

P-3231



e-PRODUCTION DATAFLOW MANAGEMENT SYSTEM

PROJECT REPORT

Submitted By

M.JAYAPRAKASH

Register No.: 0720300014

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

MASTER OF COMPUTER APPLICATIONS

in

COMPUTER APPLICATIONS

KUMARAGURU COLLEGE OF TECHNOLOGY

(An Autonomous Institution Affiliated to Anna University, Coimbatore)

May 2010

ii

KUMARAGURU COLLEGE OF TECHNOLOGY

(An Autonomous Institution Affiliated to Anna University, Coimbatore)

COIMBATORE – 641 006.

Department of Computer Applications

PROJECT WORK

MAY 2010

This is to certify that the project entitled
e-PRODUCTION DATAFLOW MANAGEMENT SYSTEM
is the bonafide record of project work done by

M.JAYAPRAKASH

Register No: 0720300014

of MCA (Computer Applications) during the year 2009-2010.

Chakraborty 21/5/10
Project Guide

[Signature]
Head of the Department

Submitted for the Project Viva-Voce examination held on 17/05/10

Chakraborty 21/5/10

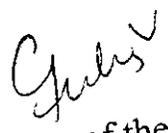
[Signature]
External Examiner

DECLARATION

I affirm that the project work titled **e-PRODUCTION DATAFLOW MANAGEMENT SYSTEM** being submitted in partial fulfilment for the award of **MASTER OF COMPUTER APPLICATIONS** is the original work carried out by me. It has not formed the part of any other project work submitted for award of any degree or diploma, either in this or any other University.


(Signature of the Candidate)
M.JAYAPRAKASH
Name of the Candidate
0720300
Register Number

I certify that the declaration made above by the candidate is true


Signature of the C

V.C.
Assistant Professor

Coimbatore
05th May 2010

TO WHOM SO EVER IT MAY CONCERN

This is to certify that Mr. Jayaprakash has successfully completed his project work in our company. During the project period he has successfully completed a web based project "Electronic process dataflow management system".

He was a hard working and self motivated person with the willingness and ability to learn new technologies and concepts.

We wish him all the very best in all his future endeavours.

Yours truly,

For SPEEDSTEP SOFTWARE SOLUTIONS (INDIA) PVT. LTD.


D. Surendran Babu
Manager (Operation)

ABSTRACT

The e-PDM (Production Dataflow Management) is application software aimed at developing a web application package that assists entities like Buyers, Agents, Merchandisers and Suppliers maintain and track production data work flow. It integrates all the processes that take place from order by a customer to delivery to a customer.

The application repository holds information pertaining to every Buyer, Agent, Merchandiser and Supplier who becomes part of the application once they are registered. The registration process is simple. Registration contains company and a single user registration who may act as an administrator. Once the company and single user registration is over, the user can login to create the initial setup or ground work for further processing.

The initial setup includes Branch registration, Department registration, and Users' registration. The users are categorized into the following categories: Buyer, Supplier, Agent and Merchandiser. This happens through a session.

Once the ground work is laid, the user with the right role and permission login to setup the process. The process setup includes creating a process with unique type and process description. Process steps are atomic steps a Process type has. These atomic steps have to be completed for completion of an order.

The process setup is followed by order creation. The order creation involves filling of order form that contains certain order data as agreed by the Supplier and Buyer. A Service Level Agreement (SLA) is signed by the Buyer and Supplier after registering with the e-PDM application software.

The order that is created has to be connected with the process type and the process type are confirmed further.

v

The order follow up includes various searching options based on the attributes available in the database. Each order includes process steps ranging from 30 to 50 tasks which have to be carried out for the order completion. Process steps belated are presented in red and scheduled today are posed in yellow by the e-PDM application.

The system provides provision for closing, deleting, recalculating dates, editing, adding an order process step. The system also includes the garment details as a part of the order. This order processing has certain level of abstraction where in editing, deleting, and viewing is done based on the user roles and permissions. Not every user has the right to create or edit an order and its corresponding process setup.

This application helps flexibility in communication between Buyers and the Suppliers who may be located in different locations on the globe.

The front end operations for the application are implemented using C#.NET and back end with SQL Server 2005.

ACKNOWLEDGEMENT

I wish to express sincerest thanks to **Dr. J. Shanmugam**, Director, Kumaraguru College of Technology, **Dr. S. Ramachandran**, Principal, Kumaraguru College of Technology, and **Dr. S. Thangasamy**, Dean, Department of Computer Science & Engineering, Kumaraguru College of Technology for providing necessary facilities in carrying out my project work.

I am very glad to express a special word of thanks to **Dr.A.Muthukumar**, Professor and Course coordinator, Department of Computer Applications, Kumaraguru College of Technology, for encouraging me to do this work.

I express my deep felt gratitude to **Mrs.V.Geetha**, Assistant Professor, Department of Computer Applications, Kumaraguru College of Technology, for her guidance, support, cooperation and valuable suggestions during the course of this project.

I wish to express my sincere thanks to **Mr.D. Surendran Babu**, Executive, **Speed Step Software Solutions** and Project Manager **Mr. John Prabu Harris** providing me this wonderful opportunity and their constant support throughout project.

Also I would like to thank my parents, friends, and all those helped me in this project and whose names are leftover.

CONTENTS

CHAPTER	Page No.
Abstract	iv
List of figures	ix
List of Tables	x
1. INTRODUCTION	1
1.1 Project Overview	1
2. SYSTEM ANALYSIS	4
2.1 Existing System	4
2.2 Proposed System	4
3. DEVELOPMENT ENVIRONMENT	7
3.1 Hardware Specification	7
3.2 Software Specification	7
3.3 Programming Environment	7
4. SYSTEM DESIGN	13
4.1 Data Model	13
4.1.1 Table Structure	13
4.2 Data Flow Diagram	22
4.3 Input and Ouput Design	27
5. SYSTEM IMPLEMENTATION	28
5.1 Implementation Environment	28
6. TESTING	30
6.1 Verification	30
6.1.1 Verification Phases	30
6.2 Validation	30
6.2.2 Validation Phases	30
7. CONCLUSION	30
7.1 Merits of Application	30
7.2 Limitations of Application	30
7.3 Future Enhancements	30

APPENDIX
REFERENCES

LIST OF FIGURES

S.NO	Figure No.	Title	Page No.
1	2.1	Architecture Diagram	6
2	3.1	ADO.NET Architecture	10
3	4.2.1	Level-0 DFD for e-PDM	22
4	4.2.2	Level-1 DFD for e-PDM	23
5	4.2.3	Level-2 DFD for e-PDM	24
6	4.2.4	Level-2 DFD for e-PDM	25
7	4.2.5	Level-2 DFD for e-PDM	26

LIST OF TABLES

S.NO	Table No.	Title	Page No
1	4.1.1.1	Company Table	13
2	4.1.1.2	Department Table	14
3	4.1.1.3	User Table	15
4	4.1.1.4	Process Type Table	16
5	4.1.1.5	Process steps Table	16
6	4.1.1.6	Orders Table	17
7	4.1.1.7	Order Process Table	19
8	4.1.1.8	Function Table	20
9	4.1.1.9	Function Details	21

CHAPTER 1

INTRODUCTION

This chapter gives an overall introduction about the e-Production Dataflow Management (e-PDM) application.

1.1 PROJECT OVERVIEW

e-Production Dataflow Management (e-PDM) application software is aimed at tailoring the needs of the garment industry. This application can be used to create orders and track everyday processes till the product is delivered to the buyer. All atomic tasks related to the order are explicitly defined in the system and anybody with right authentication can edit, view or delete data related to order.

e-PDM consists of the following modules:

- 1) Registration
- 2) Process Setup
- 3) Order Creation
- 4) Order Processing
- 5) Order Follow up

Registration module involves company registration and user registration. Company registration seeks information about company details like name, address, phone number, website or e-mail id and description of the company. Registration process also includes user registration that provides login name and password for setup and further modules.

Process setup module consists of Branch, Department, and Users' registration for the registering company. This seeks information about Branch details, Department details, and information about users who are part of the registering company. This module also includes creation of process types which further includes number of process steps for each process type. These process steps are atomic steps a process type contains to complete the order. A process type includes Process Type ID and Process Description. Every process step contains information like step number, step description, planned days, buffer days, responsible type (Agent, Buyer, Supplier, Merchandiser), name of the person responsible, base date(Process start date, Order date, ETD(Expected Time of Departure), ETA(Expected Time of Arrival), Buyer Delivery Date.

Order creation module takes in Order information. Order Information includes order number, style number, article number, collection type, production group, age group, buyer, supplier, and agent names, order quantity, shipped quantity, order date, process start date, ETD, ETA, buyer delivery and mode of shipment.

Order processing module connects the order to the process type. Confirm process binds the order to the unique process type. The system on connection of order and process type generates calculated planned date, calculated buffer date, follow up planned date, follow up buffer date, and remainder date. This generation is based on the allocated planned days and buffer days for every step in the process type. Each step is dependent either on base date (Process start date, Order date, ETD (Expected Time of Departure), ETA (Expected Time of Arrival), Buyer Delivery Date or its an of previous step number.

Closed date for a step is calculated when a step's task is done. This generates closed date for the step. When the calculated planned date of the step exceeds current date it is earmarked in red, steps on due are shown in yellow and closed step stroked out.

