

P-1089

Kumaraguru College of Technology

Department of Computer Science and Engineering

Coimbatore – 641006.

September 2003

L.P.G Stock Accounting System

Project Work done at

Global Software Ltd., Chennai

PROJECT REPORT

**Submitted in partial fulfillment of the
Requirements for the award of the degree of**

M.Sc Applied Science (Software Engineering)

Bharathiar University, Coimbatore

Submitted by

S.VIVEKA

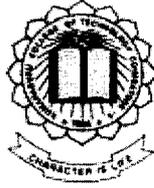
Reg. No : 0037S0116

Internal Guide

**Mr. Manikandan M.C.A.,
Dept. of computer Science & Engineering,
Kumaraguru College of Technology,
Coimbatore**

External Guide

**Mrs. Meenakshi
Global Software Ltd.
Chennai**



Department of Computer Science and Engineering

Kumaraguru College of Technology

Coimbatore – 641006

CERTIFICATE

This is to certify that the project entitled

L.P.G Stock Accounting System

Done by

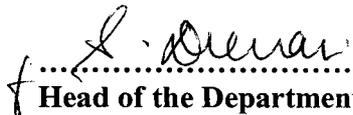
S. VIVEKA

Reg. No. : 0037SO116

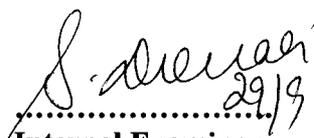
**Submitted in partial fulfillment of the requirements for the award of the degree of
M.Sc. Applied Science (Software Engineering) of Bharathiar University,**

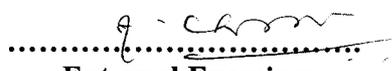
Coimbatore


.....
Guide


.....
Head of the Department

Submitted for the University examination held on .29.9.03.....

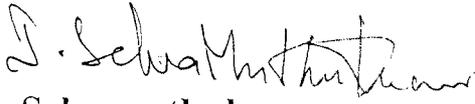

.....
Internal Examiner


.....
External Examiner 29/9

16/09/03

CERTIFICATE

This is to certify that Ms. Viveka .S. of IV th year Msc
(S/w Engineering) student of Kumaraguru College of
Technology , Coimbatore , has done her project on "L.P.G
Stock Accounting System " under my guidance during
June 2003 to September 2003 . We find her project work
satisfactory.

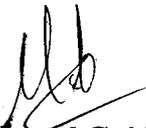

J. Selvamuthukumar
(Deputy – Manager Training)


Meenakshi Ramesh
(Project Guide)

Declaration

DECLARATION

I hereby declare that the project entitled **L.P.G Stock Accounting System**, Submitted towards the fulfillment of M.Sc Applied Science (Software Engineering) from Bharathiar University has not formed the basis of any degree, diploma or association of similar titles. The project work was done independently by me under the guidance of **Mr. Manikandan** (Internal Guide) and **Mrs. Meenakshi** (External Guide).



Internal Guide

S.Viveka

Acknowledgment

ACKNOWLEDGEMENT

I express my sincere gratitude to **Dr. K.K Padmanabhan**, Ph.D., esteemed Principal, Kumaraguru College Of Technology for giving me this opportunity to do a project work.

I express my profound gratitude to **Mr. S.Thangasamy** HOD, CSE Department, Kumaraguru College of Technology for his support in doing this project.

I am grateful to **Mr. Selvamuthu Kumar**, Deputy training manager of Global Software Ltd. for giving me an opportunity to do a project in their esteemed organization.

I express my sincere thanks to **Mr. Suresh Kumar** and **Mrs. Meenakshi**, Global Software Ltd. for their support and guidance in doing this project.

I express my deep sense of gratitude and indebtedness to **Mr. Manikandan**, Lecturer Kumaraguru College of Technology for his invaluable guidance throughout the length of this project.

I thank my **Beloved Parents** who have been a pillar of support in all my endeavors, my friends and the department teaching and non-teaching staff for their support in completing this project.

Synopsis

SYNOPSIS

The project work entitled '**L.P.G Stock Accounting System**' is an application system developed for **Global Software Ltd**, Chennai.

The L.P.G Stock Accounting System consists of Consumer Master, Billing, Booking, Stock, Purchase, Complaints and Employee Details.

The system will take customers for booking of regulators or cylinders with interval of fifteen days. The system will provide information regarding the availability of Cylinders, Regulators.

It will also keep track of the waiting list at any point to determine the purchase schedule. The waitlist number for every customer is also shown on enquiry.

The system will also keep track of new customers and also the Transfer In and Transfer out Customers to determine the billing as also the nature of complaints received and the response time for attending to such complaints.

In addition to maintaining the stock, the system will also maintain the employee's details for the distributor.

SYNOPSIS

The project work entitled '**L.P.G Stock Accounting System**' is an application system developed for **Global Software Ltd**, Chennai.

The L.P.G Stock Accounting System consists of Consumer Master, Billing, Booking, Stock, Purchase, Complaints.

The system will register transactions of the company through bank customers for booking of regulators or cylinders with interval of fifteen days. The system will provide information regarding the availability of Cylinders, Regulators and also the monthly sales.

It will also keep track of the waiting list at any point to determine the purchase schedule.

The system will also keep track of the Transfer In and Transfer out Customers to determine the billing as also the nature of complaints received and the response time for attending to such complaints.

In addition to maintaining the stock, the system will also maintain the employee's details.

CONTENTS

Page No.

1. Introduction	
1.1 Project Overview	1
1.1.1 Purpose of the project	
1.1.2 Scope of the Project	
1.2 Organization Profile	2
2. System Analysis	
2.1 Existing System	3
2.2 Proposed System	4
2.3 User Characteristics	4
3. System Environment	
3.1 Product Definition	5
3.2 Project Plan	5
3.3 Working Environment	6
3.4 Languages used – Overview	7
4. System Design	
4.1 Design Details	11
4.2 Input/Output Screen Design	16
4.3 Database Design	17
4.4 Process Design	20
5. Testing	27
6. Implementation	29
7. Conclusion	30
8. Future Enhancement	31
9. Appendix	32
10. References	60

Global Software Ltd.

Introduction

1. Introduction

1.1 Project Overview

1.1.1 Purpose of the System

The aim of the project is to develop a stock accounting system for the L.P.G company (Domestic Services) using IBM software's like JCL, CICS, TSO under the MVS operating system. The system will eliminate the drawbacks of the existing system.

1.1.2 Scope of the Project

- ✓ The proposed system will keep track of the customer's of domestic gas for a dealer of L.P.G.
- ✓ It maintains the purchase, stock and the employee's of the dealer details also.
- ✓ The system takes in customers for booking and keeps track of Transfer In and Transfer Out customers.
- ✓ The proposed system will specify the reorder level, total stock of cylinders and regulators and also the number of waiting list customers.
- ✓ The system will also prepare bills for the delivered customers.
- ✓ The system registers the complaints from customers and also maintains information when the complaints were attended and charges for the same.

Global Software Ltd.

1.2 Organization Profile:

Global software limited (GSL) is promoted by an industrial group in the IT area. It is a technology focused multinational company that focuses on contemporary ESM solutions anchored by quality.

With a world class research and development center for operating systems, databases, networks and enterprise security, GSL provides ESM solutions from min-size to large organizations, portals and ISPs.

The company has offices in US, UK, Singapore, Mauritius and India with 150 experienced ESM professionals. It is backed by core technology teams with 100+ many years of experience in systems software and networking technologies.

Global Software Ltd, India, is a backend system integration company, focusing on enterprise systems management,(ESM).The company has excellent resources to offer the entire range of backend systems integration services in IT with specialization in the following areas.

- * Managed services.
- * Operating system services
- * Network services
- * ESM solution consultancy

Our company is unique with 100% certified, thoroughly experienced, highly qualified professionals offering tangible, scalable ESM solutions to achieve increase in service deliverables, sound knowledge and vast experience to handle heterogeneous complexity of multiple systems.

With its unique Competency center,(the only one of its kind in Asia Pacific), Global's Competency center- Hardware and Software

GLOBAL continually update the competency center in line with market changes. The ESM competency in center in Chennai is a true world -class infrastructure with state-of -the-art equipments. The company has high-speed links to internet and to all its world wide offices. The links can be extended to the clients location for off shore support/remote management there by providing cost effective solutions.

GSL is an IBM/Tivoli business partner, Microsoft certified solution provider and authorized prometric testing center.

Global Software Ltd.

System Analysis

2. System Analysis

2.1 Existing System

The existing system uses PS (Physical sequential) data sets as it called in MVS (Multiple Virtual Storage) operating system terms. The Physical Sequential Data set is a simple file.

- Data is stored in the form of records in the files.
- It was not easy to retrieve records.
- Duplicate records were also created.
- No security was provided for handling the files.
- The system was not user interactive.
- The records are to be entered only into the file, each and every field length has to be remembered to insert a single record into the file.
- The record fields are not differentiable as they are stored continuously and this causes difficulty in reading the data as there is no other means for reading the stored records.
- The existing system does not handle the employee's details of the dealer.

2.2 Proposed System

- The proposed system uses VSAM (Virtual Storage Access Memory) Data Sets.
- It uses Key Sequential Data Sets (KSDS) a VSAM Data set to store the data.
- A field is set as the primary key and it automatically eliminates duplicate record entries.
- Records are stored in the ascending order according to the value of the primary key.
- Records can be processed, sequentially, randomly or dynamically.
- The system uses CICS (Customer Information Control System) as the front end tool for user interaction.
- The system provides security for using the data.
- The proposed system also handles the employee details and the reorder level.
- The system also maintains the complaints from customers and when the complaints were attended and charges for the same.

2.3 User Characteristics

The system is used by dealers of L.P.G for domestic services. The system will be very helpful for them to maintain the booking and stock details and waiting list of customers instead of doing it manually.

Global Software Ltd.

System Environment

3. System Environment

3.1 Product Definition

The product has to maintain the customer's of a dealer for L.P.G, look after their booking's for cylinders an regulators, generate bills for delivered customers, maintain the stock details and number of waiting list customers and specify the reorder level for purchase. The product has to maintain the complaints history of customers like gas leakage, regulator fault and date of attending of such complaints and the charges for the same.

3.2 Project Plan

- ✓ Understanding the concepts and software's used in Mainframes like JCL, VSAM, CICS, TSO and languages like COBOL, JCL etc.
- ✓ Studying the existing system and its drawbacks.
- ✓ Analyzing and understanding the problem statement.
- ✓ The features to be incorporated in the system are to be clearly worked out.
- ✓ Working out the different modules and their functions.
- ✓ Implementation of the different modules.
- ✓ Testing the modules separately.
- ✓ Integrating the modules.
- ✓ Testing the whole system.
- ✓ Analyzing the loop holes and fixing them.

3.3 Working Environment

Hardware Requirements :

- Processor - IBM IS 3006
- Model - BO1
- Hard disk - 78 GB
- Monitor - IBM SDT 3270

Software Requirements :

- Operating System - Operating system/390, MVS
- Version - 2.8
- Language - Cobol, JCL (Job Completion Language)
- Front End - CICS (Customer Information Control System)
- Back End - VSAM

3.4 Software's and Languages used – a Quick Guide

Time Sharing Option (TSO)

Time-sharing Option (TSO) is an option of the MVS operating System that all allows users to interactively share computer time and resources.

It is a tool with which you can

- Develop and maintain programs in languages such as COBOL, REXX, PL/I etc.
- Create an office environment
- Process data
- Communicate with other TSO users
- Access the MVS operating system

Time Sharing

- Computer time is divided into time slots
- Each terminal user is allocated a time slot
- Can support multiple users

Features

Program Development Functions (PDF)

- Create and edit program source libraries
- Compile a source program Link-edit a program
- Test a Link-edit program. Send output to a printer.
- View compiler and linkage-editor output

Global Software Ltd.

- Data set Management Functions
- Dynamically allocate data users
- List data sets
- Copy data sets
- Scratch data sets
- Rename data sets
- List catalog entries
- Use VSAM commands

Batch Job Facility

- Submit a job for background processing
- Monitor the progress of a background job
- View output from a background job
- Route output to a local or remote site
- Cancel submitted job

JOB CONTROL LANGUAGE

JCL is IBM's system product used for entering jobs into the system for background execution. JCL is the means by which a user program requests resources from the operating system. JCL acts as an interface between the program and the operating system.

For every job that you submit, you will need to tell MVS where to find the appropriate input, how to process that input (that is what program to run) and what to do with the resulting output.

You use JCL to convey this information to MVS through a set of statements known and job control statements.

Global Software Ltd.

Within each job, the control statements are grouped into job steps. A job step consists of all the control statements needed to run one program.

Every job must contain minimum of two control statements

A **JOB statement**, to mark the beginning of a job and assign a name to the job. It is also used to provide certain administrative information, including security.

A **EXEC (execute) statement**, to mark the beginning of a job step, to assign a name to the step, and to identify the program or procedure to be executed in the step.

VSAM (Virtual Storage Access Method)

VSAM is by far the most commonly used access method on MVS systems. VSAM provides efficiency improvements and a catalog facility that centralizes information about all VSAM data sets.

