DESIGN AND DEVELOPMENT OF A SOFTWARE FOR PRODUCTION PLANNING IN TEXTILE INDUSTRIES

By
Md. Habibur Rahman

POST GRADUATE DIPLOMA IN INFORMATION AND COMMUNICATION TECHNOLOGY

Institute of Information and Communication Technology
BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY
March, 2016

The project report titled "Design and Development of a Software for Production Planning in Textile Industries" submitted by Md. Habibur Rahman, Roll No: 1008311047, Session 2008-2009, has been accepted as satisfactory in partial fulfillment of the requirement for Postgraduate Diploma (ICT) held on 6 March, 2016.

BOARD OF EXAMINERS

1. Dr. Md. Liakot Ali — Chairman
   Professor and Director
   Institute of Information and Communication Technology
   BUET, Dhaka – 1000

2. Dr. Muhammad Mahbubul Alam — Member
   Professor
   Department of Mechanical Engineering
   BUET, Dhaka – 1000

3. Dr. Mohammad Shah Alam — Member
   Assistant Professor
   Institute of Information and Communication Technology
   BUET, Dhaka – 1000

CANDIDATE'S DECLARATION

It is hereby declared that this project report or any part of it has not been submitted elsewhere for the award of any degree or diploma.

Md. Habibur Rahman

Dedicated

To

My Parents and Family

Table of Contents

Appendix B

List of Tables

List of Figures

List of Abbreviations

HTTP	HyperText Transport Protocol
PHP	Preprocessor HyperText
XML	Extensible Markup Language
HTML	HyperText Markup Language
ERD	Entity Relationship Diagram
UML	Unified Modeling Language
PPS	Production Planning System
SRS	Software Requirements and Specification

ACKNOWLEDGEMENTS

First of all I would like to thank Almighty Allah who has given me the ability, strength, patience, chance that carried me through the work and enabled me to complete this project.

I would like to express my deepest gratitude to my project supervisor Dr. Md. Liakot Ali, Professor and Director, IICT, for his continuous guidance, advice, encouragement and help throughout the work. I would also like to convey my thanks to Dr. Md. Shah Alam, Assistant Professor, IICT for his motivation and inspiration which gave me the courage to do this work.

I am grateful to my family, particularly my parents, for their beautiful love, prayers, and support. I would also like to thank all of my friends specially Sultan Mahmud for giving his valuable time during the development period of the project. My thanks also go to my wife Lailatul Ferdous and all the staffs of IICT, BUET.

ABSTRACT

Production planning is a very important phase of production for a textile industry. It is basically the process of determining a tentative plan for how much production will occur in the next several time periods, during an interval of time called the planning horizon. Proper production planning ensures that the resources are used in the best possible manner and the cost of production is minimized and delivery date is maintained. It ensures maximum utilization of machine, labor and material and minimizes idle time, overtime, process delays, lead time, and work-in-process (WIP) inventory and organizes production schedule in conformity with the demand forecasts. To implement proper production planning, computer automation is very much necessary. Few commercial automation software for textile industries have already in the market but they are very expensive and planning phase is not integrated with it. The project in this report presents the development of software for production planning purpose and it has been integrated with the commercial software named as DyeMaster which are widely used in textile industry. To develop the software system, the requirements for the software has been analyzed and then E-R and dataflow diagram of the proposed system have been drawn and then necessary table for the database have been determined and designed. Based on this design computer programming of the system has been developed. It provides a user friendly interface for (i) Order Entry (ii) Item wise auto batch creation (iii) machine batch planning (iv) Production Process Tracking (v) Production plan report. The software is expected to facilitate efficient production management for any textile industry.

Chapter 1

Introduction

1.1 Introduction

Now-a-days, production planning is a very important part of production management in any big or small industries specially in textile/yarn dying industries because companies have to react to dynamic market conditions and fulfill customers´ requirements for shorter delivery times, lower prices and better quality and services [1]. Production planning is basically the process of determining a tentative plan for how much production will occur in the next several time periods, during an interval of time called the planning horizon. Proper production planning ensures that the resources are used in the best possible manner and the cost of production is minimized and delivery date is maintained [2]. It is expected that if a textile industry adopts automation in production planning then it will provide the following facilities:

- Improved customer service and satisfaction.
- Increased flexibility in operations.
- Improved resource utility, reduced quality cost and information accuracy.
- Improved decision making processes due to availability of online information.
- Benefit for the clients to publish the information about their order status.
- Benefit of order execution status at all levels through interactive Plan board.
- Allows monitoring of consumption of Raw materials.
- Quickly identify stock position & stock-out situation and take necessary action.
- Allows monitoring of daily/monthly report at any time with data accuracy.
- Minimize duplicate data entry works
- Consistent format by avoiding human errors in the manual system.
- Improves supply chain monitoring and client relationship management at all levels
- Greater and effective control on accounts payable through better invoicing and payment processing.
- Reduction in paperwork because of online formats for entering and receiving information and improved cost control. Single write, multi read data.
- Quicker response and follow up with customers.

- Availability of accurate information with detailed content and better presentation timely.
- Better monitoring and quicker resolution of queries from within and outside.
- Quick response to change in business operations and market consumption.
- Improved business processes providing a competitive advantage
- Improved supply demand linkage with remote locations and branches in other countries.

DyeMaster is one of the best user friendly software for yarn dyeing industries. This software is developed for integrated management information system for textile industries specially yarn dyeing and knit composite textile mills. The software already covered the fields like Procurement, Inventory Control, Marketing, Production, Dyeline, Stores, LCs Management, Delivery, Shipment, Cash Bill and Payment etc. There is major drawback at this software, how much time is machine engaged at production line and the remaining orders input schedule is not confirmed. So there are scope of research and development to develop software in this area. The project presented in this report is a software that can be integrated with DyeMaster for production planning purpose.

1.2 Objectives with Specific Aims and Possible Outcome

To develop Production Planning software we can get following objectives:

- To develop a production planning software for Textile Industries.
- To design a flexible front end system where user can easily posted transactional data by visual mode.
- To implement the system in a three tier client server environment.
- To generate various report for the management for understanding the business and decisions making.
- To facilitate for the clients to publish the information about their order status.

To develop Production Planning software we can get following Possible Outcomes:

- Monitoring of order execution status at all levels through interactive Plan board.
- Allows monitoring of consumption of Raw materials.
- Quickly identify stock position & stock-out situation and take necessary action.
- Allows monitoring of daily/monthly report at any time with data accuracy.
- Consistent format by avoiding human errors in the manual system.
- Improves supply chain monitoring and client relationship management at all levels.

1.3 Organization of the Project Report

- Chapter 1 describes the overview and objectives of the project.
- Chapter 2 describes about the project requirements and specifications. Different types of software development models, DBMS are also discussed in this chapter.
- Chapter 3 describes the software system design describes the desired software features in detail, including database design (ER diagram), software design (UML is produced here), screen layouts and other documents to develop the project.
- Chapter 4 describes the System / Software Design, results and discussions of the project. It includes database design, UML diagram and different functional features of the project.
- Finally, conclusion and recommendations/future work for future works are included in Chapter 5.

Chapter 2

System Development Process

2.1 Introduction

This chapter describes the basics of software requirements and its development. The knowledge gained from this chapter will help the System Development Life Cycle (SDLC). A SDLC is essentially a series of steps, or phases, that provide a model for the development and lifecycle management of an application or piece of software. The methodology within the SDLC process can vary across industries and organizations, but standards such as ISO/IEC 12207 represent processes that establish a lifecycle for software, and provide a mode for the development, acquisition, and configuration of software systems. The project is maintained through the stages of SDLC, which is the traditional methodology for developing, maintaining and replacing information system.

Computer systems are complex and often (especially with the recent rise of service-oriented architecture) link multiple traditional systems potentially supplied by different software vendors. To manage this level of complexity, a number of SDLC models or methodologies have been created, such as "waterfall"; "spiral"; "Agile software development"; "rapid prototyping"; "incremental"; and "synchronize and stabilize".

SDLC is used during the development of an IT project; it describes the different stages involved in the project from the drawing board, through the completion of the project.