Order follow up includes searching based on various criterions. Searching can be based upon process start date, order date, ETD, ETA, buyer delivery, name of buyer, supplier, and agent, tasks to be completed today, order number ,style number and article number.

CHAPTER 2

SYSTEM ANALYSIS

System analysis is the examination and evaluation of an operation or task in order to identify and implement more efficient methods. System analysis involves the process of diagnosing, interpreting and helps us propose a new system.

2.1 EXISTING SYSTEM

Merchandiser is a person who interacts with the buyer and supplier, and also puts efforts to maintain consistent workflow from order creation to goods dispatch to the buyer. This involves fulfilling a series of tasks to complete the order. In the existing system these tasks are not transparent. The merchandiser on completing the following days tasks have to report to the supplier and buyer who is in a remote location across the globe. This reporting involves preparation of a task list of today's work in an Excel file and sending it to the supplier and buyer through e-mail. This clearly shows lack of communication and transparency in tracking orders.

Either the buyer or the supplier will be executing hundreds of orders at one time. Garment industry is based on a 60-90 day cycle so tracking of order and feedback and necessary communication among agents, buyers, suppliers and merchandisers is indispensable. Keeping a history of ongoing and completed orders is a must. Order tracking is inconvenient and difficult to pursue in the existing system.

2.2 PROPOSED SYSTEM

The proposed system overcomes the drawbacks faced in the existing system. Computerize the processes involved in the existing system. Automate the processes in the system. The proposed system will be able to track the orders in the system. The proposed system will be able to track the orders in the system.

system is provided with a user friendly GUI and provides maximum security. The data involved can be stored and retrieved using a computer terminal in any part of the world with necessary user rights.

The task list for the particular day is automatically generated by the system and assigns responsibility for the concerned task. Buyers who are located in remote locations can login and check the progress and intimate about any changes in order or scheduling tasks. The buyer can also query and get details about delay in tasks if any.

At any point in time records can be searched easily and accessed from anywhere. Order tracking is made easy and history of ongoing and completed orders can be maintained. User rights functionality provides users with necessary functionality in accessing the system. User permissions include read only, edit and update and delete functions.

Since this is a web application it can be accessed from anywhere with an internet connection. This enhances transparency and communication among entities. This application development is feasible and beneficial for the development of the organization.

What is Architecture Diagram?

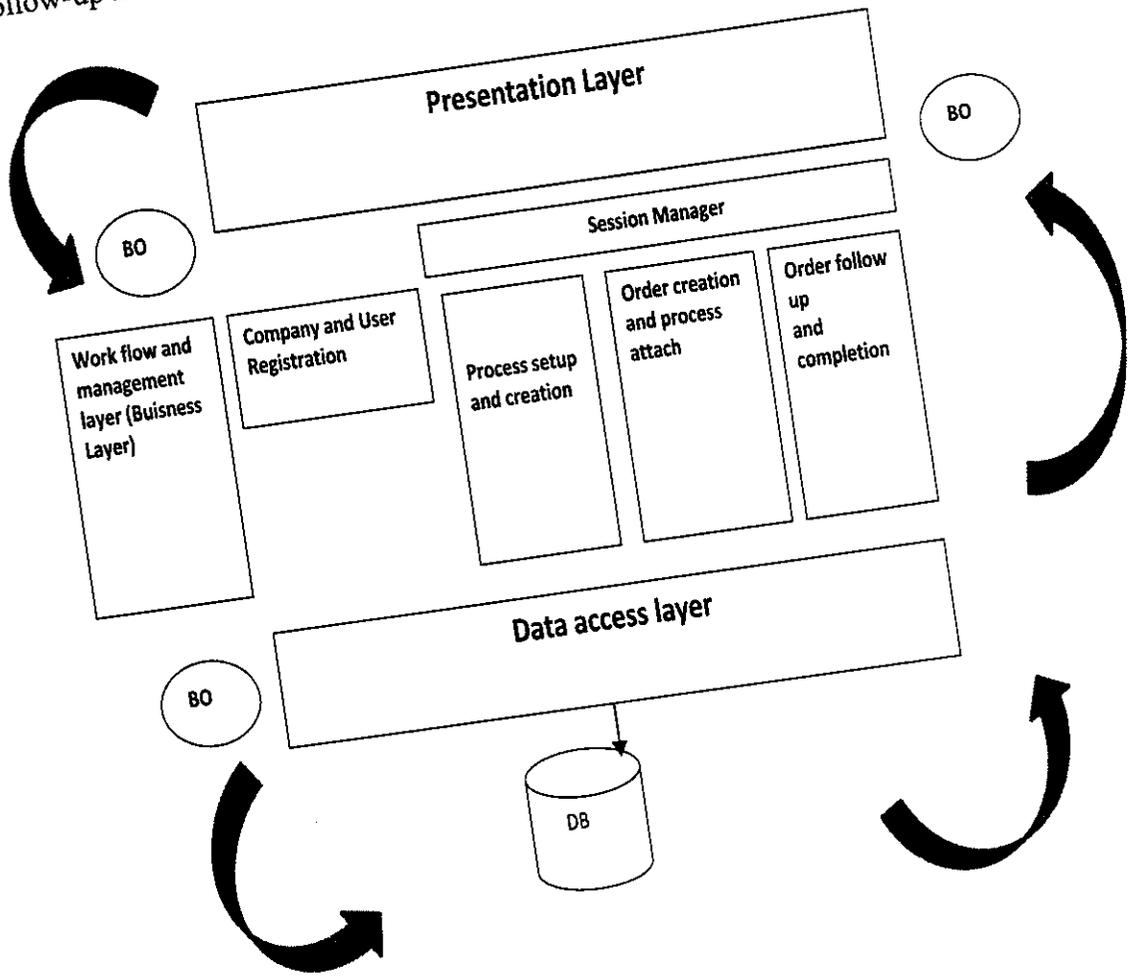
Architecture diagram shows the relationship between different components of system. This diagram is very important to understand the overall concept of system. The architecture that is followed for the e-PDM application consists of 3-tier namely the presentation, the business logic and the data access. Please refer to figure 2.1 Architecture Diagram.

2.1 Architecture Diagram.

The presentation layer consists of the User Interface (UI) which is not but web forms for the user to interact with the system. The data that is in logic layer through the business objects. Any need the data access layer.

The data access layer has close connection with the data base. All data access code pertaining to the system are written in this layer. The presentation logic has no way to access the data in the database directly. This offers greater flexibility and security to the application.

The business logic is just an interface it contains only the interface declaration and all data definition are written in the data access layer. The business logic for e-PDM consists of registration, process setup, order processing and order follow-up interface.



BO – Business Objects

CHAPTER 3

DEVELOPMENT ENVIRONMENT

This chapter describes the hardware and software requirements of the e-PDM application.

3.1 HARDWARE SPECIFICATION

Processor	-	Intel Core 2 Duo
Clock speed	-	2.10 GHz
RAM capacity	-	2.96 GB
Hard disk capacity	-	180 GB

3.2 SOFTWARE SPECIFICATION

Platform	-	Windows XP
Software	-	Microsoft Visual Studio 2005
	-	AJAX Control Toolkit 2.0
Front end	-	C#.NET
Back end	-	MS SQL SERVER 2005

3.3 PROGRAMMING ENVIRONMENT

VISUAL STUDIO.NET

Microsoft Visual Studio is an Integrated Development Environment (IDE) from Microsoft. It can be used to develop console and graphical user interface applications along with Windows Forms applications, web sites, web applications and web services. It supports both native code together with managed code for all platform

supported by Microsoft Windows, Windows Mobile, Windows CE, .NET Framework, .NET Compact Framework and Microsoft Silverlight.

Common Language Runtime

The .NET Framework provides a run-time environment called the common language runtime, which runs the code and provides services that make the development process easier.

Visual C#

Microsoft Visual C# 2005, pronounced C sharp, is a programming language designed for building a wide range of applications that run on the .NET Framework. C# is simple, powerful, type-safe, and object-oriented. With its many innovations, C# enables rapid application development while retaining the expressiveness and elegance of C-style languages.

Some of the advantages of creating C# applications in Visual Studio.NET are

- ✓ Visual Studio.NET is a Rapid Application (RAD) tool. Instead of adding each control to the form programmatically, it helps to add these controls by using Toolbox, saving programming efforts.
- ✓ Visual Studio.NET supports custom and composite controls. Can create custom controls that encapsulate a common functionality that might be used in a number of applications.
- ✓ Visual Studio.NET does a wonderful job of simplifying the creation and consumption of Web Services. Much of the programmer-friendly stuff (creating all the XML-based documents) happens automatically, without much effort on the programmer's side. Attribute based programming is a powerful concept that enables Visual Studio.NET automate a lot of programmer-unfriendly tasks.

ADO.NET

ADO.NET is a set of computer software components that can be used by programmers to access data and data services. It is a part of the base class library that is included with the Microsoft .NET Framework. It is commonly used by programmers to access and modify data stored in relational database systems, though it can also be used to access data in non-relational sources. ADO.NET is sometimes considered an evolution of ActiveX Data Objects (ADO) technology, but was changed so extensively that it can be considered an entirely new product.