Very little JCL coding is needed for processing VSAM file. The DD statement is a one-line statement needing only the DSNNAME and DISP parameters.

Records can be accessed through one or more secondary keys using alternate indexes built and maintained by VSAM.

VSAM Dataset Types

Entry Sequenced Dataset (ESDS)

ESDS contains records in the order in which they are entered. Records are added to the end of the dataset, and can be accessed sequentially

Key Sequenced Dataset (KSDS)

KSDS contains records in ascending collating sequence. Records can be accessed by a field called a key, or by a relative byte address (RBA- Relative position of the record from the beginning of the dataset.), or sequentially.

Linear Dataset (LDS)

LDS contains data that has no record boundaries. The manipulation of the data is completely controlled by the user.

Global Software Ltd.

System Design

Global Software Ltd.

Relative Record Dataset (RRDS)

RRDS contains records in the order of relative record number. These records can be accessed only by this number.

CICS

Customer Information Control System (CICS) was developed in 1960 by IBM. It is an Online Control System, and a General Purpose Database/Data communication Control System.

This runs as a subsystem on IBM mainframes. It provides an environment to handle all the special requirements of online processing.

CICS acts as an interface between the operating system and the database products.

CICS Terminology

TASK

A basic unit of work, which is scheduled by Operating System is called Task. CICS as a whole is one of the many tasks under OS. CICS schedules a unit of work within a CICS region. This is called a CICS task. Each execution of a transaction is a task.

TRANSACTION

Transaction is an entity, which initiates execution of a task. Under CICS, a CICS transaction is a CICS task which is initiated through a transaction identifier (1 to 4 characters). This transaction identifier must be registered in the Program control table (PCT).

MAPS

To make the application program device independent and format independent, CICS provides Basic Mapping Support (BMS) to deal with the formatted screen operations. Screens defined through BMS are called "MAPs"

BMS is the facility that moves data streams to and from the terminal.

4. System Design

4.1 System Design – An Overview

The whole system can be split into individual modules, each of which has its own importance. The main core of the system lies in maintaining the customers details along with their booking details and maintenance of the stock. These modules only serve user's request, other sub-modules give support to these main modules and have their own importance.

The system contains eight modules. They are

- Main Menu
- Consumer Master
- Purchase
- Enquiry
- Booking
- Billing
- Complaints
- Employee Details

Global Software Ltd.

Module Details

Main Menu

This program will display the Main Menu Screen when the Transaction ID (GCxx) is invoked in the CICS region. The Main Menu will have different options as provided in the system that is, different tasks like booking, billing etc. on choosing the valid option the control should be transferred to corresponding program for execution.

Proper Function Keys are provided to select the options and cross validations to trap the wrong keys are provided in the program.

Consumer Master

This module will display the options for updations for consumer master. The options should include provision for the following :

Function Key	Function
Pf1	Clear
Pf2	Add
Pf3	Modify
Pf4	View
Pf9	Exit

Add option should provide for addition of new consumers or Transfer In consumers from other distributors.

The customer code is system generated and the Transfer Out number for customers to other distributors will also be system generated.

In the Modify Option, provision for changing the address details and also to allow Transfer Out customers to other distributors.

Global Software Ltd.

In the view option the system will display the details of the customer given the customer code.

The clear option will clear all the entered fields for the next entry.

The exit option will help us to come out of the customer master screen to the Main Menu.

Purchase

This module receives as input the purchase details for the purchase of both cylinders and regulators.

The information captured are

- Bill No
- Bill Date
- Product Code (11 – Cylinders, 12 – Regulators)
- Quantity
- Supplier Name

Here Bill No is the key field for retrieving the purchase records. The supplier field will contain the supplier names like IOC, BPCL etc. and the Quantity field contains the number of cylinders or regulators got from the suppliers.

Functions performed :

- Validation of the product code is done
- Quantity should be always greater than zero. This has to be checked.
- Stock is to be updated in the stock table.

Global Software Ltd.

Enquiry

This module displays the total stock available in the store for cylinders and regulators. The information will be captured from the stock table and transaction table. The total number of wait list customers will also be displayed.

Booking

This module accepts booking of cylinders and regulators from customers and creates a transaction record for every booking. The quantity will be one by default and the rate will be standard.

Validations to be Done:

- Booking should not be taken for Transfer Out Consumers
- Booking should not be accepted if already booked but not supplied.
- Consumer code validation
- Booking not to be accepted within 15 days of earlier booking.

Billing

This module prepares bills for customers depending on the cylinders and/or regulators supplied. Given the customer code the system will generate the bill for the customer.

The information is processed from the transaction file.

Complaints

In this module complaints reported by consumer should be registered according to the nature of complaint with date and time of complaint reported

Once complaint has been attended, the table should be updated indicating the date and time attended and charges collected if any.

Global Software Ltd.

The common complaints noted are leakage of gas, regulator fault.

Employee details

This module is added on request by the user to maintain the details of the employee's of the user that is, distributor.

Information's captured are employee code, name, date of joining, designation, salary etc.

The employee code will be system generated.

Functions keys are provided to add, modify, clear, view the details of the customer.

Function key	Function
Pf2	Add
Pf3	Modify
Pf4	View
Pf1	Clear
Pf9	Exit

4.2 Input Design

Input design is a part of the overall system design. It is the phase that requires careful attention of the developer. The main objectives during this phase are

- Achieve high level accuracy
- Ensure input is free of ambiguity

The input design involves converting the user-originated inputs into computer based format. The input design must make it easier for the user to enter the data's and must be in a fashion which is easy to understand.

The input design must filter wrong inputs in the earlier stage itself as much as possible and make the input logical error free. This helps us to filter data in the input stage itself that are otherwise entered into the database and will be inconsistent.

Input design involves capturing of the data, verifying and then passing them on to the system. After choosing input medium, attention is focused on error-handling, control and grouping and validation procedures.

During the input screen designing care should be taken to make our system extremely user friendly and organize our screens and data fields such that the possibilities of making error are maintained.

Each and every field entered by the user is checked using validation procedures before storing them into the database.

Output Design

Output of a system is the heart of all software design. The developed system will be successful only if the output desired by the user is got from the system.

The system must provide various reports in various required formats to the user.

Characteristics of the output design

- Fit for the user needs – the output produced must be on time.
- The outputs should abstract the complexity of the system.

Global Software Ltd.

- The output must be easy to understand and accurate.
- Output should be properly formatted.

4.3 Database Design

Data are the raw facts that we use to represent information. Processed data is information. Data must be manipulated (Organized, Formatted, Summarized, etc) before it can be used.

Database is used to store data in an organized fashion, which allows us to access and manipulate data. Different techniques and data structures are used to organize and manipulate data in the database.

The system uses VSAM (Virtual Storage Access Memory) Data Sets. In VSAM there are four data set types. They are Entry Sequence dataset, Key sequence dataset, Linear dataset and Relative Record dataset.

The system uses Key Sequence dataset, in which a field is taken as the primary key. The value of the primary key occurs only once in the whole dataset. Many alternate keys can also be set for a single dataset.

When a primary key is set for a dataset, duplicate records can be easily avoided. The duplicate record will not be entered into the dataset, it will be automatically indicated as responses after the execution of the command.

Table Specifications for Stock Accounting System

1. Consumer Master

Column Name	Field Type	Attribute
Cons code	Char(4)	Primary key
Addr1	Varchar(20)	Not Null
Addr2	Varchar(20)	Null
Place	Varchar(20)	Null
Phone	Char(7)	Null
TVI No	Char(5)	Null
TVI Date	Date	
TVO No	Char(5)	Null
TVO Date	Date	
Cyl Deposit	Decimal(6,2)	Null
Reg Deposit	Decimal(5,2)	Null

2. Stock Table

Column Name	Field Type	Attribute
Op Balance	Small Int	Null
Recpt	Char(1)	Not Null
Quantity	Smallint	Not Null
Supplier	Char(5)	Not Null

3. Transaction Table

Column Name	Field Type	Attribute
Cons Code	Char(4)	Not Null
Booking Date	Date	Not Null
Qty	Decimal(1,0)	Not Null
Status	Char(1)	Not Null
WaitList No	Smallint	Not Null

Global Software Ltd.

4. Complaint Table

Column Name	Field Type	Attribute
Cons Code	Char(4)	Not Null foreign Key
Doc No	Small int	Not null
Doc Date	Date	Not Null
Type of Comp	Char(2)	Not Null
Date Attended	Date	Not Null
Charges	Decimal(5,2)	Null

5. Purchase Table

Column Name	Field Type	Attribute
Bill-No	Char(4)	Primary Key
Bill-Date	Date	
Product Code	Small int	Not Null
Quantity	Small Int	Not Null

6. Employee Table

Column Name	Field Type	Attribute
Emp code	Smallint	Primary Key
Emp name	Char(20)	Not Null
Date of Joining	Date	
Designation	Char	Not Null
Salary	Smallint	Not Null

Global Software Ltd.

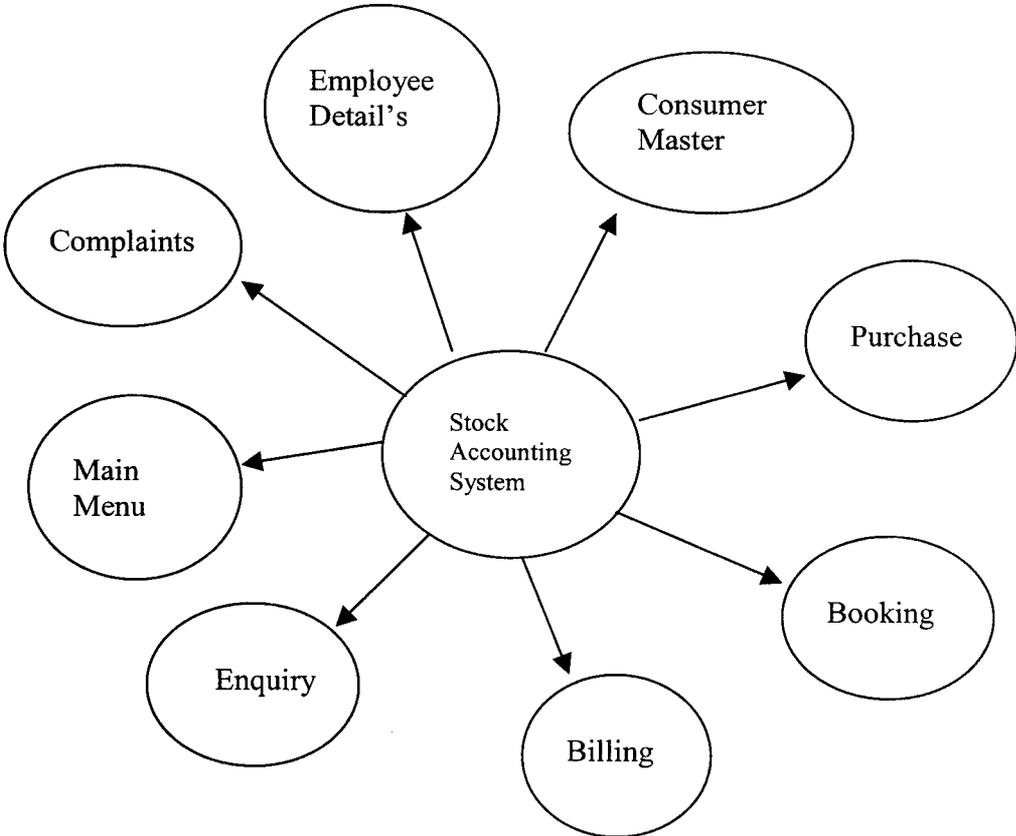
4.4 Process Design

Designing the process of the system is its main core. In the process design, details about how the data flows is to clearly stated.

The Context diagram shows the basic entities involved in the L.P.G Stock Accounting System. The Consumer Master, Purchase, Enquiry, Booking, Billing, Complaints and Employee Details are the main entities.

The Data Flow Diagram shows how the data flows through the system within different modules and to and from the terminal.