2.2 Systems development phases

The SDLC framework provides a sequence of activities for system designers and developers to follow. It consists of a set of steps or phases in which each phase of the SDLC uses the results of the previous one.

A SDLC adheres to important phases that are essential for developers, such as planning, analysis, design, and implementation, and are explained in the section below.

Figure 2.1: SDLC phases.

- **Requirement gathering**: in this step, the basic requirement of the project is identified and identifies the problems. Describes the costs and benefits.

- **Analysis**: Defines project goals into defined functions and operation of the intended application. Analyzes end-user information needs.

- **Design**: Describes desired features and operations in detail, including screen layouts, business rules, process diagrams, pseudo code and other documentation.

- **Development**: The real code is written here.

- **Testing**: Brings all the pieces together into a special testing environment, then checks for errors, bugs and interoperability.

- **Implementation**: The final stage of initial development, where the software is put into production and runs actual business.

- **Maintenance**: During the maintenance stage of the Life-Cycle, the system is assessed to ensure it does not become obsolete. This is also where changes are made to initial software. It involves continuous evaluation of the system.

2.3 Software Requirements and Specification (SRS) Document

After the analyst has collected all the requirements information regarding the software to be developed, and has removed all the incompleteness, inconsistencies, and anomalies from the specification, he starts to organize the requirements systematically in the form of an SRS document.

The important parts of the SRS document are:

i. Functional requirements of the system
ii. Non-functional requirements of the system, and
iii. Goals of implementation.

2.3.1 Functional Requirements

The part related to functional requirements discusses the functionalities required in the system. Here we list all high-level functions that the system should perform. Each high-level function is considered as a transformation of a set of input data to some corresponding output data. The user can get some meaningful piece of work done using a high-level function.

2.3.2 Non-Functional Requirements

Non-functional requirements deal with the characteristics of the system which cannot be expressed as functions - such as the maintainability of the system, portability of the system, usability of the system, etc.

Non-functional requirements may include:

i. Reliability issues,
ii. Accuracy of results,
iii. Human-computer interface issues,
iv. Constraints on the system implementation, etc.

2.3.3 Goals of Implementation

The goals of implementation are general suggestions regarding the development. These suggestions guide trade-off among design goals. The goals of implementation section include issues such as revisions to the system functionalities that may be required in future, new devices to be supported in future, reusability issues, etc. These are the items which the developers might keep in their mind during development so that the developed system may meet some aspects that are not required immediately.

2.4 Software System Attributes

2.4.1 Reliability

It is an important attributes for any system. The number of defects should not exceed 10 per function. Before the release of the final version, the system must be tested in case of the defects over 10 per function.

2.4.2 Availability

The availability of this software depends on the internet connection of the client. Since this is client-server related web-site, web-site shall be attainable all the time. User should have an account to enter the system, if user does not have an account; for the availability of this software user should sign up to the system by clicking the sign up link from the home page.

2.4.3 Security

The authorization mechanism of the system will block any unwanted attempts to the server and also let the system decide on which privileges may the user have. The system has different types of users so there are different levels of authorization. There will be also a firewall installed on the server so the incoming transactions can be filtered. Data integrity for critical variables will also be checked.

2.4.4 Maintainability

Maintainability is the ease with which a product can be maintained in order to (i) isolate defects or their cause, (ii) correct defects or their cause, (iii) meet new requirements, (iv) make future maintenance easier, or (v) cope with a changed environment.

In some cases, maintainability involves a system of continuous improvement learning from the past in order to improve the ability to maintain systems, or improve reliability of systems based on maintenance experience.

2.4.5 Portability

This software is an online service. So, anyone can use the service. One and only the server of the system must have the required software including MySQL, Apache.

2.5 Software Development Life Cycle (SDLC) Models

There are various SDLC models defined and designed which are used or employed during development process of software; these approaches are also referred as "Software Development Process Models". Each process model follows a particular life cycle in order to ensure success in the process of software development. Some important SDLC models are described below.

2.5.1 Waterfall Model

Waterfall approach was the first process model to be introduced and followed widely in Software Engineering to ensure success of the project. Figure 2.1 shows different steps of this model.

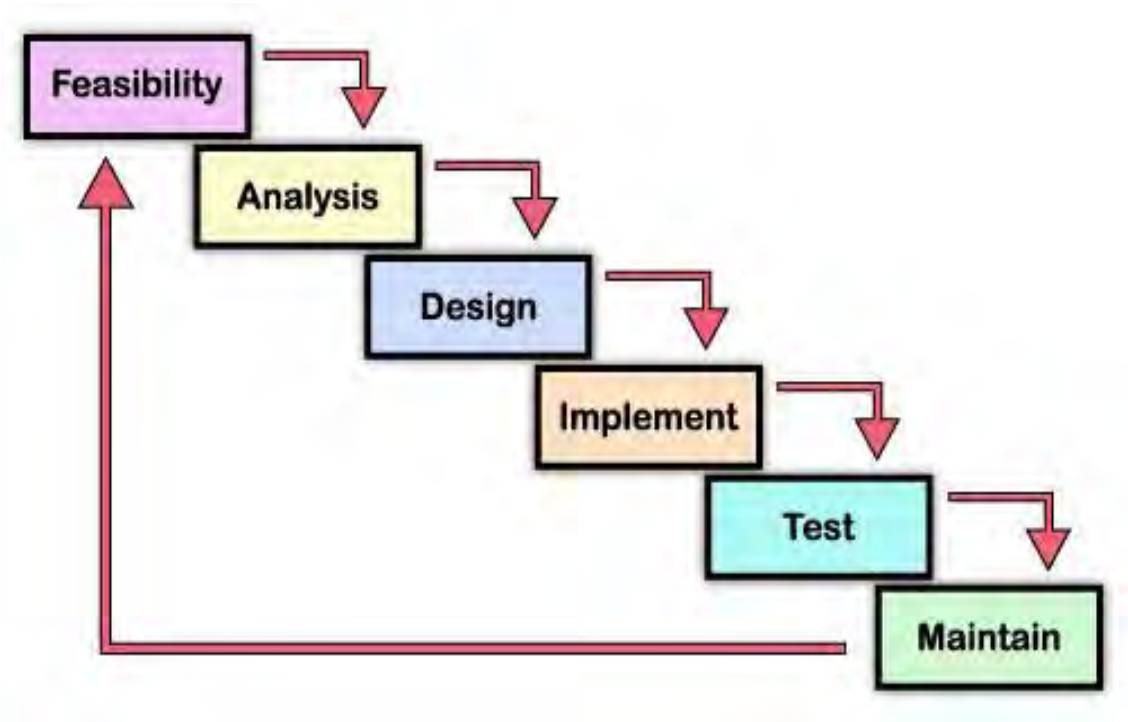


Figure 2.2: Waterfall Model

In "The Waterfall" approach, the whole process of software development is divided into separate process phases. The phases in Waterfall model are: Requirement Specifications phase, Software Design, Implementation, Testing and Maintenance. All these phases are cascaded to each other so that second phase is started as and when defined set of goals are achieved for first phase and it is signed off, so the name "Waterfall Model".

2.5.1.1 Advantages

The advantage of waterfall development is that it allows for departmentalization and has good managerial control. A schedule can be set with deadlines for each stage of development and a product can proceed through the development process, be delivered on time. Development moves from concept, through design, implementation, testing, installation, troubleshooting, and ends up with operation and maintenance. Each phase of development proceeds in strict order, without any overlapping or iterative steps.

2.5.1.2 Disadvantages

The disadvantage of waterfall development is that it does not allow for much reflection or revision. Once an application is in the testing stage, it is very difficult to go back and change something that was not well-thought out in the concept stage.

2.5.2 V-Model

The V-model is a software development model which is considered as the extension of the waterfall model. Instead of moving down in a linear way, the process steps are bent upwards after the coding phase, to form the typical V shape. Figure 2.2 demonstrates the relationships between each phase of the development life cycle and its associated phase of testing.