The ADO.NET Data Architecture

Data Access in ADO.NET relies on two components: DataSet and Data Provider.

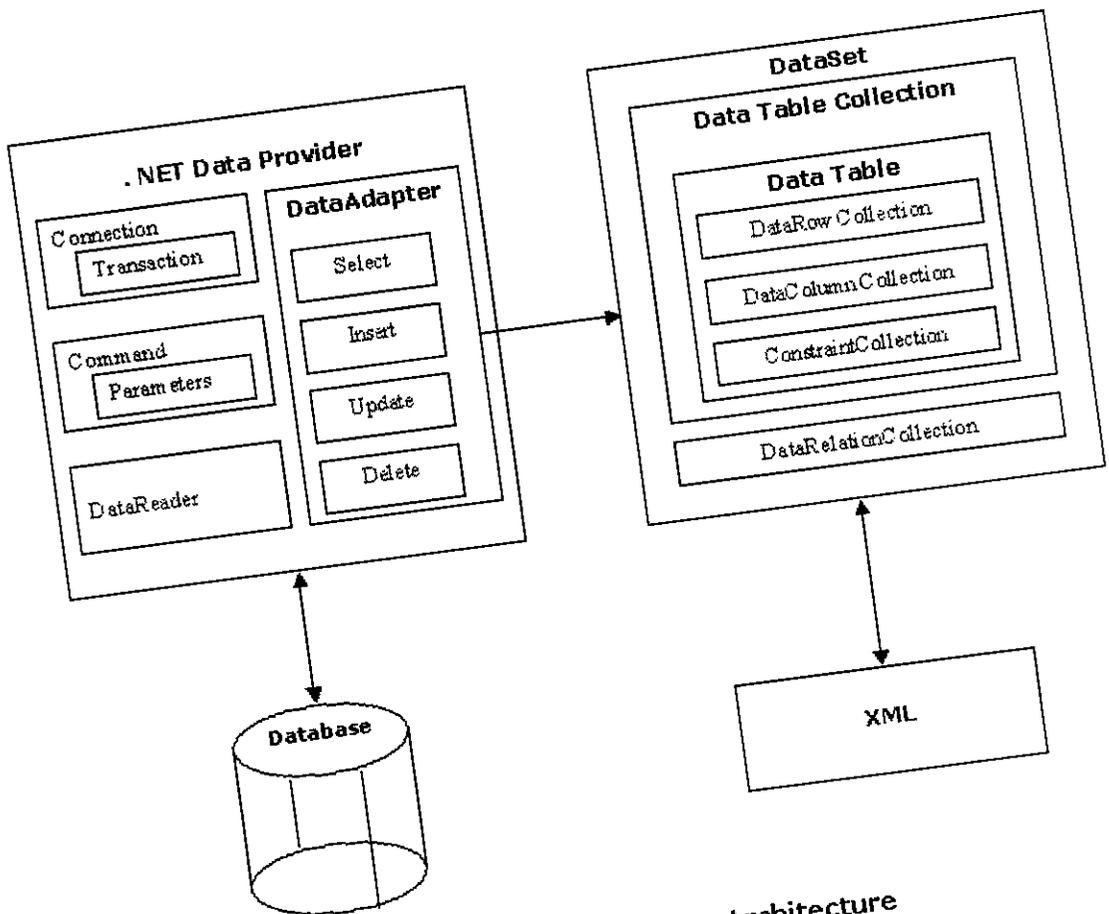
DataSet

The dataset is a disconnected, in-memory representation of data. It can be considered as a local copy of the relevant portions of the database. The DataSet is persisted in memory and the data in it can be manipulated and updated independent of the database. When the use of this DataSet is finished, changes can be made back to the central database for updating. The data in DataSet can be loaded from any valid data source like Microsoft SQL server database, an Oracle database or from a Microsoft data database.

Data Provider

The Data Provider is responsible for providing and maintaining the connection to the database. A DataProvider is a set of related components that work together to provide data in an efficient and performance driven manner. The .NET Framework currently comes with two DataProviders: the SQL Data Provider which is designed only to work with Microsoft's SQL Server 7.0 or later and the OleDb DataProvider which allows us to connect to other types of databases like Access and Oracle. The Data Provider consists of the following component classes:

- ✓ The Command object which is used to execute a command.
- ✓ The DataReader object which provides a forward-only, read only, connected recordset.
- ✓ The DataAdapter object which populates a disconnected DataSet with data and performs update.



ADO .NET Data Architecture

Figure 3.1 ADO.NET Architecture



P-3 231

MS SQL SERVER 2005

Microsoft SQL Server is a relational model database server produced by Microsoft. Its primary query languages are T-SQL and ANSI SQL. Microsoft SQL Server 2005 is a comprehensive, integrated data management and analysis software that enables organizations to reliably manage mission-critical information and confidently run today's increasingly complex business applications.

SQL Server 2005 has a number of tools to help the Database Administrator and the Developer. The relational database engine has been improved to give better performance and support for both structured and unstructured (XML) data. The Replication services include services for distributed or mobile data processing applications. It provides for high systems availability, scalable concurrency with secondary data stores, enterprise reporting solutions and integration with heterogeneous systems such as Oracle databases.

Ajax Control Toolkit

The ASP.NET AJAX Control Toolkit is an open-source project built on top of the Microsoft ASP.NET AJAX framework. It is a joint effort between Microsoft and the ASP.NET AJAX community that provides a powerful infrastructure to write reusable, customizable and extensible ASP.NET AJAX extenders and controls, as well as a rich array of controls that can be used out of the box to create an interactive Web experience.

The AJAX Control Toolkit contains more than 30 controls that enable you to easily create rich, interactive web pages.

ModalPopupExtender

The ModalPopup extender allows a page to display content to the user in a "modal" manner which prevents the user from interacting with the rest of the page. There can be any hierarchy of controls and is displayed above the page. When displayed, only the

modal content can be interacted with; clicking on the rest of the page does nothing. When the user is done interacting with the modal content, a click of an OK/Cancel control dismisses the modal content and optionally runs custom script. The custom script will typically be used to apply whatever changes were made while the modal mode was active. If a postback is required, simply allow the OK/Cancel control to postback and the page to pre-render. We can also absolutely position a modal popup by setting the X and Y properties. By default it is centered on the page, however if just X or Y is specified then it is centered vertically or horizontally.

CalendarExtender

Calendar is an ASP.NET AJAX extender that can be attached to any ASP.NET TextBox control. It provides client-side date-picking functionality with customizable date format and UI in a popup control. We can interact with the calendar by clicking on a day to set the date, or the "Today" link to set the current date.

In addition, the left and right arrows can be used to move forward or back a month. By clicking on the title of the calendar you can change the view from days in the current month, to months in the current year. Another click will switch to Years in the current decade. This action allows you to easily jump to dates in the past or the future from within the calendar control.

CHAPTER 4

SYSTEM DESIGN

This chapter explains the data model and the process model of the application.

System design is the process or art of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements.

4.1 DATA MODEL

A data model in software engineering is an abstract model that describes how data are represented and accessed. Data models formally define data elements and relationships among data elements for a domain of interest. A data model explicitly determines the structure of data or structured data.

4.1.1 Table Structure

The following are the list of tables (structured data) that assists data representation and access.

Table name: TBLCOMPANY

Description: This table stores company details and branch details. If the PARENTID is 0 then its company information with PARENTID having COMPANYID represents branch.

S.no	Column name	Data type	Allow null? (Y/N)	Constraint
1	Companyid	Int	N	Primary Key, Identity(1,1)
	Name	nvarchar(100)	N	
			N	

4	Country	nvarchar(50)	N	
5	Phone	nvarchar(50)	N	
6	Mobile	nvarchar(50)	Y	
7	Fax	nvarchar(50)	N	
8	Email	nvarchar(50)	N	
9	Contact	nvarchar(50)	Y	
10	description	ntext	Y	
11	isactive	Bit	N	
12	isdeleted	Bit	N	
13	type	Bit	N	
14	parentid	Int	N	
15	updatetime	datetime	N	

Table 4.1.1.1 Company Table

Table name: TBLDEPT

Description: This table stores information about departments.

S.no	Column name	Data type	Allow nulls? (Y/N)	Constraint
1	Deptid	Int	N	Primary Key, Identity(1,1)
2	Companyid	Int	N	Foreign key (TBLCOMPANY)
3	Dept code	nchar(5)	N	
4	Name	nvarchar(50)	N	
5	Description	nvarchar(100)	Y	

7	Isdeleted	Bit	N	
8	Updatetime	Datetime	N	

Table 4.1.1.2 Department table

Table name: TBLUSER

Description: This table stores information about users.

S.no	Column name	Data type	Allow null? (Y/N)	Constraint
1	Userid	Int	N	Primary Key, Identity(1,1)
2	Companyid	Int	N	Foreign key (TBLCOMPANY)
3	Deptid	Int	Y	Foreign key (TBLDEPT)
4	Name	nvarchar(50)	N	
5	Desc	Ntext	Y	
6	Address	nvarchar(200)	N	
7	dob	datetime	N	
8	Phone	nvarchar(50)	Y	
9	Email	nvarchar(50)	N	
10	Password	nvarchar(50)	N	
11	Isactive	Bit	N	
12	Isdeleted	Bit	N	
13	Updatetime	datetime	N	
14	Designation	nvarcha(50)	Y	
15	Loginname	nvarchar(50)	N	
16	Type	nvarchhar(50)	Y	

Table 4.1.1.3 User table

Table name: TBLPROCESSTYPE

Description: This table contains information about process types.