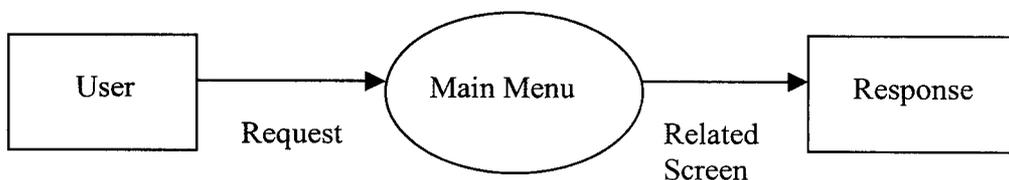
CONTEXT DIAGRAM



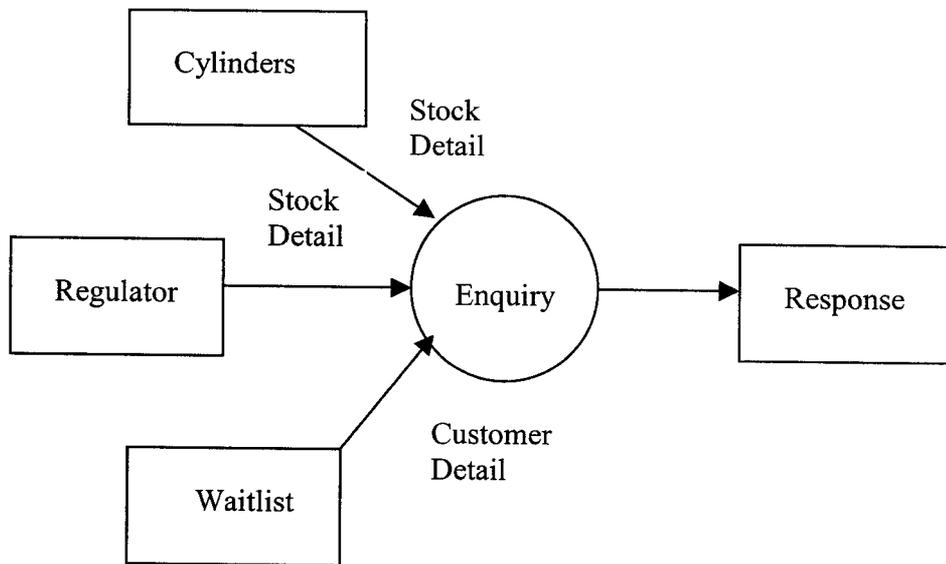
Global Software Ltd.

Data Flow Diagram

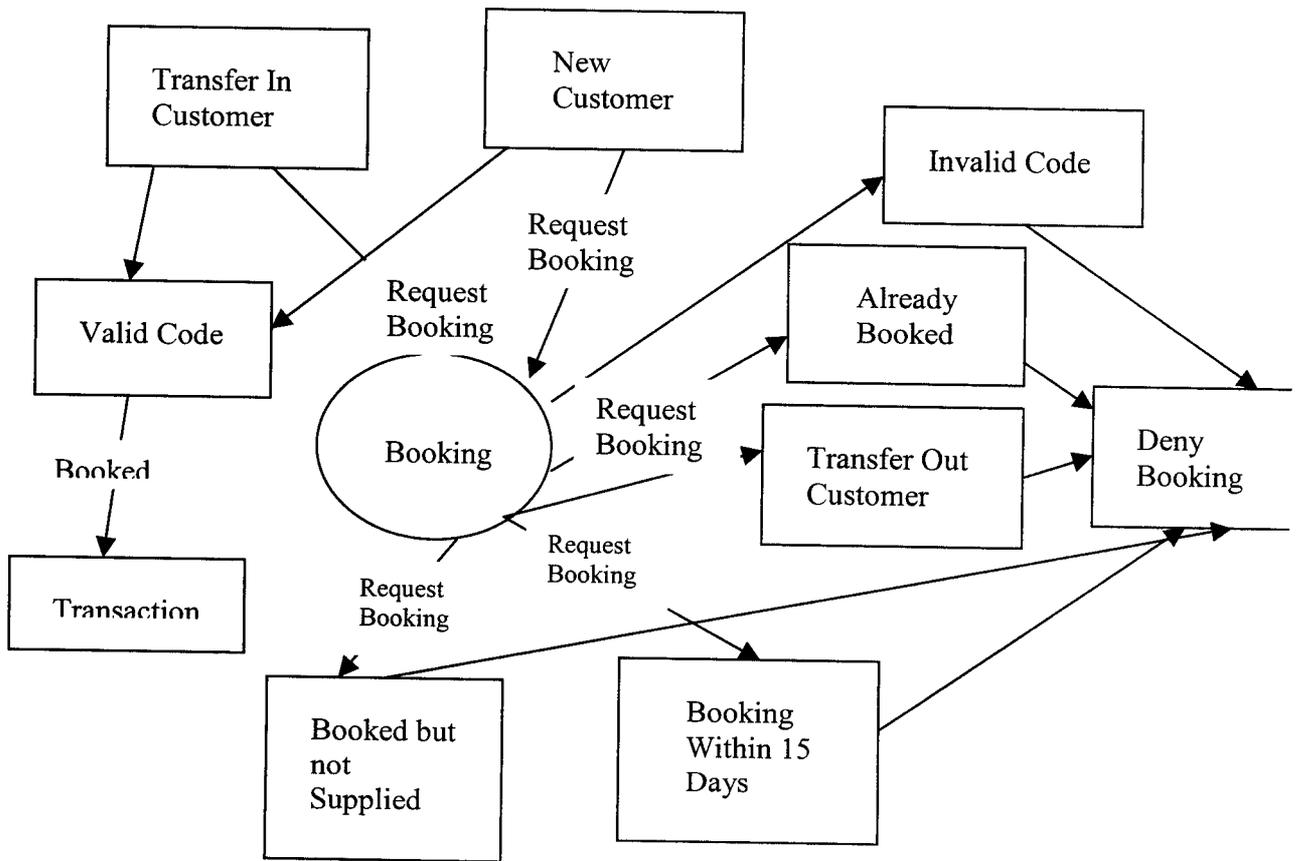
Main Menu



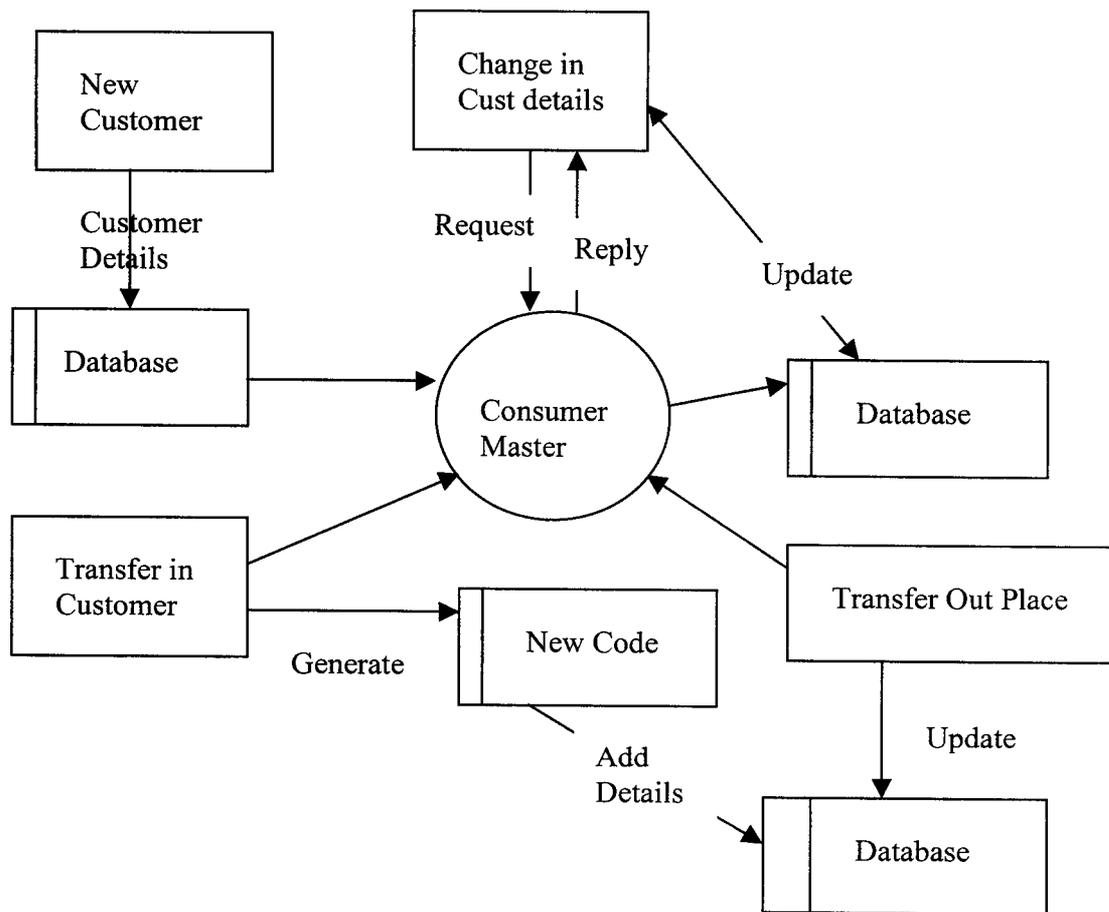
Enquiry



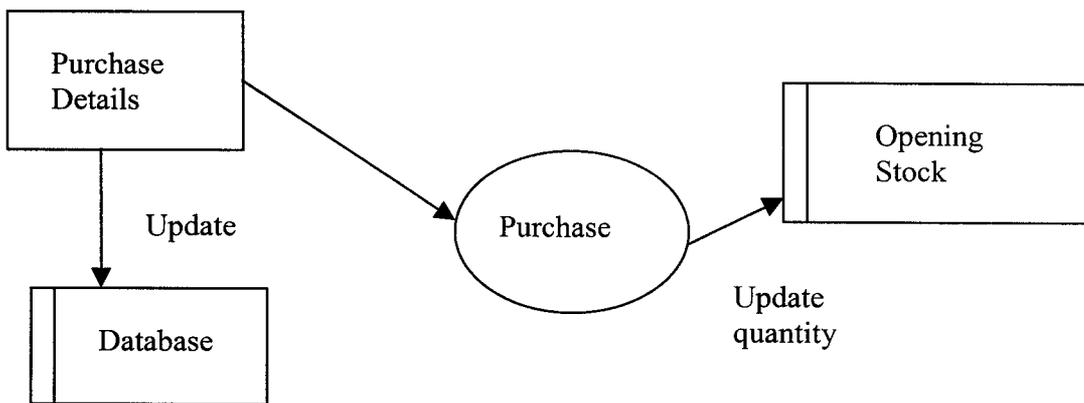
Booking



Consumer Master



Purchase



Global Software Ltd.

Testing

5. Testing

5. Testing

Software is only one element of a larger computer based system. Ultimately Software is incorporated with other system elements (Ex. New hardware) and a series of system integration and validation tests are conducted.

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer based system.

Testing presents an interesting anomaly for the software development. The testing phase creates a series of test cases that are intended to 'Demolish' the software that has been built.

A good test case is one that has a high probability of finding an as yet undiscovered error. A successful test is one that uncovers an as yet undiscovered error.

Testing process breaks applications down in to two main parts : Unit Testing and System testing. In Unit testing, the modules of the system are tested as individual units. Each unit has definite input and output parameters and often a definite single function. In System testing, the system is tested as a whole; that's inter communication among the individual units and functions of the complete system is tested.

5.1 Test Scenarios

Test case scenario is a single or a sequence of steps of checks that are made on the system to check how the system reacts for each scenario and recheck the points where the system is considered sensitive. Here are a few sample scenarios.

Global Software Ltd.

Sample Scenarios :

SCENARIO 1

- A new customer is added to the customer list

SCENARIO 2

- An invalid Customer tries to book for cylinders or regulators

SCENARIO 3

- A purchase is done with wrong product code

SCENARIO 4

- Booking is done within 15 days of earlier delivery.

Global Software Ltd.

Implementation

6. Implementation

6. Implementation

Implementation is concerned with those tasks leading immediately to a full operational system. It involves all the people like programmers, users and operations management, but its planning and timing is a prime work of system analysis. The last stage of implementation is testing the whole system to user's satisfaction.

Before implementing this system in any environment it is necessary for us to customize the system for that client. We need to create transaction id for the user to work in the user environment. We will have to make change to the existing architecture in a way easily maintainable to the user.

Before implementing the System, it's forced in to many severe testing phases. After the system clears all the tests, it's released for implementation. After the data has been initially set, the system is ready for use. The implementation type or the change over technique from the existing system is a step by process .

First a module in the part of the system is implemented and checked for suitability and the efficiency. If the user related to the particular module is satisfied, the next step of implementation is processed with. The modules related to the previous modules are implemented.

Global Software Ltd.

Conclusion

7. Conclusion

7. Conclusion

The main core of the project is to develop a system that is user friendly. The system is very helpful to the user as input data can be keyed in easily. The data processed by the system is also conveyed to the user in a very understanding manner. Function keys are provided to help the user to quickly get the required result. The main power of the system is its use of mainframe technologies and a flexible solution that can be modified to fit in new requirements and types in the future. Since each of the modules exists as separate entities, any given module can be used as a plug-in module into another system with a similar need.

Global Software Ltd.

Future Enhancement

8.Future Enhancements

8. Future Enhancements

The core competence of this product lies in the factor that it is totally expandable and flexible to the latest and future technologies. The product has been developed with the present working condition and environment in mind. The Current environment is a fast growing area and new features, new technologies and different work styles are expected. Hence this software has been developed with near future needs in mind and it has appropriate slots for any future modifications. The product can be modified to support more additions to the functionality. The flexible nature of the system and the technologies used make the system expandable and scalable.

Future modifications though are limitless the immediate step towards future enhancement would be to make it Compatible. Future enhancement in terms of customization, scalability , compatibility and technology has been made widely possible due to the style of design adopted and the strict adherence to the design, makes the system even more flexible for future enhancements.

Global Software Ltd.