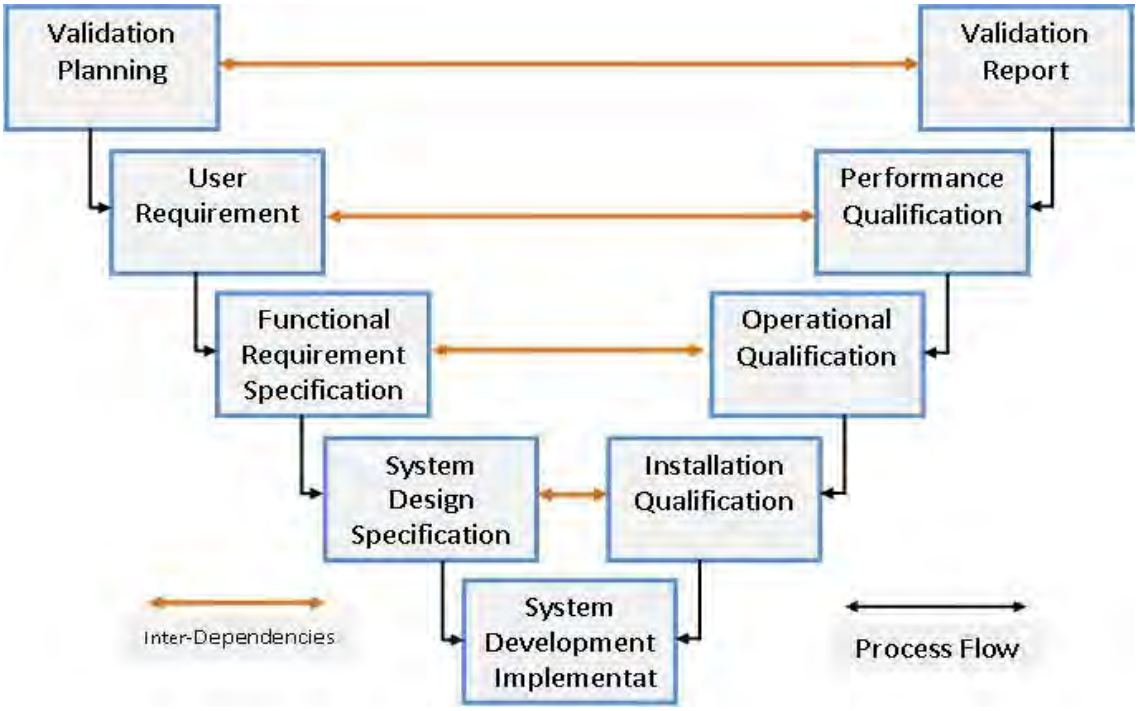


Figure 2.3: V-Model

2.5.2.1 Advantages of V-Model

The main advantages of V-Model are

i. Proactive defect tracking i.e defects are found at early stages even may be in the development phase before application is tested.

ii. It avoids the downward flow of the defect

iii. It reduces the cost for fixing the defect since defects will be found in early stages

iv. It is a fast method

2.5.2.2 Disadvantages of V-Model

It has the following disadvantages

i. It requires more people to work

ii. It needs an established process to implement.

2.5.3 Iterative Model

An iterative lifecycle model does not attempt to start with a full specification of requirements. Instead, development begins by specifying and implementing just part of the software, which can then be reviewed in order to identify further requirements. This process is then repeated, producing a new version of the software for each cycle of the model. Figure 2.3 shows an iterative lifecycle model which consists of repeating the following four phases in sequence:

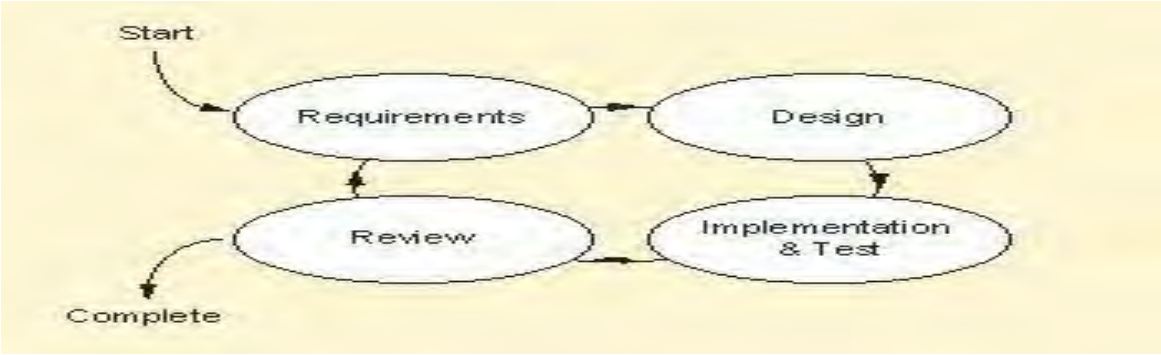


Figure 2.4: Iterative Model

In the requirements phase, requirements for the software are gathered and analyzed. Iteration should eventually result in a requirements phase that produces a complete and final specification of requirements. In the design phase, software solution to meet the requirements is designed. This may be a new design, or an extension of an earlier design. In the implementation and test phase, the software is coded, integrated and tested. In the review phase, the software is evaluated, the current requirements are reviewed, and changes and additions to requirements are proposed.

For each cycle of the model, a decision has to be made as to whether the software produced by the cycle will be discarded, or kept as a starting point for the next cycle.

2.5.4 Rapid Application Development (RAD) Model

RAD is a concept that products can be developed faster and of higher quality through

i. Gathering requirements using workshops or focus groups
ii. Its prototyping is very easy and user testing of design is re-iterative.
iii. The re-use of software components

Figure 2.5 shows the steps of the RAD model

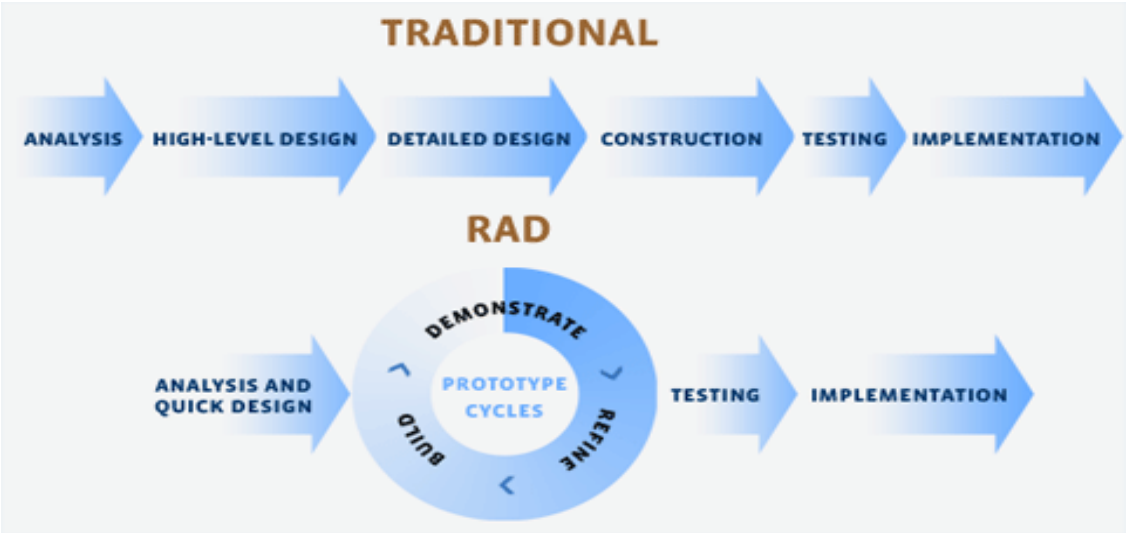


Figure 2.5: RAD Model

Some companies offer products that provide some or all of the tools for RAD software development. The concept can be applied to hardware development as well. These products include requirements gathering tools, prototyping tools, computer-aided software engineering tools, language development environments such as those for the Java platform, groupware for communication among development members, and testing tools. RAD usually embraces object oriented programming methodology, which inherently fosters software re-use. The most popular object-oriented programming languages, C++ and Java, are offered in visual programming packages.