S.no	Column name	Data type	Allow null? (Y/N)	Constraint
1	Processtypeid	Int	N	Primary Key, Identity(1,1)
2	Companyid	Int	N	Foreign key (TBLCOMPAN Y)
3	Processtype	Int	N	
4	Typedesc	nvarchar(50)	Y	
5	Updateby	Int	Y	
6	Updatetime	datetime	N	

Table 4.1.1.4 Process Type table

Table name: TBLPROCESSTEPS

Description: This table contains information about Process type's process steps.

S.no	Column name	Data type	Allow null? (Y/N)	Constraint
1	Processstepid	Int	N	Primary Key, Identity(1,1)
2	Processtypeid	Int	N	Foreign key (TBLPROCESST YPE)
			N	Foreign key

				(TBLCOMPANY)
4	Stepnumber	Int	N	
5	Steprefname	nvarchar(50)	N	
6	Stepdesc	Ntext	Y	
7	Planneddays	Int	N	
8	Bufferdays	Int	N	
9	Resptype	nvarchar(50)	Y	
10	Respname	nvarchar(50)	Y	
11	Stepmemo	nvarchar(50)	Y	
12	Basestepnum	Int	Y	
13	Basedate	nvarchar(50)	Y	

Table 4.1.1.5 Process steps table

Table name: TBLORDERS

Description: This table contains order details.

S.no	Column name	Data type	Allow null? (Y/N)	Constraint
1	Ordered	Int	N	Primary Key, Identity(1,1)
2	Companyid	Int	N	Foreign key (TBLCOMPANY)
3	order_desc	ntext	Y	
4	Ordernum	nvarchar(50)	Y	
5	confirm_proce ss	Bit	N	
6	order_status	Bit	N	
7	Collectionnu	nvarchar(50)	Y	

8	Articlenum	nvarchar(50)	Y	
9	Articledesc	nvarchar(50)	Y	
10	production_group	nvarchar(50)	Y	
11	Stylenum	nvarchar(50)	Y	
12	Styledesc	nvarchar(50)	Y	
13	Supplierid	nvarchar(50)	Y	
14	Agented	nvarchar(50)	Y	
15	Buyerid	nvarchar(50)	Y	
16	Orderdate	datetime	Y	
17	Processstartdate	datetime	Y	
18	Etd	datetime	Y	
19	Eta	datetime	Y	
20	Buyerdelivery	datetime	Y	
21	Orderqty	Int	Y	
22	Shippedqty	Int	Y	
23	Agegroup	nvarchar(50)	Y	
24	option2	nvarchar(50)	Y	
25	Shipment	nvarchar(50)	Y	

Table 4.1.1.6 Orders table

Table name: TBLORDRPROCESS

Description: This table contains information about process steps and calculated dates on connecting process type with order.

S.no	Column name	Data type	Allow null? (Y/N)	Constraint
		Int	N	Primary Key

	d			Identity(1,1)
2	Companyid	Int	N	Foreign key (TBLCOMPAN Y)
3	Ordered	Int	N	Foreign key (TBLORDERS)
4	Processtypeid	Int	N	Foreign key (TBLPROCES STYEPE)
5	Stepnum	Int	N	
6	Steprefname	nvarchar(50)	Y	
7	Stepdesc	ntext	Y	
8	Planneddays	Int	Y	
9	Bufferdays	Int	Y	
10	Resptype	nvarchar(50)	Y	
11	Respname	nvarchar(50)	Y	
12	Basestepnum	Int	Y	
13	Basedate	nvarchar(50)	Y	
14	Calcplanned date	datetime	Y	
15	Calcbufferdat e	datetime	Y	
16	Followupplan neddate	datetime	Y	
17	Followupbuff date	datetime	Y	
18	Remainderdat e	datetime	Y	
19	Closeddate	datetime	Y	
20	Processstepid	int	N	Foreign key (TBLPROCES

21	Toggle	bit	N	
22	Allow	bit	N	

Table 4.1.1.7 Order Process table

Table name: TBLFUNCTION

Description: This table contains information about every webpage which is seen as a function for allocating user permissions.

S.no	Column name	Data type	Allow null? (Y/N)	Constraint
1	Functionid	int	N	Primary Key, Identity(1,1)
2	Code	nvarchar(50)	N	
3	Name	nvarchar(50)	N	
4	Desc	ntext	Y	
5	Type	int	Y	
6	url	nvarchar(200)	N	
7	Parented	int	N	
8	updatetime	datetime	N	

Table 4.1.1.8 Function table

Table name: TBLFUNDET

Description: This table contains details about allocating permissions like read only, edit/update, and delete permissions.

S.no	Column name	Data type	Allow null? (Y/N)	Constraint
1	functionid	int	N	Primary Key, Identity(1,1)
2	Code	nvarchar(50)	N	
3	Name	nvarchar(50)	N	
4	Desc	ntext	Y	
5	Type	int	Y	
6	url	nvarchar(200)	N	
7	parentid	Int	N	
8	updatetime	Datetime	N	

Table 4.1.1.9 Function details table

4.2 DATA FLOW DIAGRAM

Data flow diagrams illustrate how data is processed by a system in terms of inputs and outputs. The e-PDM application is illustrated in a precise way by the following DataFlow diagrams Level-0 through Level-1.

Level-0 DFD

The first level DFD shows the main processes within the system. Each of these processes can be broken into further processes until we reach pseudocode.

In this e-PDM application the external entities like buyer, agent, merchandiser, supplier are generalized as users.

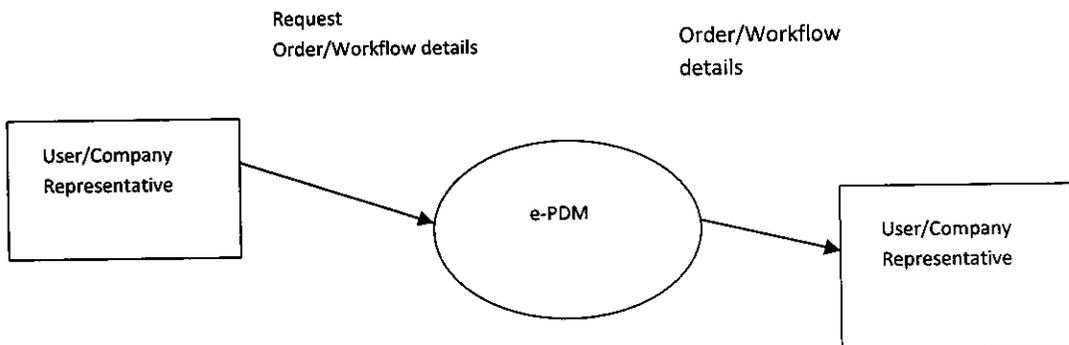


Figure 4.2.1 Level-0 DFD for e-PDM

This Level-0 DFD (refer to figure 4.2.1) of the e-PDM application shows the whole system as just one process. A User or Company representative can acquire login name and password by registering in the system. This login name and password can be used to further setup the system. Once the setup is complete entities with real authentication and authorization can access and modify data in the system.