Appendix

9. Appendix

Sample Coding

TRG017.PROJECT.PGMS(BOOKMAP)

```
000100 MAP017 DFHMSD TYPE=MAP,MODE=INOUT,LANG=COBOL, X
000200 STORAGE=AUTO,TIOAPFX=YES
000300 BOOKHD DFHMDI SIZE=(24,80),CTRL=(PRINT,FREEKB)
000310 DFHMDF POS=(04,35),ATTRB=(ASKIP,NORM),LENGTH=07, X
000320 INITIAL='BOOKING'
000330 DFHMDF POS=(05,34),ATTRB=(ASKIP,NORM),LENGTH=09, X
000340 INITIAL='-----'
000350 DATE DFHMDF POS=(02,65),ATTRB=(ASKIP,NORM),LENGTH=8
000360 DFHMDF POS=(02,75),ATTRB=(PROT,NORM),LENGTH=1, X
000370 INITIAL=' '
000380 TIME DFHMDF POS=(03,65),ATTRB=(ASKIP,NORM),LENGTH=8
000390 DFHMDF POS=(03,75),ATTRB=(PROT,NORM),LENGTH=1, X
000391 INITIAL=' '
000400 DFHMDF POS=(07,15),ATTRB=(ASKIP,NORM),LENGTH=06, X
000500 INITIAL='CODE : '
000600 CODE DFHMDF POS=(07,24),ATTRB=(UNPROT,FSET,IC),LENGTH=04,
00700 INITIAL='___'
000800 DFHMDF POS=(07,30),ATTRB=(PROT,NORM),LENGTH=1,
000900 INITIAL=' '
000910 DFHMDF POS=(10,15),ATTRB=(ASKIP,NORM),LENGTH=24,
000920 INITIAL='BOOKING DATE (DD/MM/YY): '
001000 BDATE DFHMDF POS=(10,42),ATTRB=(UNPROT,FSET),LENGTH=8,
001100 INITIAL='_____'
001200 DFHMDF POS=(10,54),ATTRB=(PROT,NORM),LENGTH=1,
001300 INITIAL=' '
001310 DFHMDF POS=(13,15),ATTRB=(ASKIP,NORM),LENGTH=10,
001320 INITIAL='QUANTITY : '
001400 QTY DFHMDF POS=(13,28),ATTRB=(UNPROT,FSET),LENGTH=02,
001500 INITIAL='__'
001600 DFHMDF POS=(13,31),ATTRB=(PROT,NORM),LENGTH=1,
001700 INITIAL=' '
```

Global Software Ltd.

001701 DFHMDF POS=(16,15),ATTRB=(ASKIP,NORM),LENGTH=08,
001702 INITIAL='STATUS :'

002196 STATUS DFHMDF POS=(16,26),ATTRB=(UNPROT,FSET),LENGTH=1,
X

002197 INITIAL='_'
002198 DFHMDF POS=(16,28),ATTRB=(PROT,NORM),LENGTH=1,
X

002199 INITIAL=''
002200 DFHMDF POS=(16,35),ATTRB=(PROT,NORM),LENGTH=29,
X

002201 INITIAL='(D - DELIVERED, W - WAITLIST)'
002202 DFHMDF POS=(19,15),ATTRB=(UNPROT,FSET),LENGTH=08,
X

002203 INITIAL='NUMBER :'
002204 NUM DFHMDF POS=(19,26),ATTRB=(UNPROT,FSET),LENGTH=2,
X

002205 INITIAL='__'
002206 DFHMDF POS=(19,30),ATTRB=(PROT,NORM),LENGTH=1,
X

002207 INITIAL=''
002208 DFHMDF POS=(22,08),ATTRB=(ASKIP,NORM),LENGTH=08,
X

002209 INITIAL='F1 - ADD'
002210 DFHMDF POS=(22,18),ATTRB=(ASKIP,NORM),LENGTH=47,
X

002211 INITIAL='F2 - UPDATE F3 - VIEW F4 - CLEAR F9 - EXIT'
002212 MSG DFHMDF POS=(20,10),ATTRB=(ASKIP,NORM),LENGTH=50
002220 DFHMDF TYPE=FINAL

002300 END

***** Bottom of Data *****

TRG017.PROJECT.PGMS(COMP)

000200 IDENTIFICATION DIVISION.
000300 PROGRAM-ID. COMPIN.
000400 ENVIRONMENT DIVISION.
000500 DATA DIVISION.
000600 WORKING-STORAGE SECTION.
000700 COPY DFHAID.
000800 COPY MAP5.
000900 01 REC.

Global Software Ltd.

```
001000      02 RCODE PIC X(4).
001100      02 RTYCOMP PIC X(35).
001200      02 RDTCOMP PIC X(10).

001300      02 RDTATT PIC X(10).
001400      02 RCHARGES PIC X(08).
002000      02 FILLER PIC X(12).
002100      77 REC-KEY PIC X(4).
002200      77 WS-RESP PIC S9(8) COMP.
002300      77 W-TEXT PIC X(27).
002500      77 WS-DT PIC S9(15) COMP-3.
002501      77 WS-TIME PIC X(8).
002510      77 WS-DATE PIC X(08).
002600      01 WS-COMMAREA.
002700          05 VARS1 PIC X(4).
002800      LINKAGE SECTION.
002900      01 DFHCOMMAREA.
003000          05 VARS2 PIC X(4).
003100      PROCEDURE DIVISION.
003200      MAIN-PROC.
003300          IF EIBCALEN = 0 THEN
003400              MOVE LOW-VALUES TO COMPHDO.
003500          IF EIBCALEN NOT = 0 THEN
003600              PERFORM RECEIVE-MAP.
003610              PERFORM TIME-PARA.
003700              PERFORM SEND-MAP.
003800              PERFORM RETURN-TRANS.
003900      SEND-MAP.
004000          EXEC CICS
004100              SEND MAPSET('MAP017') MAP('COMPHD') FROM(COMPHDO)
004200              ERASE
004300          END-EXEC.
004400      RECEIVE-MAP.
004500          EXEC CICS
004600              RECEIVE MAPSET('MAP017') MAP('COMPHD') INTO(COMPHDI)
004700          END-EXEC.
004800      EVALUATE EIBAID
004900          WHEN DFHPPF1 PERFORM ADD-PARA
005000          WHEN DFHPPF2 PERFORM MODIFY-PARA
005100          WHEN DFHPPF3 PERFORM VIEW-PARA
005200          WHEN DFHPPF4 PERFORM CLEAR-PARA
005300          WHEN DFHPPF9 PERFORM EXIT-PARA
005400          WHEN OTHER PERFORM ERROR-PARA
005500      END-EVALUATE.
005510      TIME-PARA.
005520          EXEC CICS ASKTIME ABSTIME(WS-DT) END-EXEC.
```

Global Software Ltd.

```
005530     EXEC CICS FORMATTIME ABSTIME(WS-DT)
005540         DDMMYY(WS-DATE) DATESEP("/")
005550         TIME(WS-TIME) TIMESEP(":")

005560     END-EXEC.
005570     MOVE WS-DATE TO DATEO.
005580     MOVE WS-TIME TO TIMEO.
005600     ERROR-PARA.
005700     MOVE "PLEASE PRESS VALID KEY" TO MSGO.
005800     CLEAR-PARA.
005900     MOVE LOW-VALUES TO COMPHDO.
006000     VIEW-PARA.
006100     MOVE CODEI TO REC-KEY.
006200     EXEC CICS READ FILE("TRG017F1")
006300         INTO(REC)
006400         RIDFLD(REC-KEY)
006500         RESP(WS-RESP)
006600     END-EXEC.
006700     IF WS-RESP = DFHRESP(NORMAL)
006800         MOVE RCODE TO CODEO
006900         MOVE RTYCOMP TO TYCOMPO
007000         MOVE RDTCOMP TO DTCOMPO
007100         MOVE RDTATT TO DTATTO
007200         MOVE RCHARGES TO CHARGESO
007800         MOVE "COMPLAINT RECORD READ SUCCESSFULLY !!!" TO
MSGO
007900     ELSE
008000         MOVE "NO COMPLAINT RECORDS FOUND !!!" TO MSGO.
008100     MODIFY-PARA.
008200     MOVE CODEI TO REC-KEY.
008300     EXEC CICS READ FILE("TRG017F1")
008400         INTO(REC)
008500         RIDFLD(REC-KEY)
008600         UPDATE
008700         RESP(WS-RESP)
008800     END-EXEC.
008900     IF WS-RESP = DFHRESP(NORMAL)
009000         MOVE CODEI TO RCODE
009100         MOVE TYCOMPI TO RTYCOMP
009200         MOVE DTCOMPI TO RDTCOMP
009300         MOVE DTATTI TO RDTATT
009400         MOVE CHARGESI TO RCHARGES
010000     EXEC CICS REWRITE FILE("TRG017F1")
010100         FROM(REC)
010200         RESP(WS-RESP)
010300     END-EXEC
```

```

Global Software Ltd.
010400 IF WS-RESP = DFHRESP(NORMAL)
010500 MOVE "RECORD UPDATED SUCCESSFULLY !!!" TO MSGO
ELSE
010700 MOVE "RECORD NOT UPDATED!" TO MSGO
ELSE
010800
010900 MOVE "NO COMPLAINTS FROM CUSTOMER !!!" TO MSGO
ADD-PARA.
011010 MOVE CODEI TO RCODE
011020 MOVE TYCOMP1 TO RTYCOMP
011030 MOVE DTCOMP1 TO RDTCOMP
011040 MOVE DTATT1 TO RDTATT
011050 MOVE CHARGESI TO RCHARGES
012200 MOVE CODEI TO REC-KEY
EXEC CICS WRITE
012300 FILE(TRG017F1)
FROM(REC)
RIDFLD(REC-KEY)
012600 LENGTH(LENGTH OF REC)
RESP(WS-RESP)
012800 END-EXEC.
013000 IF WS-RESP = DFHRESP(NORMAL)
013100 MOVE "COMPLAINT NOTED !!!" TO MSGO
013200 ELSE IF WS-RESP = DFHRESP(DUPREC)
013300 MOVE "COMPLAINT ALREADY NOTED !!!" TO MSGO
ELSE
013400
013500 MOVE "COMPLAINT NOT NOTED !! PLEASE CHECK !!!" TO
MSGO.
RETURN-TRANS.
013600 EXEC CICS
RETURN-TRANSID("GC17")
013800 COMAREA(WS-COMAREA)
013900 LENGTH(LENGTH OF WS-COMAREA)
END-EXEC.
EXIT-PARA.
014200 MOVE "THANK YOU !! TRY AGAIN !!!" TO W-TEXT
EXEC CICS
SEND TEXT FROM(W-TEXT) ERASE
014400 END-EXEC.
014500 EXEC CICS RETURN END-EXEC.
014700 STOP RUN.
014800
*****
***** Bottom of data *****

```

TRG017.PROJECT.PGMS(BILLING)

000200	IDENTIFICATION DIVISION.
000300	PROGRAM-ID. BILL.
000400	ENVIRONMENT DIVISION.
000500	DATA DIVISION.
000600	WORKING-STORAGE SECTION.
000700	COPY DFHAIID.
000800	COPY MAP9.
000900	01 REC.
001000	02 RBILLNO PIC X(6).
001100	02 RCODE PIC X(04).
001200	02 RQTY PIC 9(02).
001300	02 RAMT PIC 9(6).
002000	02 FILLER PIC X(61).
002010	01 REC1.
002020	02 RCODE1 PIC X(4).
002030	02 RBDAT PIC X(10).
002040	02 RQTY1 PIC 9(2).
002050	02 RSTATUS PIC A.
002051	02 RNO PIC 99.
002060	02 FILLER PIC X(63).
002100	77 REC-KEY PIC X(4).
002110	77 AMOUNT PIC 9(6).
002200	77 WS-RESP PIC S9(8) COMP.
002300	77 W-TEXT PIC X(27).
002600	77 WS-DT PIC S9(15) COMP-3.
002700	77 WS-DATE PIC X(08).
002800	77 WS-TIME PIC X(8).
002900	01 WS-COMMAREA.
003000	05 VARS1 PIC X(4).
003100	LINKAGE SECTION.
003200	01 DFHCOMMAREA.
003300	05 VARS2 PIC X(4).
003400	PROCEDURE DIVISION.
003500	MAIN-PROC.
003600	IF EIBCALEN = 0 THEN
003700	MOVE LOW-VALUES TO BILLHDO.
003800	IF EIBCALEN NOT = 0 THEN
003900	PERFORM RECEIVE-MAP.
004000	PERFORM TIME-PARA.
004100	PERFORM SEND-MAP.