2.5.4.1 Advantages and Disadvantages of RAD Model

RAD reduces the development time and reusability of components help to speed up the development. All functions are modularized. So, it is easy to work with. For large projects RAD requires highly skilled engineers in the team. Both end customers and developer should be committed to complete the system in a much abbreviated time frame. If commitment is lack, RAD will fail. RAD is based on Object Oriented approach and if it is difficult to modularize the project the RAD may not work well.

2.6 Database:

Database systems are design to manage large bodies of information. Management of data involves both defining structures for storage of information and proving mechanisms for the manipulation of information. In addition, the database system must ensure the safety of the information stored, despite system crashes attempts to unauthorized access. If data are to be shared among several users, the system must avoid possible anomalies result.

2.7 Definition of Database

A database management system (DBMS) is a collection of interrelated data and a set if program to access those data the collection of data, usually refer to as the database, containing information relevant to an enterprise. The primary goal of a database is to provide a way to store and retrieve database information that is both convenient and efficient.

2.8 Types of Database

There are two types of database:

1. Desktop Based
    - Standalone
    - Network Support
2. Web Based

2.9 Database system versus File system

The typical file system is supported by a conventional operating system. The system stores permanent record in various files and it needs different application programs to extract records from it and add records to the application file. Before database management system came along, organization usually stored information/data in such system.

Kipping data/information in file processing system has number of major disadvantages:

- Data redundancy and inconsistency
- Difficult in data access
- Data isolation
- Integrity problem
- Atomicity problem
- Concurrent access anomalies
- Security problem

2.10 Procedures of Database

There are various procedures for database Design. Some of the database procedures are shown below:

- Create Database User
- Create Database Tables
- Normalization
- Relationship
- Implement with Oracle
- Coding
- Debugging and Testing
- Database level Modification
- Finalization
- Implementation

2.11 Information Modeling

Whenever understanding of a real world phenomenon or interaction desired the first step is to build a simple model representing the phenomenon and perform study and

evaluation of the model rather that actual interaction. In building model the ambiguity and complexity in the real world is avoided totally or compensated with valid approximation. In fact a systematic study of any phenomenon whether economic, industrial or scientific is done through modeling. Information modeling pertains to development of model in information generation, storage, destruction, evaluation, manipulation, synthesis and utilizations. These models help in systematization of information generation, flow, interpretation synthesis of more information.

2.12 Concept of Information Modeling

The first step of information modeling is to precisely choose that part of reality, which is of interest. This is known as Entity. Entity defines other parameter. Entities are atomic that is they are invisible. The parameters if as entity are known as attributes. Attributes are known as quantum of information, which describe the entity entirely.The second step in information modeling is relationship among entities. It is most important in making sense if the entity and inter-entity relationship. The relationship is the information, which links two entities. The relationship can be of four categories:

- One-to-One (1:1) –One instance of the first entity can correspond to only one instance of the second entity. It is known as binary relationship.

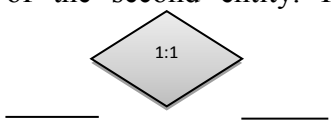


Figure 2.6: One-to-One Relation

- One-to-Many (1:M) – One instance of the first entity can correspond to more than one of the second entity.

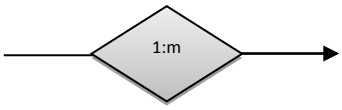


Figure 2.7: One-to-Many Relation

- Many-to-One (M:1) – More than one instance of the first entity can corresponds to the same one instance of the second entity

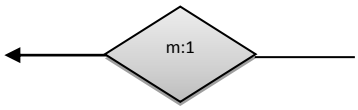


Figure 2.8: Many-to-One Relation

- Many-to-Many (M : M) – More than one instance of the first entity can correspond to more than one instance of the second entity.

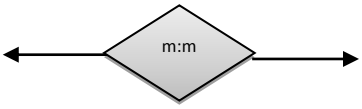


Figure 2.6: Many-to-Many Relation

2.13 Procedures for Information Modeling

The steps for Information Modeling and rules of data normalization were described below:

- The universe of discourse or the part if real world should be selected properly with minimum redundancy but completely in terms of entities.
- Entities are to be classified according to their attributes contents and merged, which have One-to-One (1:1) Relationship into composite entities.
- Their relationship among entities should be defined and named.
- Normalization should be done to reduce redundancy and increase integrity.

2.14 Data Modeling

A data model is a model describing the data in an organization. It provides frame work for abstracting the essential qualities or characteristics of data. Data modeling is the process of abstracting and documentation using a data model.

Data modeling creates hierarchies of abstraction along two dimensions: aggregation and generalization. Aggregation identifies data item as art's of higher-level, more

aggregate descriptor. Generalization creates categories into which a data item may be classified.

There are two major classes of data models – logical data models and physical data models. These two classes reflects the fact that efficient physical storage and retrieval of data must be Designed around the physical characteristics if storage media and devices, but user of data should be able to describe, think about and use data without being concentrated about its physical storage.

2.15 Types of data modeling

The data modeling consists of three interrelated pieces of information, the data object, the attributes that describe the data object and the relationship that connect data objects to one another. A data object is a representation of almost any composite information that must be understood by software. Composite information means something that has a number of different properties or attributes.
Six different types of data models are given below:

- Entity-Relationship Model (E-R Model)
- Relational Model
- Object-Oriented Data Model
- Object-Relational data Model
- Hierarchical Data Model
- Network Data Model

2.16 Summary

The software development lifecycle (SDLC) is not only a great way to ensure the software to meet the needs of business and customers, but it is also essential in supporting the software once it's published. SDLC is a process used by software industry to design, develop and test high quality software. The SDLC aims to produce high quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates. SDLC is a framework to guide the development to make it more systematic and efficient. Using SDLC we will be able to tell how long it will take to complete the project, to test and deploy. Not only that,

we'll have an easier time debugging and finding flaws in the software program or make enhancement to it. We have planned our development according to the standard SDLC.

We have gone through all the steps of SDLC and analyses as details as possible and made our development decisions.

Chapter 3

System Design

3.1 Introduction

The software system design describes the desired software features in detail, including database design (ER diagram), software design (UML is produced here), screen layouts and other documents. In system design, the software's overall structure is defined with a full data dictionary. These design elements are intended to describe the software in detail that helps to develop the software with minimal additional input.

3.2 Database Design

A database is a collection of information, organized in such a way that a computer program can quickly select desired pieces of data. Databases are designed to offer an organized mechanism for storing, managing and retrieving information. This includes detailed specification of data elements, data types, indexing options and other parameters residing in the DBMS data dictionary. Many models and languages are used for design of the database. To design the database the Entity-Relationship (ER) Diagram is used.

1. T_WORDER: The software presented in this report has the following table fields database. **Table 3.1** is regarding information about work order details of a client order. It contains a two composite primary key which are “BasicTypeID” and “OrderNo” and foreign key “ClinetID”, “SalesPersonID”. It’s also contain Client Name, Sales term, Client Order Ref, Salesperson Name etc.

2. T_WOITEMS: The software presented in this report has the following table fields database. **Table 3.2** is regarding information about order items of a client order. It contains a three composite primary key which are “BasicTypeID” , “OrderNo” and “WOITEMSL” and foreign key “YarnTypeID”, “YarnCountID”. It’s also contain YarnType, YarnCount, Shade, UnitPrice , Quantity etc.

3. T_DBATCH : The software presented in this report has the following table fields database. **Table 3.3** is regarding information about batches of orders items for yarn dyeing. It contains a primary key which are “DBatchID” and foreign key “BasicTypeID”, “OrderNo” and “DMachineID”. It’s also contain PlanDate, Dyeing Duration, Client Order Ref, Salesperson Name etc.

4. T_EMPLOYEE : The software presented in this report has the following table fields database. **Table 3.4** is regarding information about Employee. It contains a

primary key which is “EmployeeID” and foreign key “InopertionID” It’s also contain Employee, Designation, Emppwd, Empmanager, EmpgroupID etc.

5. T_CLIENT : The software presented in this report has the following table fields database. **Table 3.5** is regarding information about a client. It contains a primary key which is “ClientID” and foreign key “InopertionID”. It’s also contain Client Name, Caddpress, Cfactory,Ctelephone,CFax, CEmail,Ctelex etc.