Level-1 DFD e-PDM application

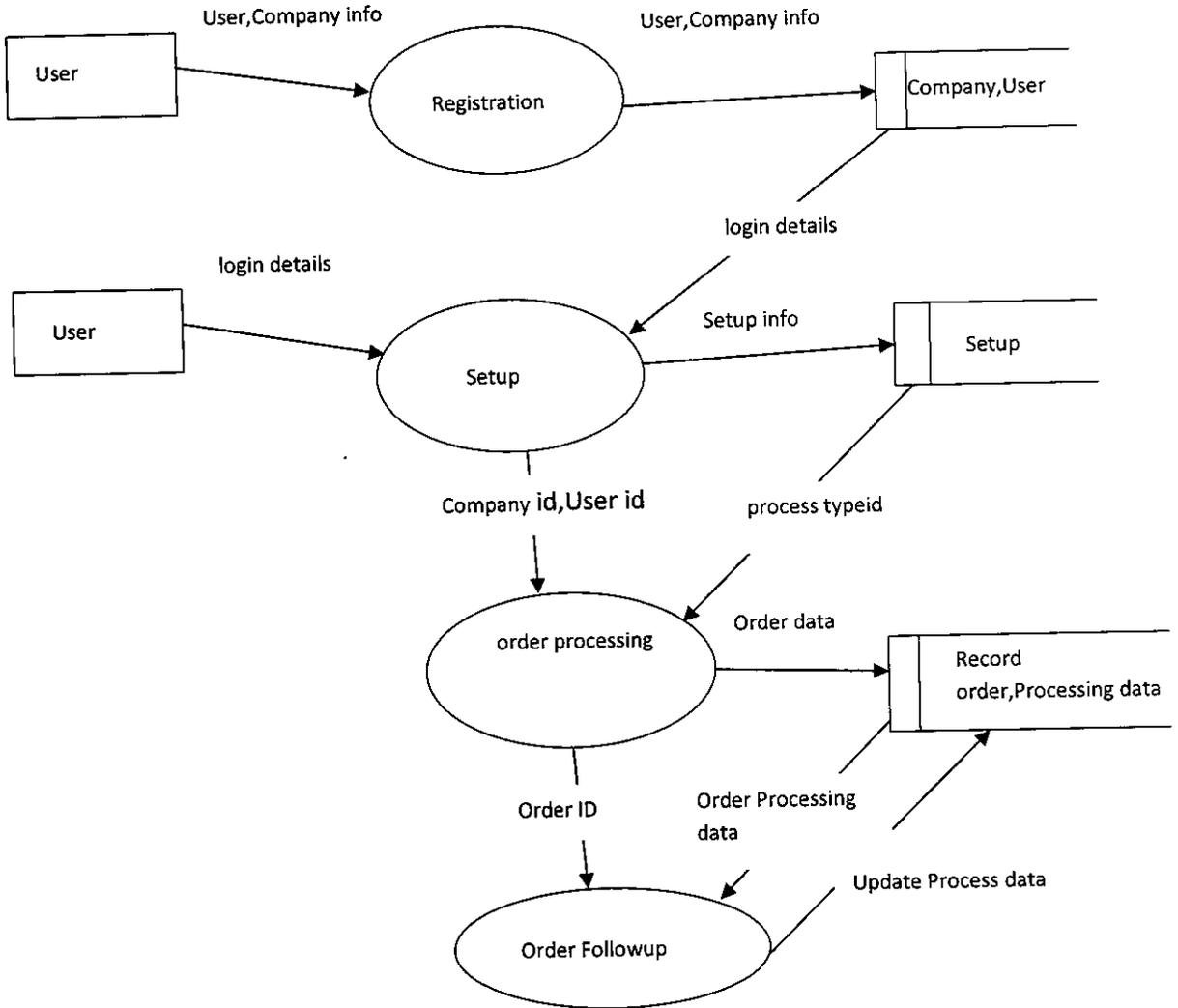


Figure 4.2.2 Level-1 DFD for e-PDM

Level-1 (refer to figure 4.2.2) shows all major high-level processes of the system and their relation. The major processes include Setup which comprises Branch, Department, User, and Process Setup. The data which is stored in the datastore is used for order processing and order followup.

Level-2 DFD e-PDM application

DFD for Registration Process

This level is decomposition of processes shown in level-1. There should be a level-2 diagram for each and every process shown in level-1.

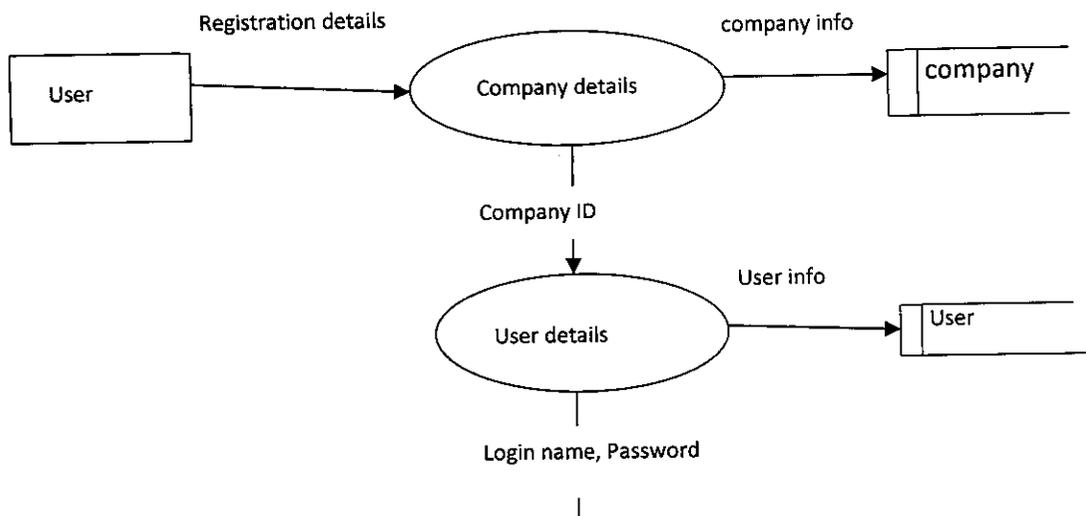


Figure 4.2.3 Level-2 DFD for e-PDM

Level-2 DFD for registration process of the e-PDM application (refer to figure 4.2.3) shows entity user involved in two processes namely company details registration and user details registration. The information that is output is stored in two data stores company and user data store.

Level-2 DFD for Setup

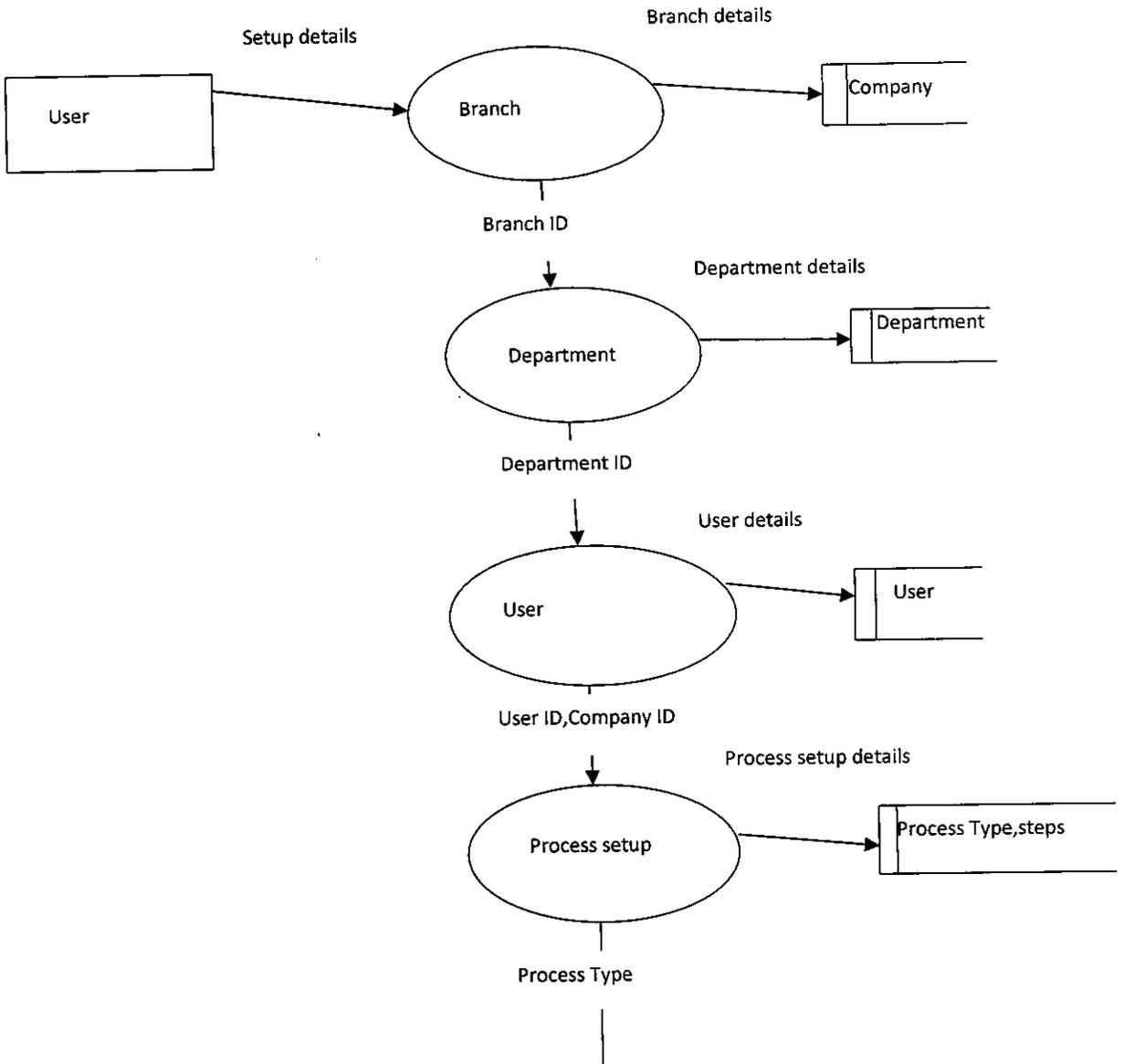


Figure 4.2.4 Level-2 DFD for e-PDM

Level-2 DFD for setup (refer to figure 4.2.4) involves entity user setting up the branch, department, and user details registration. The completion of this initial setup will lead to process setup. The output data of each of these processes are stored in data stores branch, department, user, process type and process steps.