PERFORM RETURN-TRANS.	004200
SEND-MAP.	004300
EXEC CICS	004400
SEND MAPSET(MAP017) FROM(BILLHD)	004500
ERASE	004600
END-EXEC.	004700
RECEIVE-MAP.	004800
EXEC CICS	004900
RECEIVE MAPSET(MAP017) MAP(BILLHD) INTO(BILLHD)	005000
END-EXEC.	005100
EVALUATE EIBRID	005200
WHEN DFHPF3 PERFORM VIEW-PARA	005500
WHEN DFHPF1 PERFORM CLEAR-PARA	005600
WHEN DFHPF9 PERFORM EXIT-PARA	005700
WHEN OTHER PERFORM ERROR-PARA	005800
END-EVALUATE.	005900
TIME-PARA.	006000
EXEC CICS ASKTIME ABSTIME(WS-DT) END-EXEC.	006100
EXEC CICS FORMATIME ABSTIME(WS-DT)	006200
DDMMYY(WS-DATE) DATESEP("/")	006300
TIME(WS-TIME) TIMESEP(":;")	006400
END-EXEC.	006500
MOVE WS-DATE TO DATEO.	006600
MOVE WS-TIME TO TIMEO.	006700
ERROR-PARA.	006800
MOVE "PLEASE PRESS VALID KEY" TO MSGO.	006900
CLEAR-PARA.	007000
MOVE LOW-VALUES TO BILLHDO.	007100
VIEW-PARA.	007200
MOVE CODEI TO REC-KEY.	007300
EXEC CICS READ FILE("TRG017F1")	007400
INTO(REC1)	007500
RIDFLD(REC-KEY)	007600
RESP(WS-RESP)	007700
END-EXEC.	007800
IF WS-RESP = DFHRESP(NORMAL)	007900
IF RSTATUS = 'D'	007910
MOVE '0' TO AMOUNT	007911
COMPUTE AMOUNT = RQTY1 * 000250	007920
MOVE AMOUNT TO RAMT	007930
MOVE RCODEI TO CODEO	007940
MOVE RQTY1 TO QTYO	008600
MOVE RAMT TO AMTO	008700
MOVE "BILL GENERATED SUCCESSFULLY !!!" TO MSGO	009000

009010 ELSE MOVE "CUSTOMER IN WAITING LIST !!!" TO MSGO
009100 ELSE

009200 MOVE "CUSTOMER CODE NOT VALID !!!" TO MSGO.
RETURN-TRANS.
EXEC CICS

015000 RETURN TRANSID("GC17")
COMAREA(WS-COMAREA)
LENGTH(LENGTH OF WS-COMAREA)

015300 END-EXEC.
EXIT-PARA.

015500 MOVE "THANK YOU !! TRY AGAIN !!!" TO W-TEXT
EXEC CICS

015700 SEND TEXT FROM(W-TEXT) ERASE
END-EXEC.

015900 EXEC CICS RETURN END-EXEC.

***** Bottom of Data *****

TRG017.PROJECT.PGMS(PUR1)

000100	IDENTIFICATION DIVISION.
000200	PROGRAM-ID. PURCHASE.
000300	ENVIRONMENT DIVISION.
000400	DATA DIVISION.
000500	WORKING-STORAGE SECTION.
000600	COPY DFHAIID.
000700	COPY MAP2.
000800	01 REC.
000900	02 RBILL-NO PIC X(6).
001000	02 RBILL-DT PIC X(10).
001100	02 RPCODE PIC 9(2).
001200	02 RQTY PIC 9(6).
001201	02 RSTP PIC X(5).
001210	02 FILLER PIC X(51).
001220	01 RECI.
001222	02 RCODE PIC 99.
001223	02 RDNO PIC X(6).
001230	02 ROB PIC 9(6).
001240	02 RRI PIC A.
001250	02 RDATE PIC X(10).
001260	02 RQTY1 PIC 9(6).
001270	02 FILLER PIC X(49).

Global Software Ltd.

```
001300    77 REC-KEY PIC X(6).
001400    77 WS-RESP PIC S9(8) COMP.

001500    77 W-TEXT PIC X(27).
001510    77 WS-RESP1 PIC S9(8) COMP.
001520    77 REC-KEY1 PIC X(6).
001530    77 REC-KEY2 PIC 99.
001540    77 WS-RESP2 PIC S9(8).
001550    77 UPD PIC 9(8).
001560    77 WS-DT PIC S9(15) COMP-3.
001570    77 WS-TIME PIC X(8).
001580    77 WS-DATE PIC X(08).
001600    01 COMMAREA.
001700        05 VARS1 PIC X(4).
001800    LINKAGE SECTION.
001900    01 DFHCOMMAREA.
002000        05 VARS PIC X(4).
002100    PROCEDURE DIVISION.
002200    MAIN-PROC.
002300        IF EIBCALEN = 0 THEN
002400            MOVE LOW-VALUES TO PURHEADO.
002500        IF EIBCALEN NOT = 0 THEN
002600            PERFORM RECEIVE-MAP.
002610            PERFORM TIME-PARA.
002700            PERFORM SEND-MAP.
002800            PERFORM RETURN-TRANS.
002900        SEND-MAP.
003000        EXEC CICS
003100            SEND MAPSET('MAP017') MAP('PURHEAD')
FROM(PURHEADO)
003200            ERASE
003300            END-EXEC.
003400        RECEIVE-MAP.
003500        EXEC CICS
003600            RECEIVE MAPSET('MAP017') MAP('PURHEAD')
INTO(PURHEAD1)
003700        END-EXEC.
003800        EVALUATE EIBAID
003900            WHEN DFHPPF1 PERFORM ADD-PARA
004000            WHEN DFHPPF2 PERFORM MODIFY-PARA
004100            WHEN DFHPPF3 PERFORM VIEW-PARA
004200            WHEN DFHPPF4 PERFORM CLEAR-PARA
004300            WHEN DFHPPF9 PERFORM EXIT-PARA
004400            WHEN OTHER PERFORM ERROR-PARA
004500        END-EVALUATE.
```

Global Software Ltd.

```
004510    TIME-PARA.
004520      EXEC CICS ASKTIME ABSTIME(WS-DT) END-EXEC.
004530      EXEC CICS FORMATTIME ABSTIME(WS-DT)
004540          DDMMYY(WS-DATE) DATESEP("/")
004550          TIME(WS-TIME) TIMESEP(":")
004560      END-EXEC.
004570      MOVE WS-DATE TO DATEO.
004580      MOVE WS-TIME TO TIMEO.
004600    ERROR-PARA.
004700      MOVE "PLEASE PRESS VALID KEY" TO MSGO.
004800    CLEAR-PARA.
004900      MOVE LOW-VALUES TO PURHEADO.
005000    VIEW-PARA.
005100      MOVE BILL-NOI TO REC-KEY.
005200      EXEC CICS READ FILE("TRG017F1")
005300          INTO(REC)
005400          RIDFLD(REC-KEY)
005500          RESP(WS-RESP)
005600      END-EXEC.
005700      IF WS-RESP = DFHRESP(NORMAL)
005800          MOVE RBILL-NO TO BILL-NOO
005900          MOVE RBILL-DT TO BILL-DTO
006000          MOVE RPCODE TO PCODEO
006100          MOVE RQTY TO QTYO
006110          MOVE RSUP TO SUPO
006200          MOVE "RECORD READ SUCCESSFULLY!!!" TO MSGO
006300      ELSE
006400          MOVE "RECORD NOT FOUND!" TO MSGO.
006500    MODIFY-PARA.
006600      MOVE BILL-NOI TO REC-KEY.
006700      EXEC CICS READ FILE("TRG017F1")
006800          INTO(REC)
006900          RIDFLD(REC-KEY)
007000          UPDATE
007100          RESP(WS-RESP)
007200      END-EXEC.
007300      IF WS-RESP = DFHRESP(NORMAL)
007400          MOVE BILL-NOI TO RBILL-NO
007500          MOVE BILL-DTI TO RBILL-DT
007600          MOVE PCODEI TO RPCODE
007700          MOVE QTYI TO RQTY
007710          MOVE SUPI TO RSUP
007800          EXEC CICS
007900              REWRITE FILE("TRG017F1")
008000                  FROM(REC)
008100                  RESP(WS-RESP)
```

Global Software Ltd.

```
008200     END-EXEC
008300     IF WS-RESP = DFHRESP(NORMAL)
008400         MOVE "RECORD UPDATED SUCCESSFULLY!!!" TO MSGO
008500     ELSE
008600         MOVE "RECORD NOT UPDATED!" TO MSGO
008700     ELSE
008800         MOVE "RECORD NOT FOUND" TO MSGO.
008900     ADD-PARA.
009000     MOVE SPACES TO REC.
009100     MOVE BILL-NOI TO RBILL-NO
009200     MOVE BILL-DTI TO RBILL-DT
009300     MOVE PCODEI TO RPCODE
009400     MOVE QTYI TO RQTY
009410     MOVE SUPI TO RSUP
009500     MOVE BILL-NOI TO REC-KEY
009510     MOVE PCODEI TO REC-KEY2
009600     IF RPCODE = 11 OR RPCODE = 12
009700         IF RQTY > 0 THEN
009800             EXEC CICS WRITE
009900                 FILE("TRG017F1")
010000                 FROM(REC)
010100                 RIDFLD(REC-KEY)
010200                 LENGTH(LENGTH OF REC)
010300                 RESP(WS-RESP)
010400             END-EXEC
010402             EXEC CICS READ FILE("TRG017F2")
010403                 INTO(REC1)
010404                 RIDFLD(REC-KEY2)
010405                 UPDATE
010406                 RESP(WS-RESP1)
010407             END-EXEC
010409             IF WS-RESP1 = DFHRESP(NORMAL)
010410                 COMPUTE UPD = ROB + RQTY1
010411                 MOVE PCODEI TO REC-KEY
010412                 MOVE SPACES TO REC1
010413                 MOVE PCODEI TO RCODE
010414                 MOVE BILL-NOI TO RDNO
010415                 MOVE SUPI TO RSUP
010421                 MOVE UPD TO ROB
010430                 MOVE 'R' TO RRI
010440                 MOVE BILL-DTI TO RDATE
010450                 MOVE QTYI TO RQTY1
010460                 EXEC CICS REWRITE FILE("TRG017F2")
010470                     FROM(REC1)
010491                     RESP(WS-RESP2)
010492                 END-EXEC
```

Global Software Ltd.

```
010500      IF WS-RESP = DFHRESP(NORMAL)
010600          MOVE "RECORD ADDED SUCCESSFULLY!!!" TO MSGO
010700      ELSE IF WS-RESP = DFHRESP(DUPREC)
010800          MOVE "DUPLICATE RECORD -RECORD NOT ADDED!!!" TO
MSGO
010900      ELSE
011000          MOVE "RECORD NOT ADDED !! PLEASE CHECK!" TO
MSGO
011100      IF WS-RESP2 = DFHRESP(NORMAL)
011020          MOVE "STOCK UPDATED" TO MSGO
011100      ELSE MOVE "QUANTITY SHOULD BE GREATER THAN '0'" TO
MSGO
011200      ELSE MOVE "PRODUCT CODE NOT VALID !! " TO MSGO.
011300      RETURN-TRANS.
011400      EXEC CICS
011500          RETURN TRANSID("GC17")
011600          COMMAREA(COMMAREA)
011700          LENGTH(LENGTH OF COMMAREA)
011800      END-EXEC.
011900      EXIT-PARA.
012000      MOVE "THANK YOU !! TRY AGAIN !!" TO W-TEXT
012100      EXEC CICS
012200          SEND TEXT FROM(W-TEXT) ERASE
012300      END-EXEC.
012400      EXEC CICS RETURN END-EXEC.
012500      STOP RUN.
```

***** ***** Bottom of Data*****

TRG017.PROJECT.PGMS(ENQUIRY)

```
000100      IDENTIFICATION DIVISION.
000200      PROGRAM-ID. BOOKING.
000300      ENVIRONMENT DIVISION.
000400      DATA DIVISION.
000500      WORKING-STORAGE SECTION.
000600      COPY DFHAID.
000700      COPY MAP4.
000800      01 REC.
000900          02 RREGST PIC 9(6).
001000          02 RCYLST PIC 9(6).
001100          02 RNWL PIC 9(2).
001110          02 RNO PIC 99.
```

Global Software Ltd.

001200 02 FILLER PIC X(66).
001300 01 REC1.
001400 02 RCODE PIC 99.

001500 02 RDNO PIC X(6).
001600 02 ROB PIC 9(6).
001700 02 RRI PIC A.
001800 02 RDATE PIC X(10).
001900 02 RQTY1 PIC 9(6).
002000 02 FILLER PIC X(49).
002100 01 REC2.
002200 02 RCODE1 PIC X(4).
002300 02 RDATE1 PIC X(8).
002400 02 RQTY PIC 99.
002500 02 RSTATUS PIC X.
002600 02 FILLER PIC X(65).
002700 77 WS-DT PIC S9(15) COMP-3.
002800 77 WS-TIME PIC X(8).
002900 77 WS-DATE PIC X(08).
003000 77 REC-KEY PIC 9(2).
003100 77 REC-KEY1 PIC 9999.
003200 77 WS-RESP PIC S9(8) COMP.
003300 77 W-TEXT PIC X(27).
003400 77 W-TEXT1 PIC X(27).
003500 77 NUM PIC 9(2).
003600 LINKAGE SECTION.
003700 01 DFHCOMMAREA.
003800 05 VARS PIC X(4).
003900 PROCEDURE DIVISION.
004000 MAIN-PROC.
004100 MOVE LOW-VALUES TO INHEADO.
004200 MAIN-PARA.
004300 PERFORM TIME-PARA.
004400 PERFORM SEND-MAP.
004500 PERFORM RECEIVE-MAP.
004600 GO MAIN-PARA.
004700 SEND-MAP.
004800 EXEC CICS
004900 SEND MAPSET('MAP017') MAP('INHEAD') FROM(INHEADO)
005000 ERASE
005100 END-EXEC.
005200 RECEIVE-MAP.
005300 EXEC CICS
005400 RECEIVE MAPSET('MAP017') MAP('INHEAD') INTO(INHEADI)
005500 END-EXEC.
005600 EVALUATE EIBAID

Global Software Ltd.

