Raw Material ID
submit()	Submit all the information of this order to generate a corresponding invoice

Payment

Private Attributes:

Attributes	Description
remark : String	The remark associate with this payment
clientID : String	The client ID
paymentID : String	Payment ID
InvoiceID: Invoice record	The set of invoices needed to be paid
paymentAmount : Real	The amount paid associate with this payment
paymentDate : String	The date which the payment is made
paymentMethod : String	The method to settle this balance
Date: int	Date of record

Public Operations:

Operations	Description
getChequeNumber() : String	Return the cheque number used to settle this payment
getClientID() : String	Return client ID
getInvoiceID() : String	Return invoice ID
getPaymentAmount() : Real	Return payment amount
getPaymentDate() : Date	Return the date which payment is made
getPaymentId() : String	Return payment ID
getPaymentMethod() : String	Return the method of payment
modChequeNumber(number : String)	Modify the cheque number
modClientID(code : String)	Modify client ID
modInvoiceID(code : String)	Modify invoice ID

modPaymentAmount(amount : Real)	Modify amount of this payment
modPaymentDate(date : Date)	Modify the date of payment
modPaymentMethod(method : String)	Modify the payment method
modReceiptID(code : String)	Modify receiptID
printPaymentInfo()	Print payment information

Raw Material Record

Private Attributes:

Attributes	Description
rawID : String	Unique raw material ID
quantity : int	Information about this raw material's quantity
supplierID : Supplier Record	Information of supplier
Date: int	Date of record

Public Operations:

Operations	Description
getMaterialInfo()	Get the material information
modMaterialInfo()	Modify material information
printRaw()	Print the raw material record

Software Interface

Public Operations:

Operations	Description
backup()	Perform the backup operation
generateInvoice() : Invoice	Prepare an invoice
enterClient()	Display interface for input client's info
enterDesign()	Display interface for input design's info
enterEmployee()	Display interface for input employee's info
enterInvoice()	Display interface for input invoice's info
enterPayment()	Display interface for input payment's info
enterRaw()	Display interface for input raw material's info
enterSupplier()	Display interface for input supplier's info
enterTrans()	Display interface for enter transaction's info
login()	Display login interface
newRecord() : Record	Create a record instance
printStatement()	Print a statement

Report Record

Private Attributes:

Attributes	Description
ReportID : String	report ID
ReportDate : String	Report generated date
EmployeeID : String	Employee who made this report
Date: int	Date of record

Public Operations:

Operations	Description
getReportInfo()	Get report infomation
modReportInfo()	Modify report information
printReport()	Print the report

Supplier Record

Private Attributes:

Attributes	Description
supplyRawRecordID : String	Raw material ID
supplierContact : String	Supplier's contact person
supplierID : String	Unique supplier ID
supplierfax : String	Supplier's fax number
supplierphone : String	Supplier's phone number
suppplierName : String	Supplier's company name
Date: int	Date of record

Public Operations:

Operations	Description
getSupplierInfo()	Get supplier infomation
modSupplierInfo()	Modify supplier information
printSupplier()	Print the supplier record

Transaction Record

Private Attributes:

Attributes	Description
designID : String	IDcode of design for this transaction
paymentID : String	This transaction associate payment ID
quanity : Integer	The quantity of this transaction
rawID : String	This transaction associate raw material

	ID
tranID : String	This transaction ID
Date: int	Date of record

Public Operations:

Operations	Description
getDesignID() : String	Return design ID of this transaction
getQuanitity() : Integer	Return the quantity of integer
getRawID() : String	Return raw material ID of this transaction
modDesignID(designID : String)	Modify design ID
modQuanitity(quantity : Integer)	Modify quantity
modRawID(rawID : String)	Modify raw material ID
modTranID(tranId : String)	Modify transaction ID
newTran(tranID : String)	Create a new instance of transaction record
printTransaction()	Print transaction record
searchClient(clientid)	Search for a client in the client database
searchProduct(designed)	Search for a product in design database
searchMaterial(rawid)	Search for a material in material database
searchShipmentDate()	Search for the shipment date

Appendix I : Details on Group Meeting

Group Meeting 1:

Meeting on Apr 5th, 2002 at 3:30pm

Details: - Discuss on the process and time line on how we are going to work with this assignment.

Group Meeting 2:

Meeting on Apr 6th, 2002 at 6:00pm

Details: - Divide the work fairly between group members.

-Talk about some functional and non-functional requirements.

-Start to write down the global architecture for the new computerize and online purchasing system.

Group Meeting 3:

Meeting on Apr 7th, 2002 at 3:30pm

Details: - Modify the use-case, sequence and state diagram based on the class diagram that we have been previously developed.

- We decide to have a meeting with the company and show them about our ideas and see if they want to make any adjustment.

Group Meeting 4:

Meeting on Apr 10th, 2002 at 2:00pm

Details: - After meeting with the company, we know that we are on the right track, and Mr. Tang seems to appreciate to our work.

- Then we continue our work on the user interface design, web page design, diagrams and supporting document.

Appendix J : Interview

Meeting with Bestley's C.E.O

Date: April 8th, 2002

Purpose: To talk about user interface and web page design and to get opinion about additional functionalities.

Feedback: We have agreement on the requirements of the suggested user interface and web page. The cost of the system is also within the budget of the company. Mr. Tang was though concerned with the advertising effects on increase of sales.

APPENDIX K: Entity Relationship Diagram

APPENDIX L: Class Diagram After Restructure

Appendix M : Team Report form

Team Report Form

[illegible subtitle text]

Name	Contribution	Signature
[illegible name]	33.3 %	[signature]
[illegible name] (participant)	34.3 %	[signature]
[illegible name]	32.5 %	[signature]