DFD for Order Processing

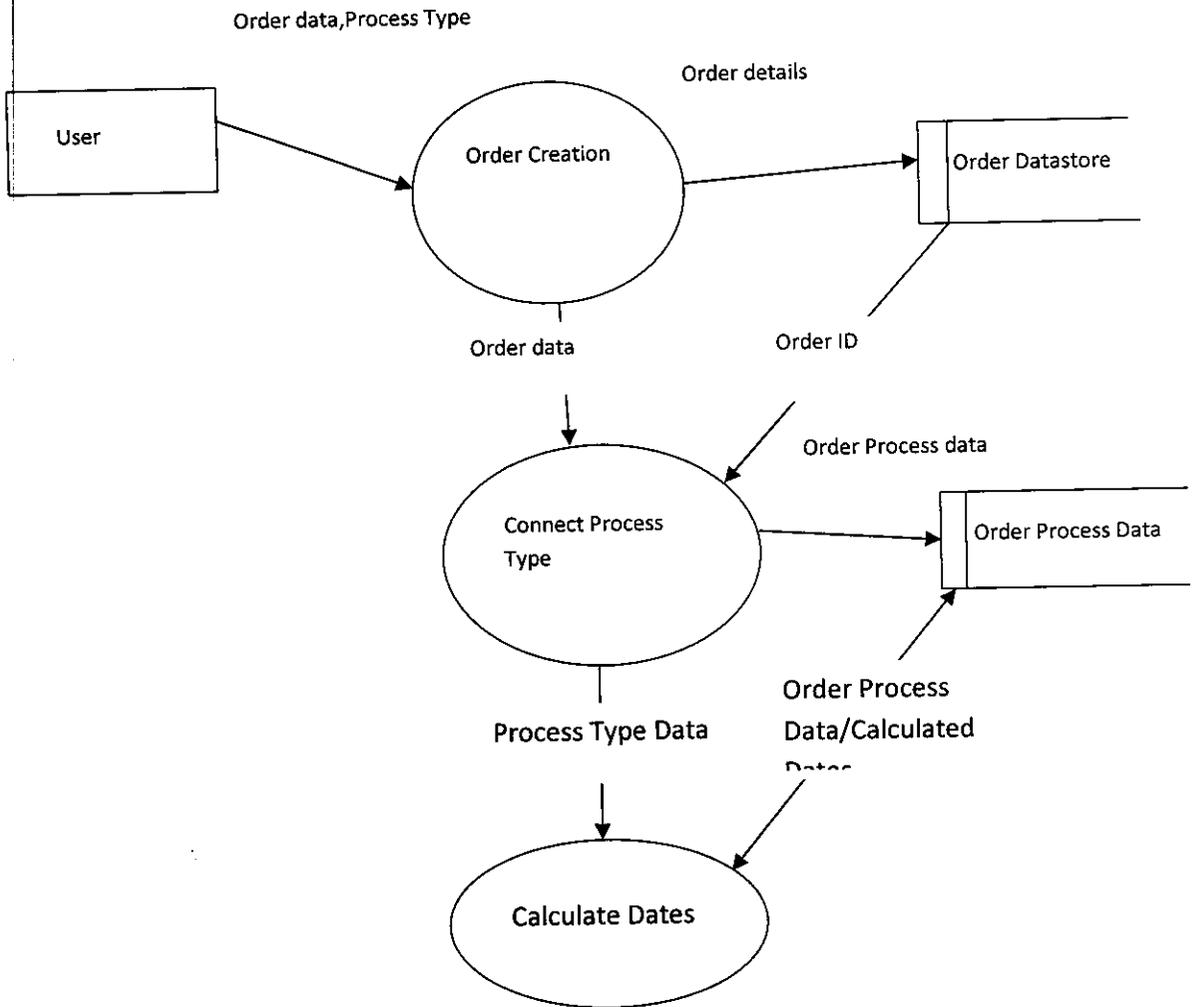


Figure 4.2.5 Level-2 DFD for e-PDM

Level-2 DFD for Order Processing (refer to figure 4.2.5) involves processes like order creation, order processing, calculate dates. Each of these processes uses the data stores in previous DFD's (refer to figure 4.2.4). The data that is output by the process are stored in data stores order and order process data store.

4.3 INPUT AND OUTPUT DESIGN

INPUT DESIGN

Input design is the process of correcting a user-oriented description of the inputs to a computer based one. Inaccurate data is one of the most common causes of data processing errors.

In e-PDM input design has been done in a manner such a way that even a naïve user can interact with the application easily. Any error or mishap is properly portrayed to the user with an appropriate error or warning message.

Input screens of the e-PDM application are shown in the Appendix (A.1,A.4,A.5,A.6,A.7).

PROCESS DESIGN

The software process can be identified as a collection of patterns that define a set of activities, actions, work tasks, work products, and related behaviors required to develop computer software. A process provides a method for describe an important characteristic of the software process. By combining patterns, the process that meets the need can be constructed.

The major process involved in this project is connecting an order with a process type which eventually leads to order followup. This data forms the basic platform to for further processes in the application.

OUTPUT DESIGN

An application is successful only when it provides the desired output to the user. In e-PDM the output is promptly displayed in a grid view which allows for editing and updating functionality based on the user rights. Grid view which displays data in rows and columns format is flexible control for displaying when the amount of data to handle is large.

The followup screen (refer to screen shot A.8) provides search options depending on the checkbox being checked. The output data is displayed in a grid view. Output screens in the e-PDM application are shown in appendix (A.2, A.6, A,8)

CHAPTER 5

SYSTEM IMPLEMENTATION

This chapter explains about how the implementation process of SDLC (Software Development Life Cycle) of the e-PDM application is carried out. Implementation converts a design into a complete information system. This includes acquiring and installing systems environment; Creating and testing databases; preparing test case procedures; Preparing coding, test files, compiling and refining programs. Coding is done in this stage using an apt frame work and programming language, which would solve the problem in a best way. Once the design is coded into a working application, it has to be verified, validated and tested in detail. The tested product if successful is deployed in the user environment.

5.1 Implementation Environment

After the design document is completed it is decided to implement the application using .NET framework. ASP.NET allows programmers to build dynamic web applications and implement Web Services. The code behind is implemented using C#.NET which is robust and assists in implementing .NET framework components. Layered approach is followed here wherein presentation, business logic and data access are separated. This brings many advantages, including cleaner code, increased maintainability and the ability to spread our application's load over multiple servers.

The processes involved in 3-tier architecture are:

1. The Presentation layer asks the BLL (Business Logic Layer) for some object.
2. The BLL may perform some validation and forwards the request to the DAL(Data Access Layer).

4. When the record is found, it is returned from the database to the DAL.
5. The DAL wraps the database data in a custom object (in our case we use Type Dataset) and returns it to the BLL.
6. Finally, the BLL returns the object to the Presentation layer, where it could be displayed on a web page for example.

SQL SERVER 2005 is opted to deploy the e-PDM database since ASP.NET 2.0 goes well with the SQL SERVER 2005. Stored Procedures are used for data access in the e-PDM application. All insert, update, delete and select are carried out using stored procedures.

AJAXCONTROLTOOLKIT 2.0 a third party component is installed to implement AJAX features in the application. AJAX technology controls unnecessary postbacks which controls flickering effects in webpages and supports in design of the web application.

Web Services

- ✓ Web services are application components
- ✓ Web services communicate using open protocols
- ✓ Web services are self-contained and self-describing
- ✓ Web services can be discovered using UDDI
- ✓ Web services can be used by other applications
- ✓ XML is the basis for Web services

Web service can be used in .NET framework by including the .asmx file into our website. This file contains a [WebMethod] part. We can write our own custom methods inside this part to implement web services. The data is sent out as XML data(unstructured data) which can be manipulated in the presentation layer to display aptly to the user. Web services can be implemented by inheriting the following namespace: [System.Web.Script.Services.ScriptService].

CHAPTER 6

TESTING

Software testing is any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results. It is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs.

Software testing can also be stated as the process of validating and verifying that a software program/application/product:

1. meets the business and technical requirements that guided its design and development;
2. works as expected; and
3. can be implemented with the same characteristics.

In software testing, Verification and Validation is the process of checking that a software system meets specifications and that it fulfills its intended purpose.

6.1 VERIFICATION

According to Capability Maturity Model (CMM), "Verification is the process of evaluating software to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase".

6.1.1 Verification Phases

Requirements Analysis Phase

In the Requirements analysis phase, the requirements of the proposed system are established based on the needs of the user(s). In a PDM functional

requirements were collected by interviewing a merchandiser in a garment manufacturing company in Perundurai and system requirements through internet.

System Design

As a part of system design, Software Requirements Document (SRS) was prepared before entering the development phase. The requirements were found to be consistent and feasible before development. The SRS serves as a blueprint to further development.

Architecture Design

Architecture design includes programming standard followed and design of database. In e-PDM, 3-tier layered approach is followed and database is normalized to 3-Normal form. This phase of design is referred to as high-level design.

Module Design

Module design can also be referred to as low-level design. The designed system is broken up into smaller units or modules and each of them is explained so that the programmer can start coding directly.

In e-PDM the following modules were identified:

- ✓ Registration
- ✓ Process Setup
- ✓ Order Creation
- ✓ Order Processing
- ✓ Order Follow up

6.2 VALIDATION

According to Capability Maturity Model (CMM), "The process of evaluating software during or at the end of the development process to determine whether it satisfies specified requirements".

6.2.1 Validation Phases

Unit Testing

Unit testing is a software development process in which the smallest testable parts of an application, called units, are individually and independently scrutinized for proper operation. Unit testing is often automated but it can also be done manually. A unit is the smallest testable part of an application.

In e-PDM, each web form is tested separately as a unit. The tracing and debugging feature of ASP.NET allows us to follow a page's execution path, display diagnostic information at run time, and debug our application. The following classes in .NET provide us the feature of tracing and debugging:

- ✓ **System.Web.TraceContext class**
- ✓ **System.Diagnostics.Trace and System.Diagnostics.Debug classes**

Integration Testing

Once all of the units in a program have been found to be working in the most efficient and error-free manner possible, larger components of the program can be evaluated by means of integration testing.