005610 WHEN DFHPF1 PERFORM NO-PARA
005700 WHEN DFHPF3 PERFORM VIEW-PARA
005710 WHEN DFHPF2 PERFORM BR-PARA

005800 WHEN DFHPF4 PERFORM CLEAR-PARA
005900 WHEN DFHPF9 PERFORM EXIT-PARA
006000 WHEN OTHER PERFORM ERROR-PARA
006100 END-EVALUATE.
006200 TIME-PARA.
006300 EXEC CICS ASKTIME ABSTIME(WS-DT) END-EXEC.
006400 EXEC CICS FORMATTIME ABSTIME(WS-DT)
006500 DDMYY(WS-DATE) DATESEP("/")
006600 TIME(WS-TIME) TIMESEP(":")
006700 END-EXEC.
006800 MOVE WS-DATE TO DATEO.
006900 MOVE WS-TIME TO TIMEO.
007000 ERROR-PARA.
007100 MOVE "PRESS VALID KEY !!!" TO MSGO.
007200 CLEAR-PARA.
007300 MOVE LOW-VALUES TO INHEADO.
007400 VIEW-PARA.
007500 MOVE '0' TO NUM.
007600 MOVE '11' TO REC-KEY.
007700 EXEC CICS READ FILE("TRG017F1")
007800 INTO(REC1)
007900 RIDFLD(REC-KEY)
008000 RESP(WS-RESP)
008100 END-EXEC.
008200 IF WS-RESP = DFHRESP(NORMAL)
008300 MOVE '0' TO RCYLST
008400 ADD ROB, RQTY1 TO RCYLST
008500 MOVE RCYLST TO CYL-STKO
008600 MOVE '0' TO REG-STKO
008700 MOVE '0' TO WLO.
008800 MOVE '12' TO REC-KEY.
008900 EXEC CICS READ FILE("TRG017F1")
009000 INTO(REC1)
009100 RIDFLD(REC-KEY)
009200 RESP(WS-RESP)
009300 END-EXEC.
009400 IF WS-RESP = DFHRESP(NORMAL)
009500 MOVE '0' TO RREGST
009600 ADD ROB, RQTY1 TO RREGST
009700 MOVE RCYLST TO CYL-STKO
009800 MOVE RREGST TO REG-STKO
009900 MOVE '0' TO WLO.

Global Software Ltd.

```
010000      MOVE ZEROS TO REC-KEY1.
010100      MOVE '1111' TO REC-KEY1.
010200      EXEC CICS STARTBR FILE("TRG017F2")

010300          RIDFLD(REC-KEY1)
010400          GTEQ
010500      END-EXEC.
010600      EXEC CICS
010700          READNEXT
010800          FILE("TRG017F2")
010900          INTO(REC2)
011000          RIDFLD(REC-KEY1)
011100          LENGTH(LENGTH OF REC2)
011200          RESP(WS-RESP)
011300      END-EXEC.
011400      IF WS-RESP = DFHRESP(NORMAL)
011500          IF RSTATUS = 'W'
011600              COMPUTE NUM = NUM + 1
011700              EXEC CICS
011800                  SEND TEXT FROM(NUM)
011900              END-EXEC
012000          END-IF
012010          MOVE NUM TO WLO
012100      END-IF.
012140      BR-PARA.
012141          COMPUTE REC-KEY = REC-KEY + 1.
012143          IF WS-RESP NOT = DFHRESP(ENDFILE)
012160              EXEC CICS
012170                  SEND TEXT FROM(WS-RESP)
012180              END-EXEC
012300          MOVE 'INSIDE BR PARA' TO W-TEXT
012400          EXEC CICS
012500              SEND TEXT FROM(W-TEXT)
012600          END-EXEC
012700          EXEC CICS
012800              READNEXT
012900              FILE("TRG017F2")
013000              INTO(REC2)
013100              RIDFLD(REC-KEY1)
013200              LENGTH(LENGTH OF REC2)
013300              RESP(WS-RESP)
013400          END-EXEC
013500          IF WS-RESP = DFHRESP(NORMAL)
013600              IF RSTATUS = 'W'
013700                  COMPUTE NUM = NUM + 1
013800              END-IF
```

Global Software Ltd.

013900 EXEC CICS
014000 SEND TEXT FROM(NUM)
014100 END-EXEC

014200 MOVE NUM TO WLO
014300 END-IF
014400 ELSE MOVE "END-OF-FILE" TO MSGO.
014410 NO-PARA.
014420 MOVE "NUMBER" TO MSGO.
014500 EXIT-PARA.
014600 MOVE "THANK YOU !! TRY AGAIN !!" TO W-TEXT
014700 EXEC CICS
014800 SEND TEXT FROM(W-TEXT) ERASE
014900 END-EXEC.
015000 EXEC CICS RETURN END-EXEC.
015100 STOP RUN.

***** Bottom of Data*****

TRG017.PROJECT.PGMS(PURMAP)

000200 MAP017 DFHMSD TYPE=MAP,MODE=INOUT,LANG=COBOL,
X
000300 STORAGE=AUTO,TIOAPFX=YES
000400 PURHEAD DFHMDI SIZE=(24,80),CTRL=(PRINT,FREEKB)
000410 DFHMDF POS=(04,30),ATTRB=(ASKIP,NORM),LENGTH=16,
X
000420 INITIAL='PURCHASE DETAILS'
000430 DFHMDF POS=(05,29),ATTRB=(ASKIP,NORM),LENGTH=18,
X
000440 INITIAL='-----'
000450 DATE DFHMDF POS=(02,65),ATTRB=(UNPROT,NORM),LENGTH=10
000460 DFHMDF POS=(02,77),ATTRB=(PROT,NORM),LENGTH=1,
X
000470 INITIAL=' '
000480 TIME DFHMDF POS=(03,65),ATTRB=(UNPROT,NORM),LENGTH=8
000490 DFHMDF POS=(03,75),ATTRB=(PROT,NORM),LENGTH=1,
X
000491 INITIAL=' '
000500 DFHMDF POS=(07,15),ATTRB=(ASKIP,NORM),LENGTH=09,
X
000600 INITIAL='BILL NO :'

Global Software Ltd.

000700 BILL-NO DFHMDF

POS=(07,26),ATTRB=(UNPROT,FSET,IC),LENGTH=06, X

000800 INITIAL=' _____ '

000900 DFHMDF POS=(07,35),ATTRB=(PROT,NORM),LENGTH=1,
X

001000 INITIAL=' '

001001 DFHMDF POS=(10,15),ATTRB=(ASKIP,NORM),LENGTH=11,
X

001002 INITIAL='BILL DATE :'

001010 BILL-DT DFHMDF POS=(10,29),ATTRB=(UNPROT,FSET),LENGTH=10,
X

001020 INITIAL=' _____ '

001030 DFHMDF POS=(10,41),ATTRB=(PROT,NORM),LENGTH=1,
X

001040 INITIAL=' '

001041 DFHMDF POS=(13,15),ATTRB=(ASKIP,NORM),LENGTH=14,
X

001042 INITIAL='PRODUCT CODE :'

001050 PCODE DFHMDF POS=(13,32),ATTRB=(UNPROT,FSET),LENGTH=02,
X

001060 INITIAL=' _ ' '

001070 DFHMDF POS=(13,35),ATTRB=(PROT,NORM),LENGTH=1,
X

001080 INITIAL=' '

001081 DFHMDF POS=(16,15),ATTRB=(ASKIP,NORM),LENGTH=10,
X

001082 INITIAL='QUANTITY :'

001090 QTY DFHMDF POS=(16,28),ATTRB=(UNPROT,FSET),LENGTH=05,
X

001091 INITIAL=' _____ '

001092 DFHMDF POS=(16,35),ATTRB=(PROT,NORM),LENGTH=1,
X

001093 INITIAL=' '

001094 DFHMDF POS=(19,15),ATTRB=(ASKIP,NORM),LENGTH=10,
X

001095 INITIAL='SUPPLIER :'

001096 SUP DFHMDF POS=(19,28),ATTRB=(UNPROT,FSET),LENGTH=05,
X

001097 INITIAL=' _____ '

001098 DFHMDF POS=(19,35),ATTRB=(PROT,NORM),LENGTH=1,
X

001099 INITIAL=' '

001100 DFHMDF POS=(21,08),ATTRB=(ASKIP,NORM),LENGTH=8,
X

Global Software Ltd.

001101 INITIAL='F1 - ADD'

001102 DFHMDF POS=(21,20),ATTRB=(ASKIP,NORM),LENGTH=47,
X

001103 INITIAL='F2 - UPDATE F3 - VIEW F4 - CLEAR F9 - EXIT'

001104 MSG DFHMDF POS=(23,10),ATTRB=(PROT),LENGTH=40

001110 DFHMSD TYPE=FINAL

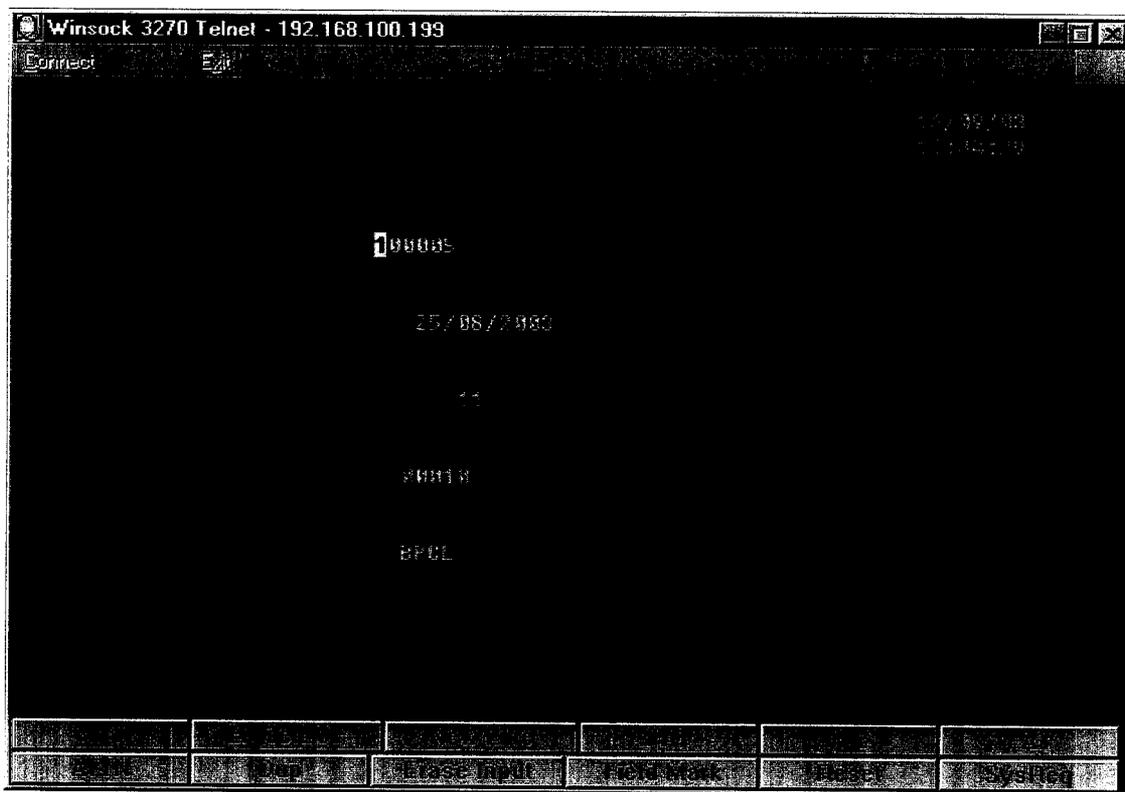
001200 END

***** Bottom of Data *****

Global Software Ltd.