6. T_YSTOCK : The software presented in this report has the following table fields database. **Table 3.6** is regarding information about yarn stock information. It contains a primary key which is “YernstockID” and foreign key “BasictypeID”, “StockTypeID”, “CurrencyID”,WobasictypeID . It’s also contain StockDate, ChallanNo, LCNO, SupName etc.

.**7. T_YSTOCKSUB :** The software presented in this report has the following table fields database. **Table 3.7** is regarding information about yarn stock information details. It contains a primary key which is “YStocksubID” and foreign key “YarnStockID”, “ClientID”, “ CountID”, “YarnTypeID” , “ BasictypeID”, “DBatchID”. It’s also contain Shade, StockQtd, ItemChallan, WoNumber, LotNo etc.

8. T_YARNTRANSACTIONTYPE : The software presented in this report has the following table fields database. **Table 3.8** is regarding information about Stock Type. It contains a primary key which is “StockTypeID”. It’s also contain StockType.

9. T_YARNTRANSACTIONTYPEDETAILS : The software presented in this report has the following table fields database. **Table 3.9** is regarding information about Stock Type details. It contains a composite primary key which are “StockTypeID” , “StockSubTypeID”. It’s also contain StockSubType, GYMS, GYMSW, GYFS, DYS, DYSW etc.

10. T_UNITOFMEAS : The software presented in this report has the following table fields database. **Table 3.10** is regarding information unit name. It contains a primary key which is “UnitofmeasID”. It’s also containing Unitofmeas.

3.3 Data Dictionary

Data Dictionary describes the table format which is used in database design. The following tables are used in database design.

Table 3.1: T_WORDER

PK	Name	Data Type	Size	Not Null	Default	Comment
	BASICTYPEID	VARCHAR2	3	☑		
	WONUMBER	NUMBER	6	☑		
	WODATE	DATE		☑		
	CLIENTID	NUMBER	4	☑		
	SALESTERMID	NUMBER	1	☑		
	CURRENCYID	NUMBER	3	☑		
	CONRATE	NUMBER	12	☑		
	DELIVERYPLACE	VARCHAR2	200	☐		
	WSTATUSID	NUMBER	1	☑		
	CLIENTSREF	VARCHAR2	100	☐		
	EMPLOYEEID	VARCHAR2	8	☑		
	WCANCELLED	NUMBER	1	☑	0	
	WREVISED	NUMBER	1	☑	0	
	WOCOMMENTS	VARCHAR2	200	☐		
	BUYERID	NUMBER	4	☑		
	PIORLCREF	VARCHAR2	200	☐		
	REVISEDCOUNT	NUMBER	3	☐		
	REVISEDDATE	DATE		☐		
	DWONUMBER	NUMBER	7	☐		
	MAXREVISEDCO...	NUMBER	3	☐		
	LCREF	VARCHAR2	200	☐		

The T_WORDER table presented in this report has the above table field's database.

Table 3.2: T_WOITEMS

PK	Name	Data Type	Size	Not Null	Default	Comment
	BASICTYPEID	VARCHAR2	3	☑		
	WONUMBER	NUMBER	6	☑		
	WOITEMSL	NUMBER	3	☑		
	TICKET	VARCHAR2	10	☐		
	SHADE	VARCHAR2	40	☑		
	COUNTID	NUMBER	3	☑		
	YARNTYPEID	NUMBER	3	☑		
	LABDIPID	VARCHAR2	30	☐		
	WOITEMSQTY	NUMBER	12	☑		
	WOITEMSUNITP...	NUMBER	12	☐		
	WOITEMSDDATE	DATE		☑		
	WOITEMSCOMM...	VARCHAR2	200	☐		
	LABDIPNO	VARCHAR2	100	☐		
	APPROVED	VARCHAR2	1	☐		
	QTYLBS	NUMBER	12	☐		
	SAMPLEORDERREF	VARCHAR2	9	☐		
	SAMPLEBATCHREF	VARCHAR2	50	☐		
	UNITPRICELBS	NUMBER	12	☐		
	ADDYARNPERC	NUMBER	12	☐	0	
	CLIENTSREF	VARCHAR2	150	☐		
	PROCESSLOSS	NUMBER	12	☐		

The T_WOITEMS table presented in this report has the above table field's database.

Table 3.3: T_DBATCH

PK	Name	Data Type	Size	Not Null	Default	Comment
	DBATCHID	NUMBER	20	☑		
	BASICTYPEID	VARCHAR2	3	☐		
	WONUMBER	NUMBER	6	☐		
	WOITEMSL	NUMBER	3	☐		
	BATCHNO	NUMBER	6	☑		
	DMACHINENO	NUMBER	6	☑		
	PLANDATE	DATE		☑		
	BDURATION	NUMBER	6	☑		
	BATCHQTY	NUMBER	12	☐		
	FINISHEDQTY	NUMBER	12	☐		
	BATCHSTATUS	NUMBER	3	☐	0	
	BATCHRESTATUS	NUMBER	3	☐	0	
	BATCHPARENT	NUMBER	20	☐		
	BATCHTYPE	NUMBER	3	☐	0	
	BSTATIONARY	NUMBER	1	☐	0	
	DESCRIPTION	VARCHAR2	300	☐		
	YARNLOTNO	VARCHAR2	100	☐		
	BATCHAPPSTATUS	NUMBER	2	☐	0	
	HASDYELINE	NUMBER	1	☐	0	
	BATCHMPARENTID	NUMBER	20	☐		
	CBATCHTEXT	VARCHAR2	200	☐		
	CMERGETEXT	VARCHAR2	1000	☐		
	MCHILDBATCHIDS	VARCHAR2	200	☐		
	BATCHMSTATUS	NUMBER	2	☐	0	

The T_DBATCH table presented in this report has the above table field's database.

Table 3.4: T_EMPLOYEE

PK	Name	Data Type	Size	Not Null	Default	Comment
	EMPLOYEEID	VARCHAR2	8	☑		
	EMPLOYEE	VARCHAR2	100	☑		
	DESIGNATION	VARCHAR2	100	☑		
	INOPERATIONID	NUMBER	1	☑		
	EMPPWD	VARCHAR2	8	☑		
	EMPMANAGER	VARCHAR2	8	☑		
	EMPGROUPID	NUMBER	2	☑		

The T_EMPLOYEE table presented in this report has the above table field's database.

Table 3.5: T_CLIENT

PK	Name	Data Type	Size	Not Null	Default	Comment
	CLIENTID	NUMBER	4	☑		
	CLIENTNAME	VARCHAR2	75	☑		
	INOPERATIONID	NUMBER	1	☑		
	CADDRESS	VARCHAR2	250	☐		
	CFACTORY	VARCHAR2	200	☐		
	CTELEPHONE	VARCHAR2	100	☐		
	CFAX	VARCHAR2	50	☐		
	CEMAIL	VARCHAR2	50	☐		
	CTELEX	VARCHAR2	20	☐		
	CURL	VARCHAR2	50	☐		
	CPERSON	VARCHAR2	100	☐		
	BONLYFACTORY	NUMBER	1	☑	0	
	REMARKS	VARCHAR2	50	☐		
	ACCCODE	VARCHAR2	20	☐		

The T_CLIENT table presented in this report has the above table field's database.

Table 3.6: T_YSTOCK

PK	Name	Data Type	Size	Not Null	Default	Comment
	YARNSTOCKID	NUMBER	20	☑		
	BASICTYPEID	VARCHAR2	3	☑		
	STOCKDATE	DATE		☑		
	STOCKTYPEID	NUMBER	3	☑		
	STOCKSUBTYP...	NUMBER	3	☐		
	CLIENTID	NUMBER	6	☐		
	CHALLANNO	VARCHAR2	30	☐		
	STOCKCOMME...	VARCHAR2	100	☐		
	LCNO	VARCHAR2	200	☐		
	SUPINVOICENO	VARCHAR2	100	☐		
	SUPINVOICED...	DATE		☐		
	SUPNAME	VARCHAR2	200	☐		
	BPRICEUPDATED	NUMBER	1	☐	0	
	CURRENCYID	NUMBER	3	☐		
	CONRATE	NUMBER	12	☐		
	SUPPLIERID	NUMBER	15	☐		
	BACCEPTED	NUMBER	1	☐		
	WONUMBER	NUMBER	6	☐		
	WOBASICTYPEID	VARCHAR2	3	☐		

The T_YSTOCK table presented in this report has the above table field's database.