In e-PDM, top-down integration testing is followed. Modules were linked to

in the real-time operating mode of the

system. The process is continued from page level to module level and finally to system level. In the final stage the whole system is taken together and tested for

integration. In e-PDM it is tested whether the data flow through the interfaces (BLL and DAL) is consistent and data from database is also consistent.

CHAPTER 7

CONCLUSION

This chapter explains of the merits and demerits of the e-PDM application and ends with a conclusion.

e-PDM application software is designed to create a generalized work flow system with user rights included. This application provides security privileges in order to avoid unauthorized access of the system. This application maintains history of data and can be imported to an Excel file. e-PDM application can be used in Garment manufacturing units to effectively track workflow as it is an industry based on 60-90 days cycle. This brings buyer and the supplier closer leading to enhanced production and clear view of everyday tasks.

7.1 MERITS OF THE APPLICATION

- ✓ Simple and easy to use User Interface (UI).
- ✓ Grouping, User rights, authentication and authorization.
- ✓ Layered architecture so application can be distributed across multiple server locations.
- ✓ Web Service methods provide easy and speedy data access from literally remote locations.
- ✓ XML data transfer allows fast moving of data.
- ✓ AJAX features helps to decrease load of the server.
- ✓ Enhancements can be added easily.
- ✓ Understandability and maintainability since 3-tier.
- ✓ Changes to database are easy retaining the presentation.

7.2 LIMITATIONS OF THE APPLICATION

- ✓ Security is an issue since data is not stored in an encrypted format.
- ✓ Chance for SQL injection.
- ✓ Not too many members can login at one time since load of server can cause server to hang.

7.3 FUTURE ENHANCEMENTS

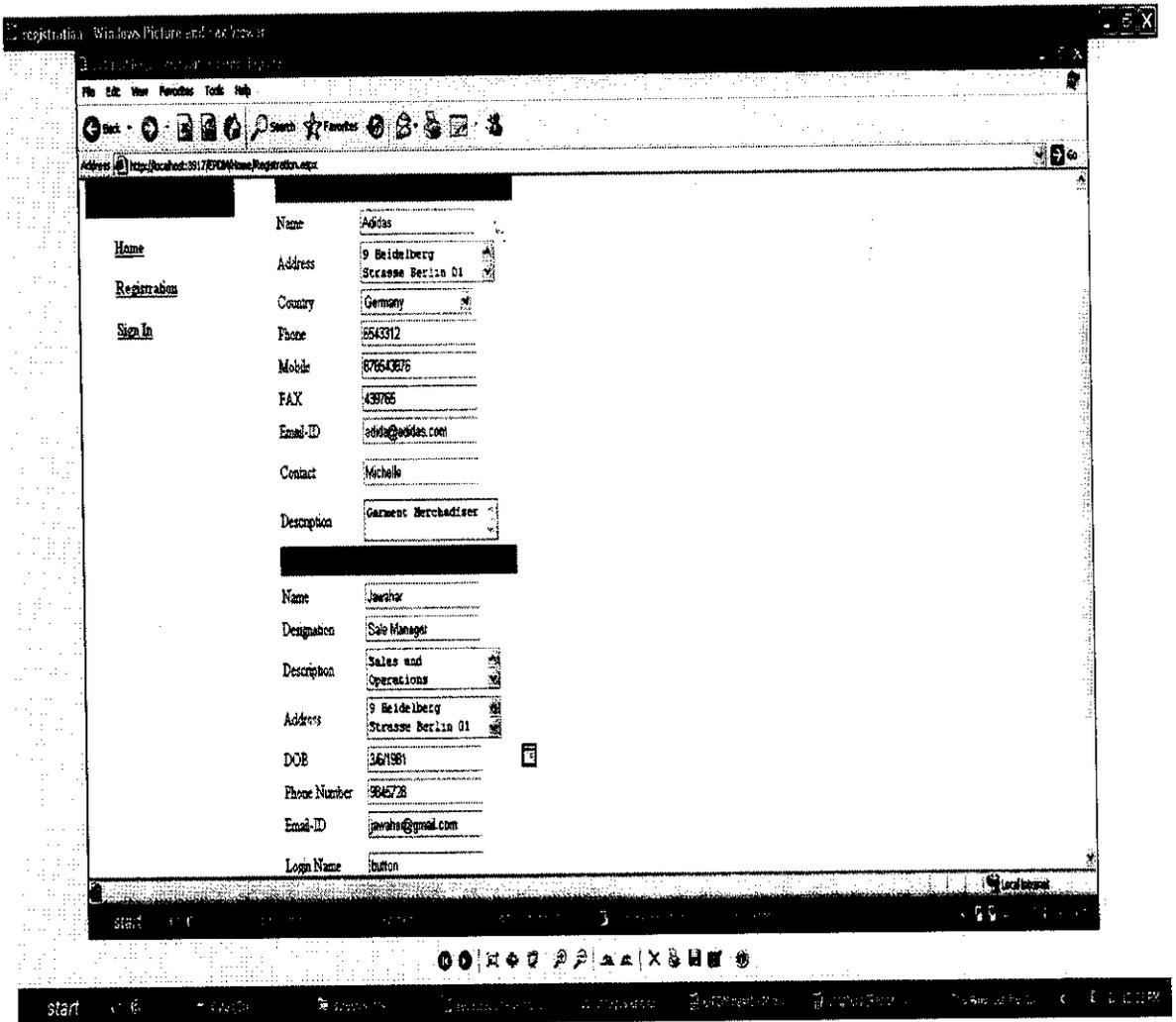
The e-PDM application would be made to implement Web services component so that online payment services can be included more securely. Any organization who would wish to implement any of the services of the e-PDM application software can do it with the availability of Web services component.

Payment calculation on every orders including Euro and Dollar conversions would be implemented. More provisions for entering order details would be included.