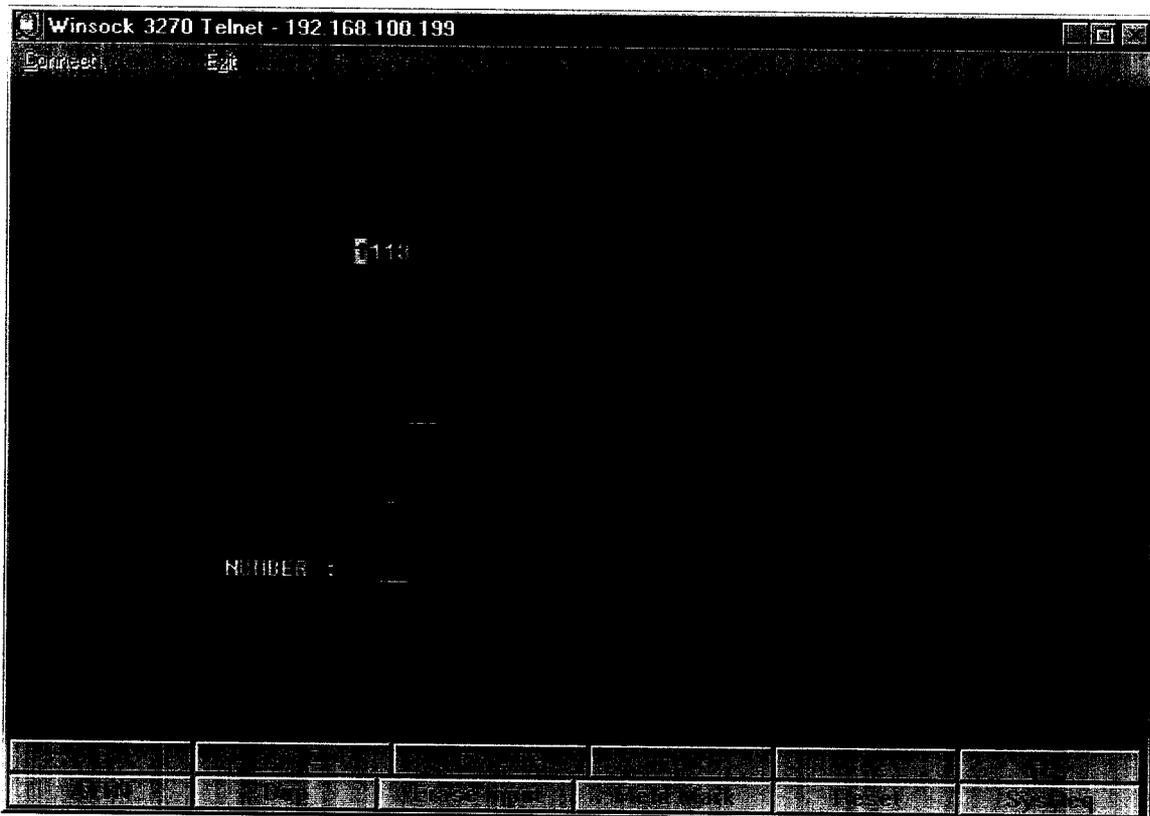
Sample Forms

Updation of Purchase Record



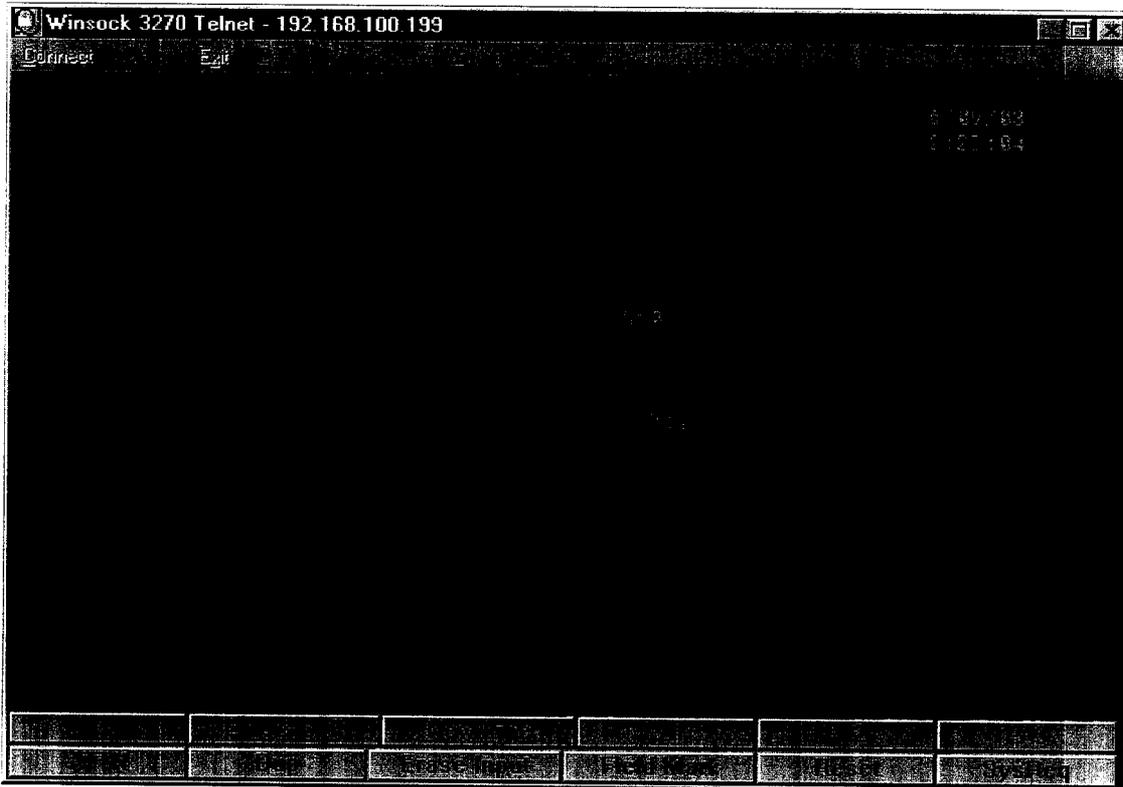
Global Software Ltd.

Booking done for customer already in Waiting List



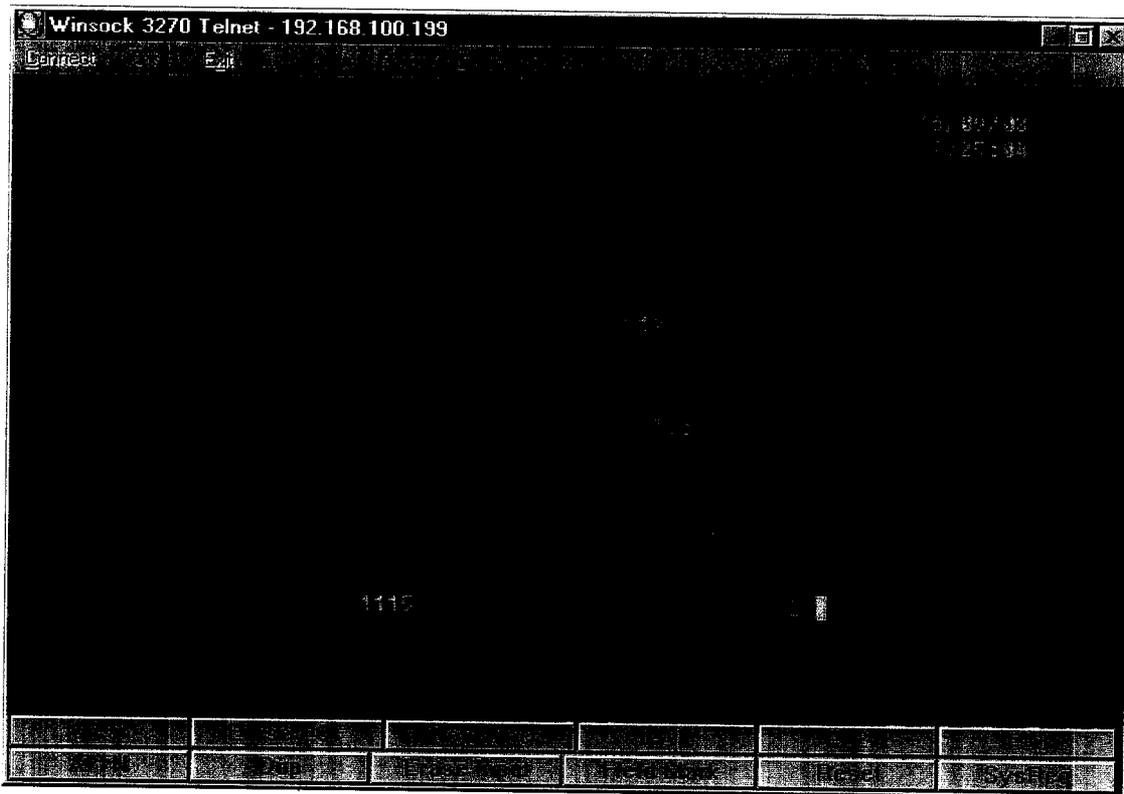
Global Software Ltd.

Stock Details



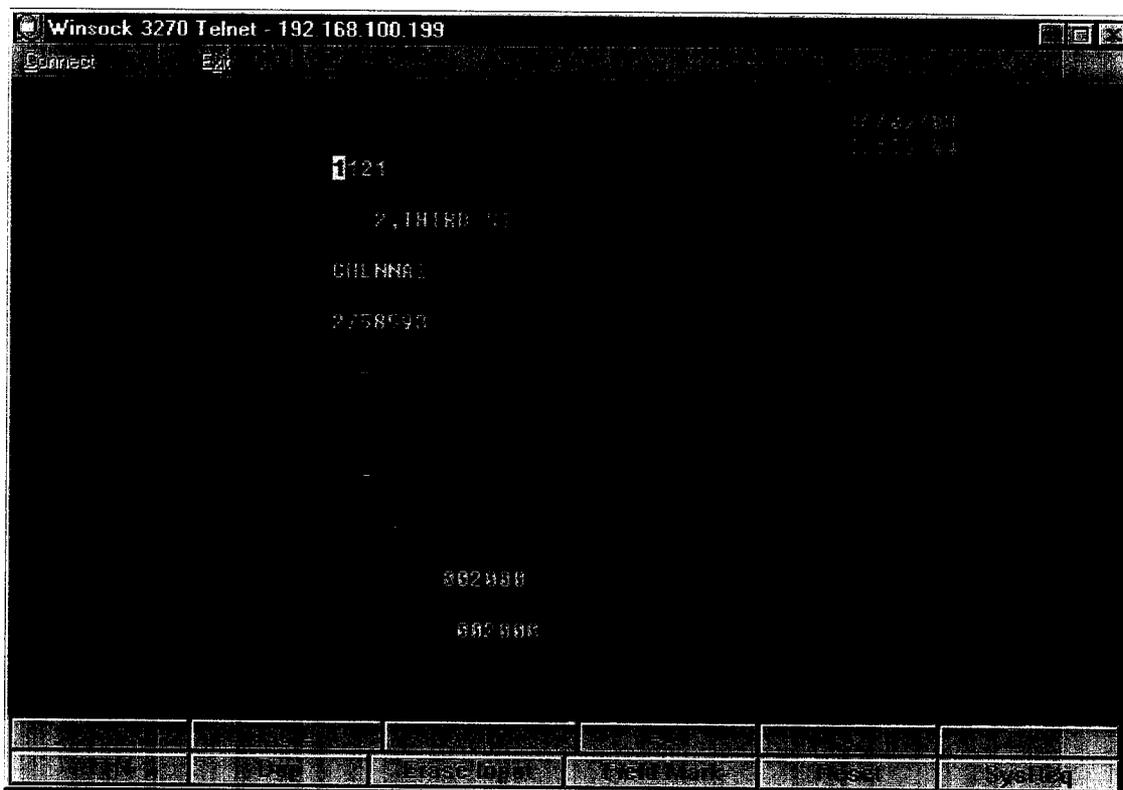
Global Software Ltd.

Displaying the Waitlist Number for a customer and the stock details



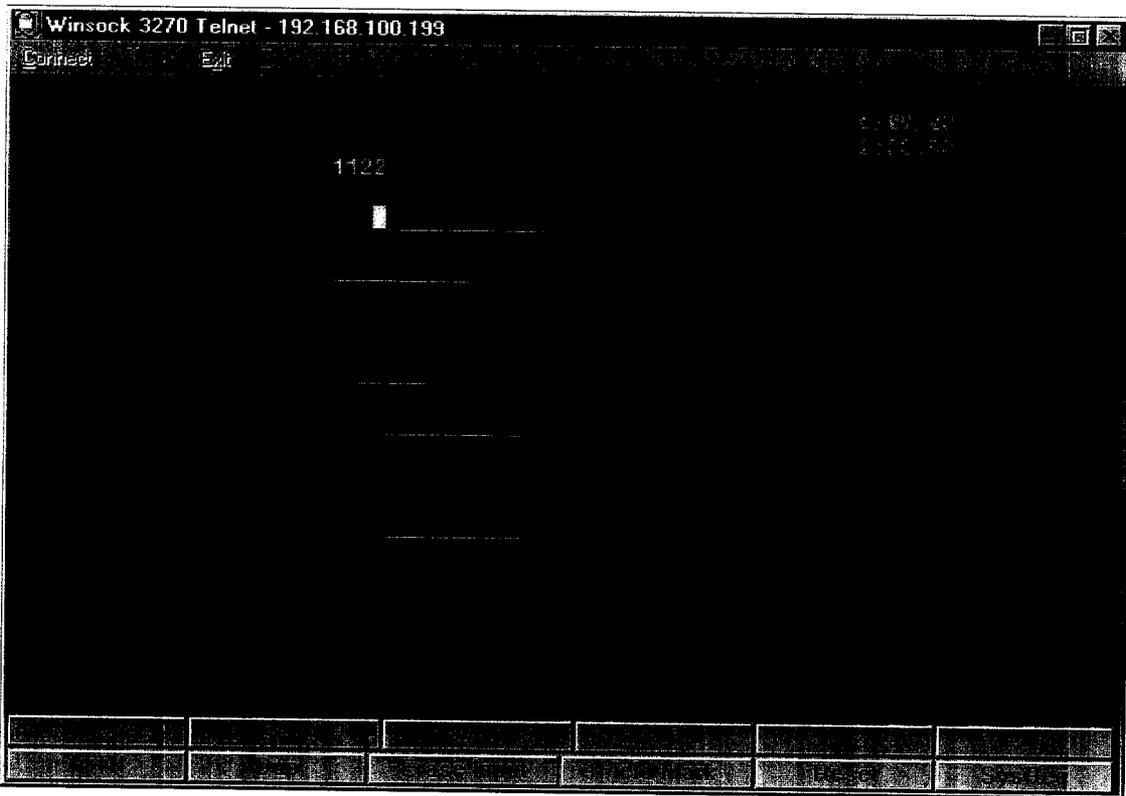
Global Software Ltd.