Table 3.7: T_YARNTRANSACTIONTYPE

PK	Name	Data Type	Size	Not Null	Default	Comment
	STOCKTYPEID	NUMBER	3	☑		
	STOCKTYPE	VARCHAR2	100	☑		

The T_YARNTRANSACTIONTYPE table presented in this report has the above table field's database.

Table 3.8: T_YSTOCKSUB

PK	Name	Data Type	Size	Not Null	Default	Comment
	YSTOCKSUBID	NUMBER	20	☑		
	YARNSTOCKID	NUMBER	20	☑		
	CLIENTID	NUMBER	6	☐		
	COUNTID	NUMBER	3	☑		
	YARNTYPEID	NUMBER	3	☑		
	SHADE	VARCHAR2	50	☐		
	STOCKQTY	NUMBER	20	☑		
	ITEMCHALLAN	VARCHAR2	30	☐		
	BASICTYPEID	VARCHAR2	3	☐		
	WONUMBER	NUMBER	6	☐		
	LOTNO	VARCHAR2	50	☑		
	FGTNNO	NUMBER	6	☐		
	DBATCHID	NUMBER	20	☐		
	REMARKS	VARCHAR2	100	☐		
	DBATCHNO	VARCHAR2	50	☐		
	WOITEMSL	NUMBER	3	☐		
	YARNPRICE	NUMBER	12	☐		
	BATCHBASICTYP...	VARCHAR2	3	☐		
	BATCHWONUMBER	NUMBER	6	☐		
	BATCHWOITEMSL	NUMBER	3	☐		
	ISSUETO	VARCHAR2	100	☐		
	REQCHALLANNO	VARCHAR2	30	☐		
	REQQTY	NUMBER	12	☐		
	REQREMQTY	NUMBER	12	☐		
	CURRENTSTOCK	NUMBER	12	☐		

The T_YSTOCKSUB table presented in this report has the above table field's database.

Table 3.9: T_YARNTRANSACTIONTYPEDETAILS

PK	Name	Data Type	Size	Not Null	Default	Comment
	STOCKTYPEID	NUMBER	3	☑		
	STOCKSUBTYP...	NUMBER	3	☑		
	STOCKSUBTYPE	VARCHAR2	100	☑		
	GYMS	NUMBER	3	☐	0	
	GYMSW	NUMBER	3	☐	0	
	GYFS	NUMBER	3	☐	0	
	DYS	NUMBER	3	☐	0	
	DYSW	NUMBER	3	☐	0	

The T_YARNTRANSACTIONTYPEDETAILS table presented in this report has the above table field's database.

Table 3.10: T_UNITOFMEAS

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	UNITOFMEASID	NUMBER(3,0)	No	(null)	1	(null)
2	UNITOFMEAS	VARCHAR2(20 BYTE)	No	(null)	2	(null)

The T_UNITOFMEAS table presented in this report has the above table field's database.

3.4 Entity diagram

3.4.1 Attributes of Entity T_WORDER

Figure 3.4.1 shows the entire Attributes of Entity **T_WORDER**. Admin/permitted users can add **T_WORDER** information in this table and after successful registration of a work order user can also modify and delete the work order. Multiple column is used as primary key. Attribute WONUMMBER and BASICTYPEID can uniquely identify an row of **T_WORDER** table.

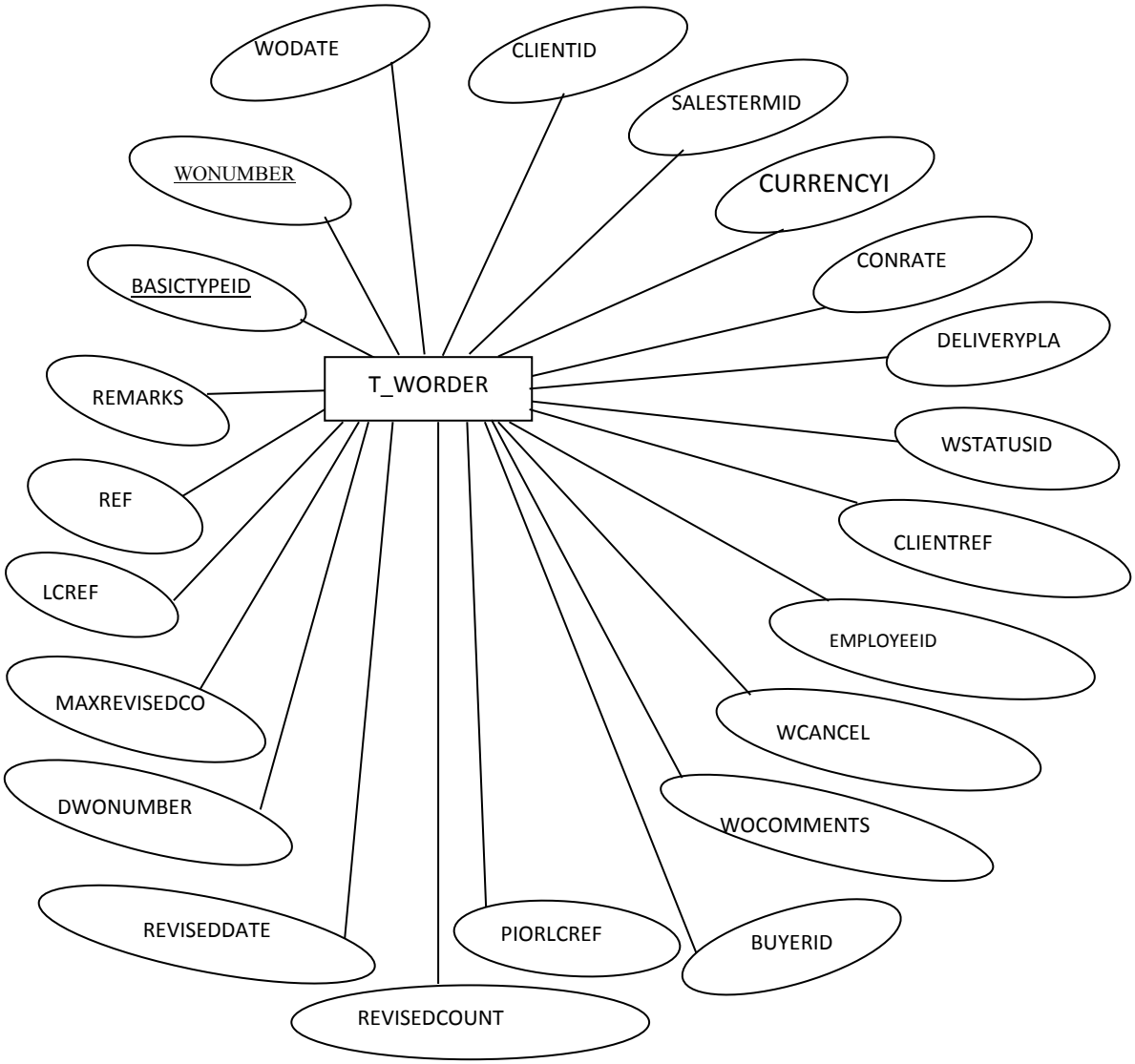


Figure 3.4.1: Attributes of Entity T_WORDER.

3.4.2 Attributes of Entity T_WOITEMS

Figure 3.4.2 shows the entire Attributes of Entity T_WOITEMS. Authorized users can add/edit/delete information in this table. This table holds the order items of all orders. **T_WOITEMS** is related with **T_WORDER** table in one to many relationship**. T_WOITEMS** is the many tables here for an individual work order the order items of that work order is saved in this table. WOITEMSL, WONUMBER, BASICTYPEID columns uniquely identifies a row of **T_WOITEMS** table. WONUMBER, BASICTYPEID is the primary key of one table.