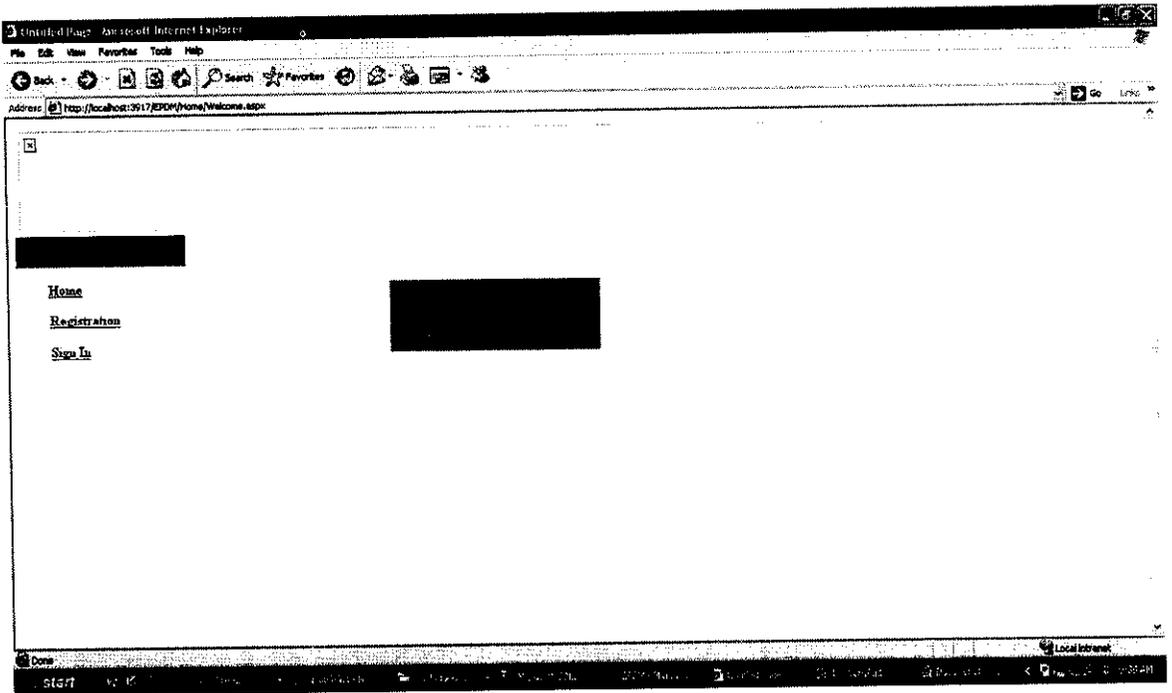
APPENDIX

SAMPLE SCREENS

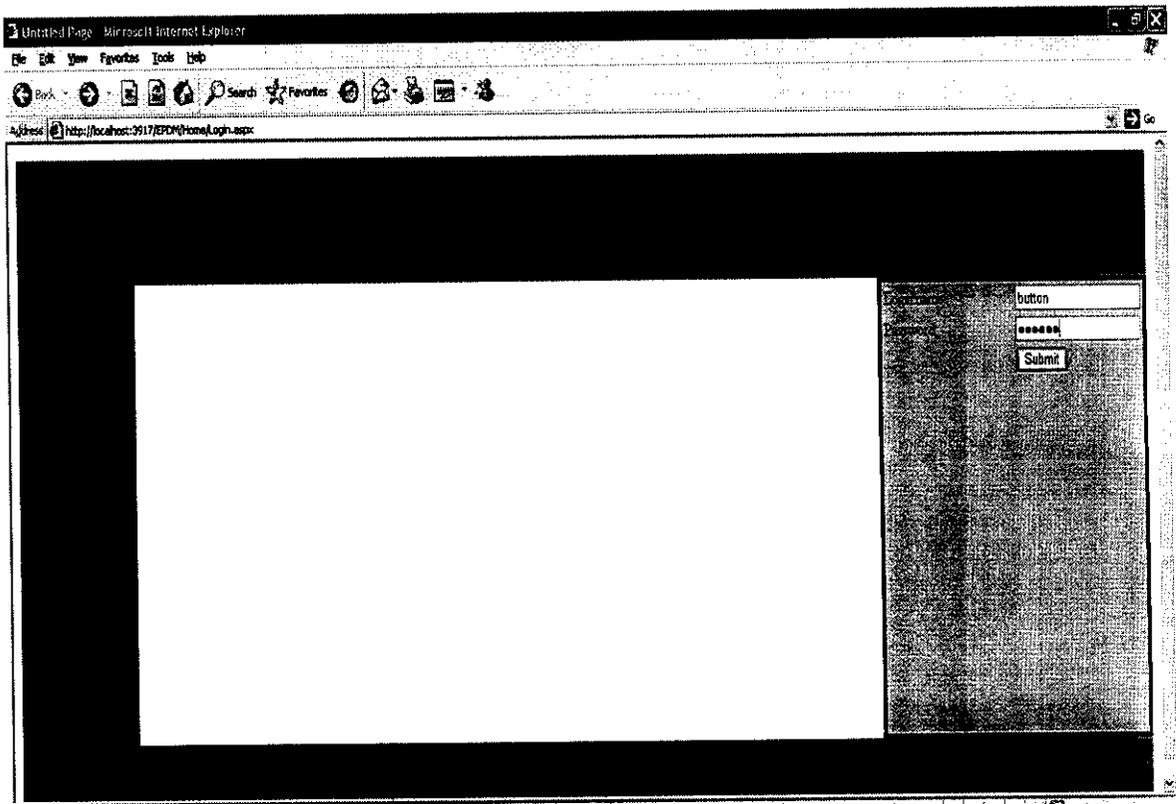
A.1 COMPANY AND USER REGISTRATION:



A.2 REGISTRATION MESSAGE :



A.3 LOGIN:



A.4 BRANCH REGISTRATION:

Untitled Page - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites Print Mail

Address http://localhost:9117/EPDM/Setup/Branch.aspx

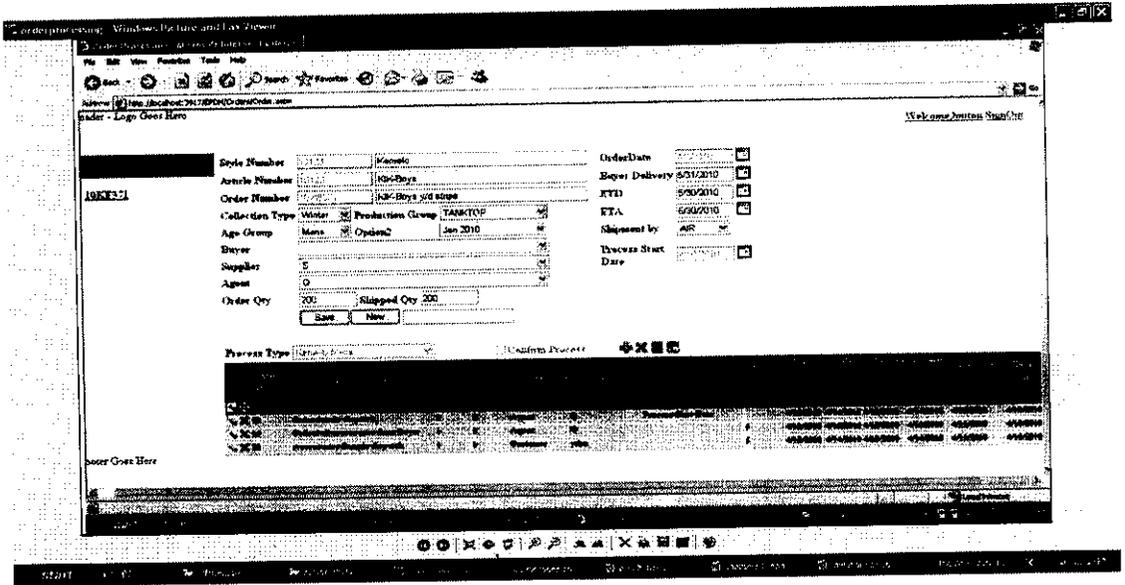
Header - Logo Goes Here Welcome john anho [SignOut](#)

Branch	Name	<input type="text" value="gy tech"/>
Department	Address	<input type="text" value="tripur"/>
User	Country	<input type="text" value="India"/>
Process Setup	Phone	<input type="text" value="5435365"/>
Order Creation	Mobile	<input type="text" value="975368766"/>
Order Processing	FAX	<input type="text" value="34354"/>
Followup	Email	<input type="text" value="fgy.bgy@gmail.com"/>
	Contact	<input type="text" value="sales reporter"/>
	Description	<input type="text" value="qty checking"/>
		<input type="button" value="Submit"/>

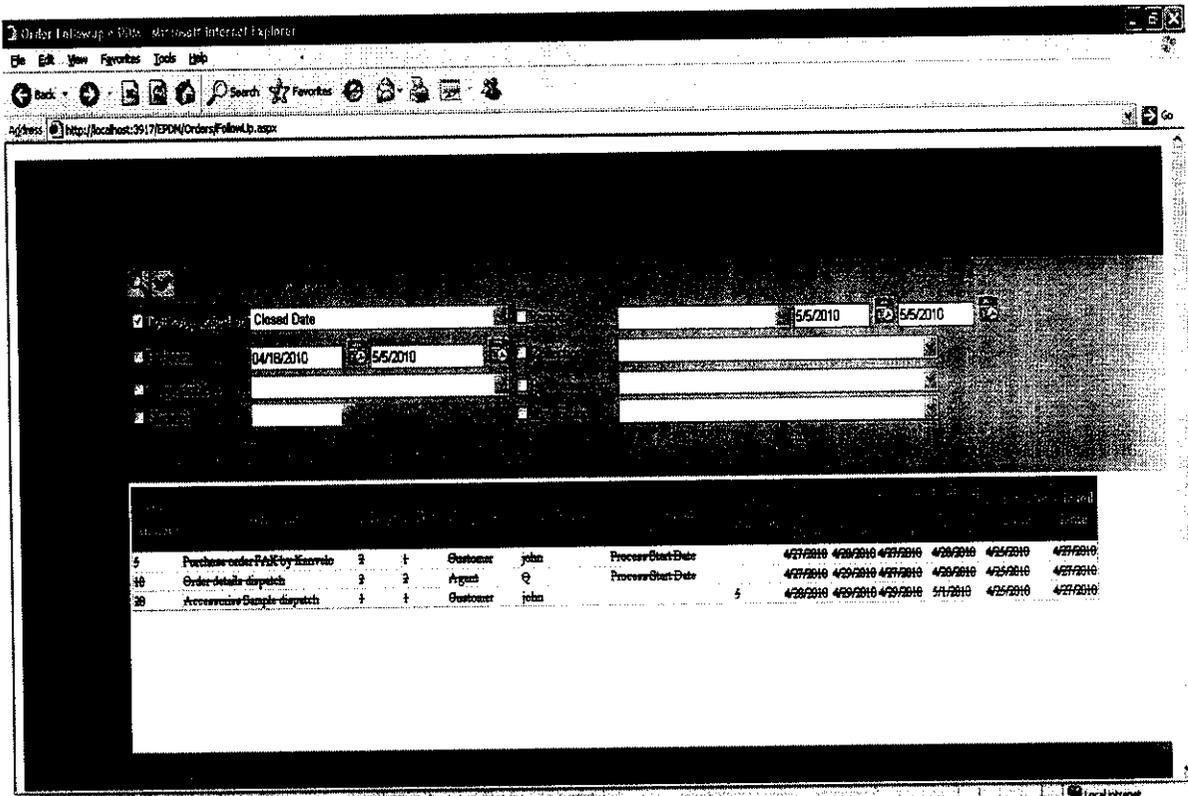
Footer Goes Here Local Intranet

start x 0

A.7 ORDER PROCESSING:



A.8 ORDER FOLLOWUP:



REFERENCES

REFERENCE BOOKS

- ✓ Bill Evjen, Scott Hanselman, Farhan Muhammad, Devin Rader, Srinivasa Sivakumar. (2002) 'Professional ASP.NET 2.0' , McGraw Hill

- ✓ Matt Gibbs, Dan Wahlin. (2004) 'Professional ASP.NET 2.0 AJAX', McGraw Hill

WEBSITES

- ✓ <http://www.codeproject.com>
- ✓ <http://imar.spaanjaars.com/BuildingLayeredWebApplications>
- ✓ <http://msdn.microsoft.com>