Adding a New Customer



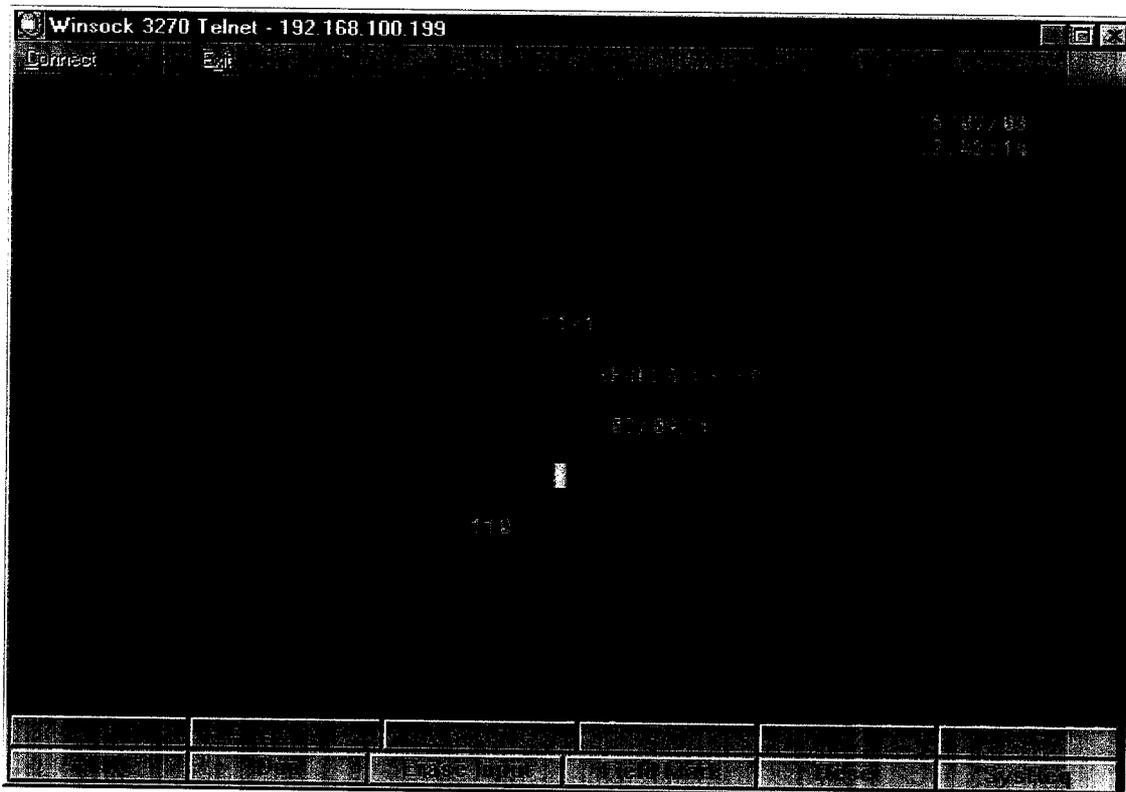
Global Software Ltd.

Automatic Generation of customer code



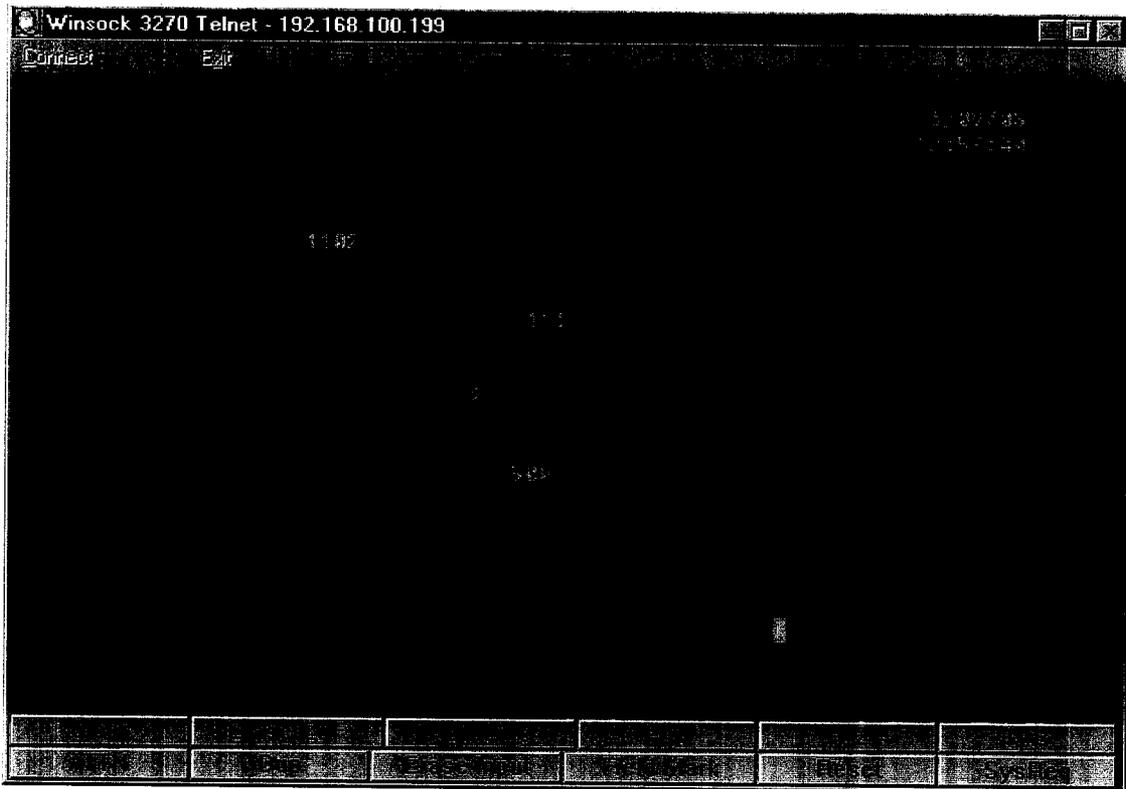
Global Software Ltd.

Complaints noting



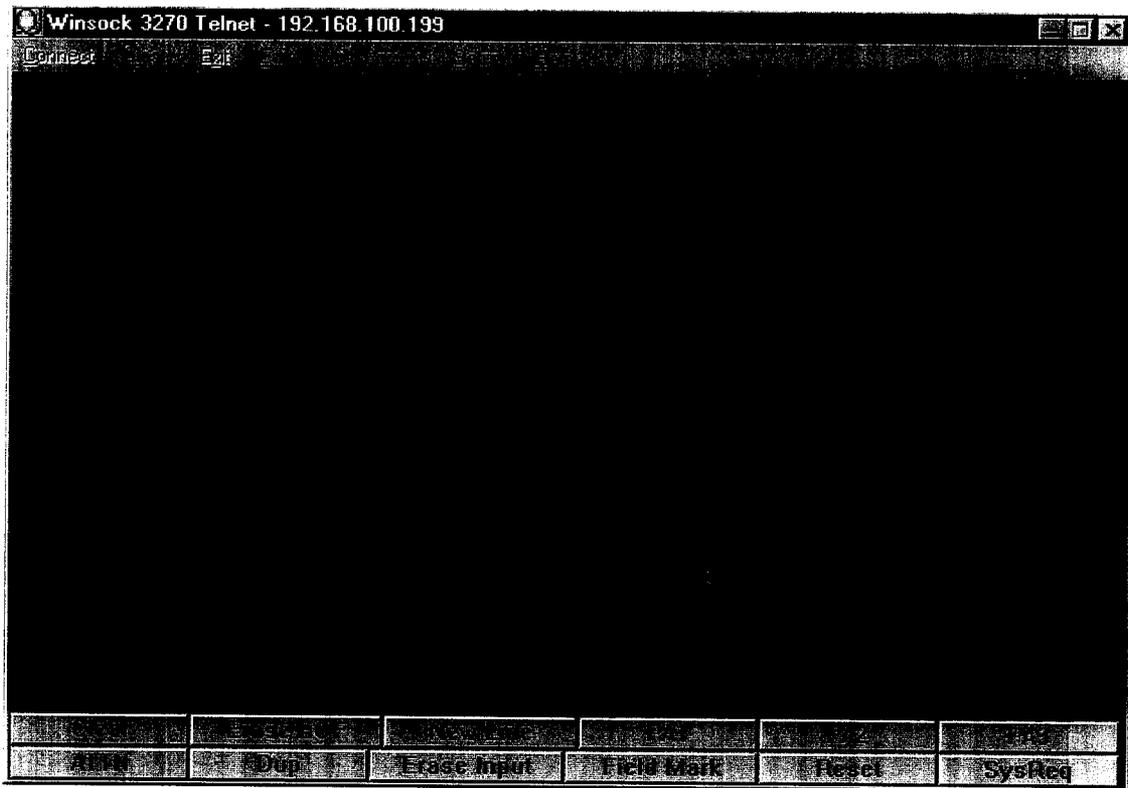
Global Software Ltd.

Bill Generation



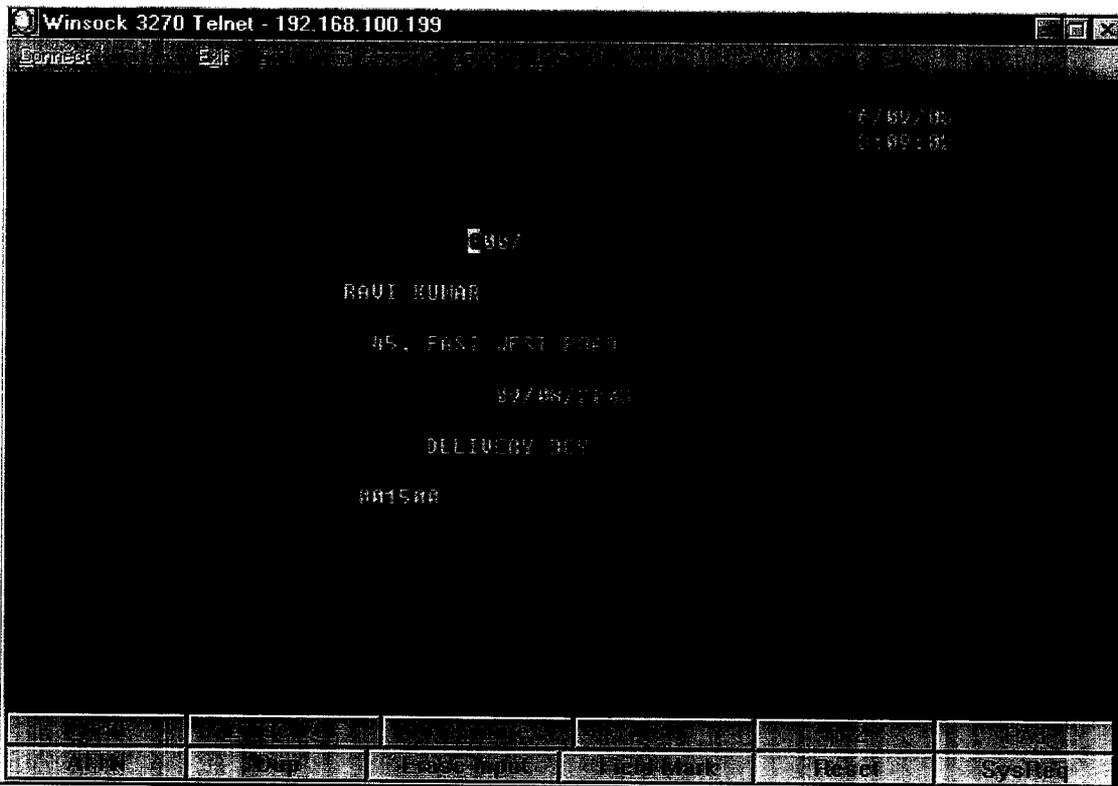
Global Software Ltd.

Main Screen



Global Software Ltd.

Reading a employee record



Global Software Ltd.

References