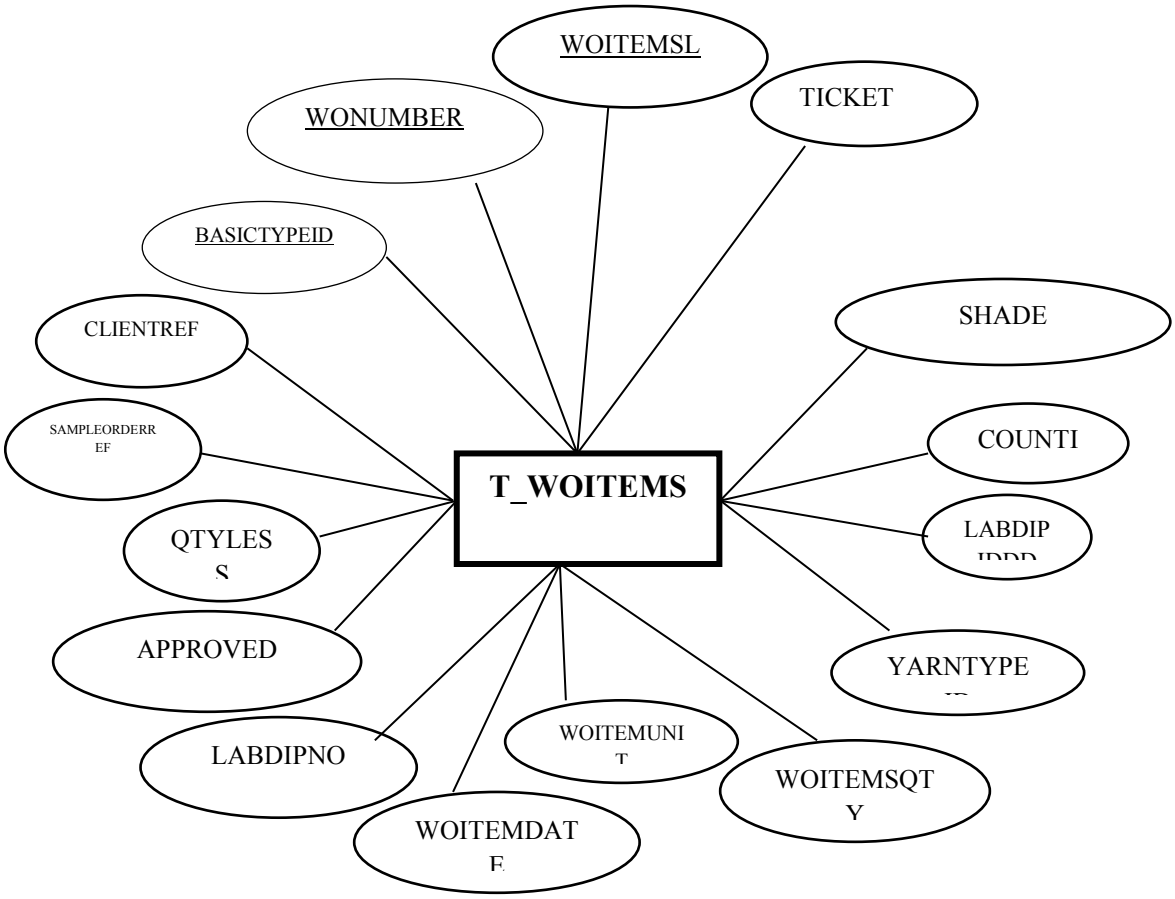


Figure 3.4.2: Attributes of Entity T_WOITEMS.

3.4.3 Attributes of Entity T_DBATCH

Figure 3.4.3 shows the Attributes of Entity **T_DBATCH**. Authorized user can users can add / delete /update this table. It holds the information of a batch of production plan. A batch holds a work order and the time it requires to be completed, machine number, batch type etc. DBatchID is the primary key here.

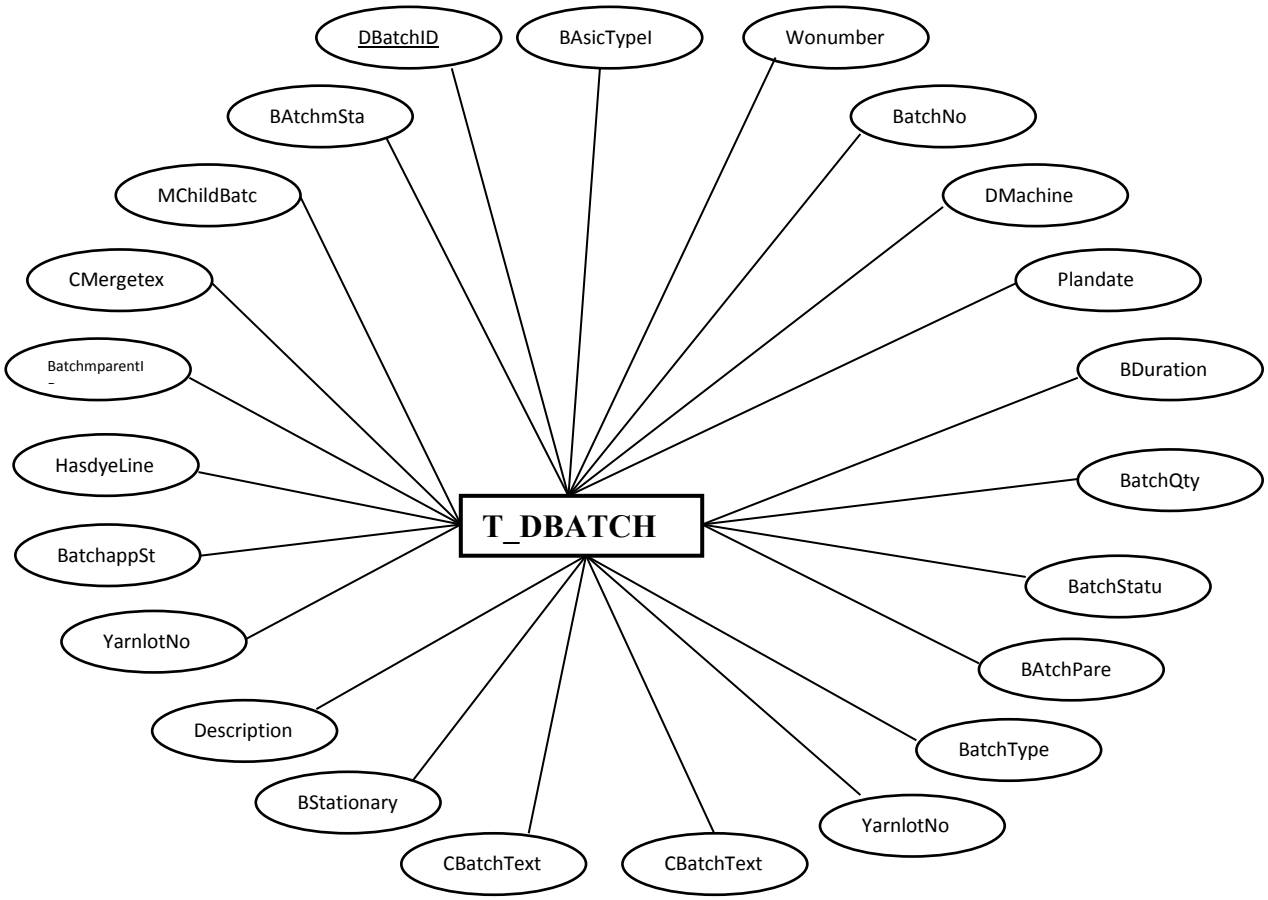


Figure 3.4.3: Attributes of Entity T_DBATCH.

3.4.4 Attributes of Entity T_EMPLOYEE

Figure 3.4.4 shows the Attributes of Entity **T_EMPLOYEE**. Authorized users can add the profile of an employee. EmployeeID is the primary key.

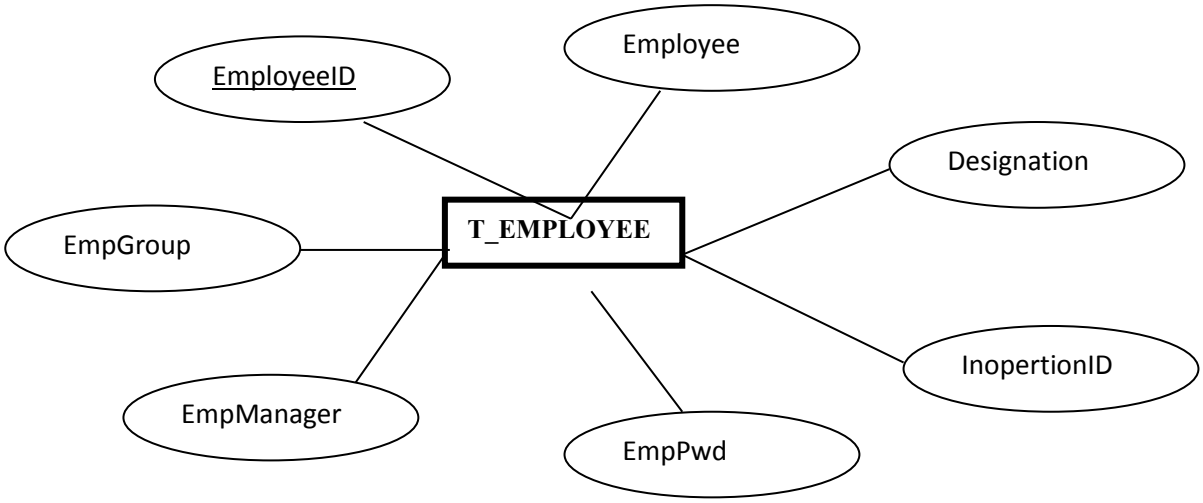


Figure 3.4.4: Attributes of Entity T_EMPLOYEE.

3.4.5 Attributes of Entity T_CLIENT

Figure 3.4.5 shows the Attributes of Entity **T_CLIENT**. The Client information is stored here. ClientID is the primary key here to identify an individual client.

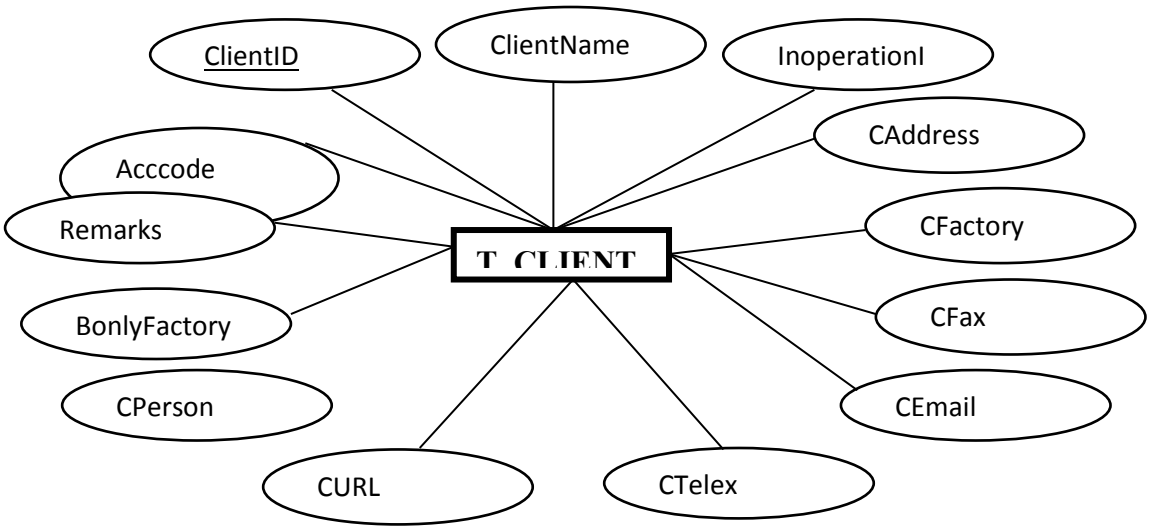


Figure 3.4.5: Attributes of Entity T_CLIENT

3.4.6 Attributes of Entity T_YSTOCK

Figure 3.4.6 shows the Attributes of Entity **T_YSTOCK**. All type of users is managed from this single table.

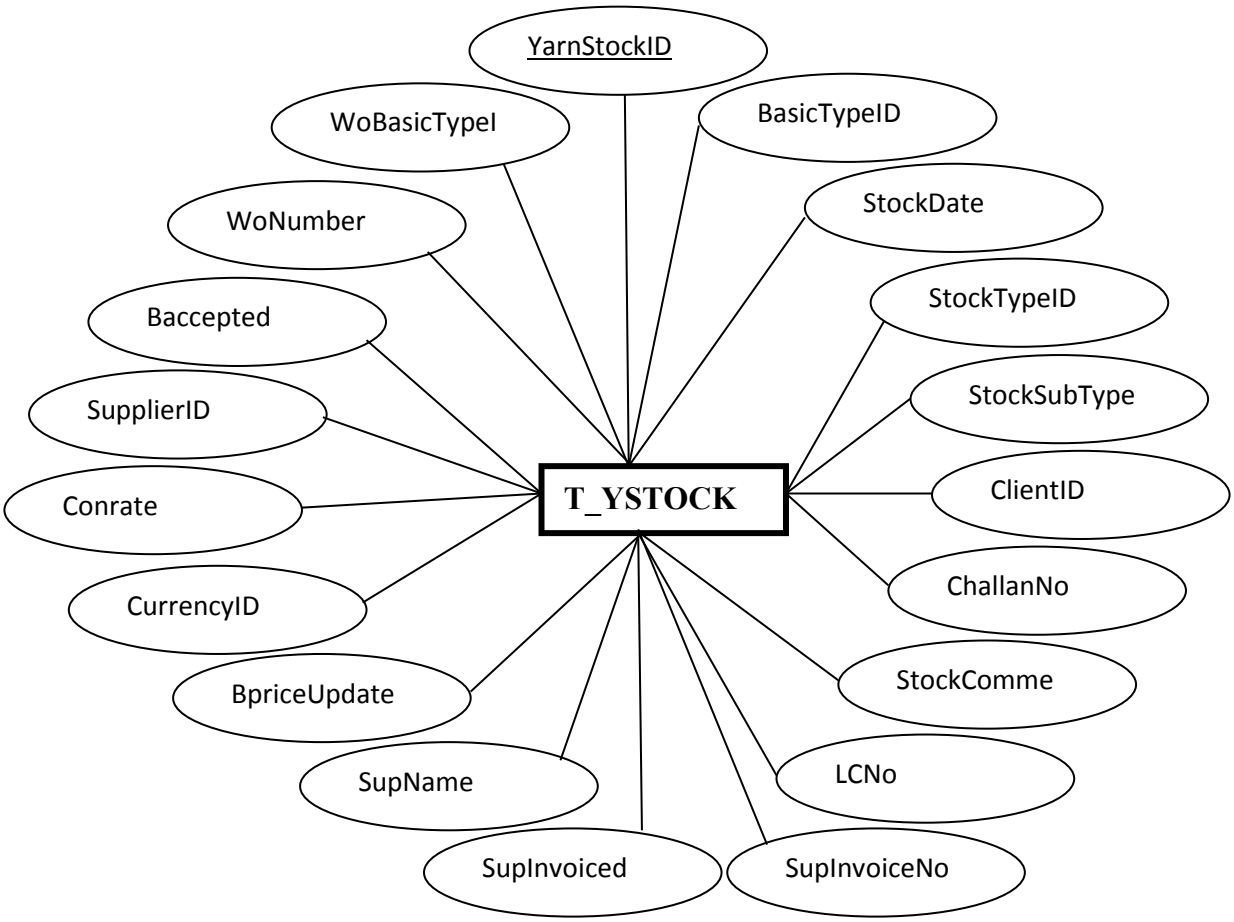


Figure 3.4.6: Attributes of Entity T_YSTOCK

3.4.7 Attributes of Entity T_YSTOCKSUB

Figure 3.4.7 shows the Attributes of Entity **T_YSTOCKSUB**.YStockSubID is the primary key of **T_YSTOCKSUB** table. This table shows the yarn stock , client , challan number etc.

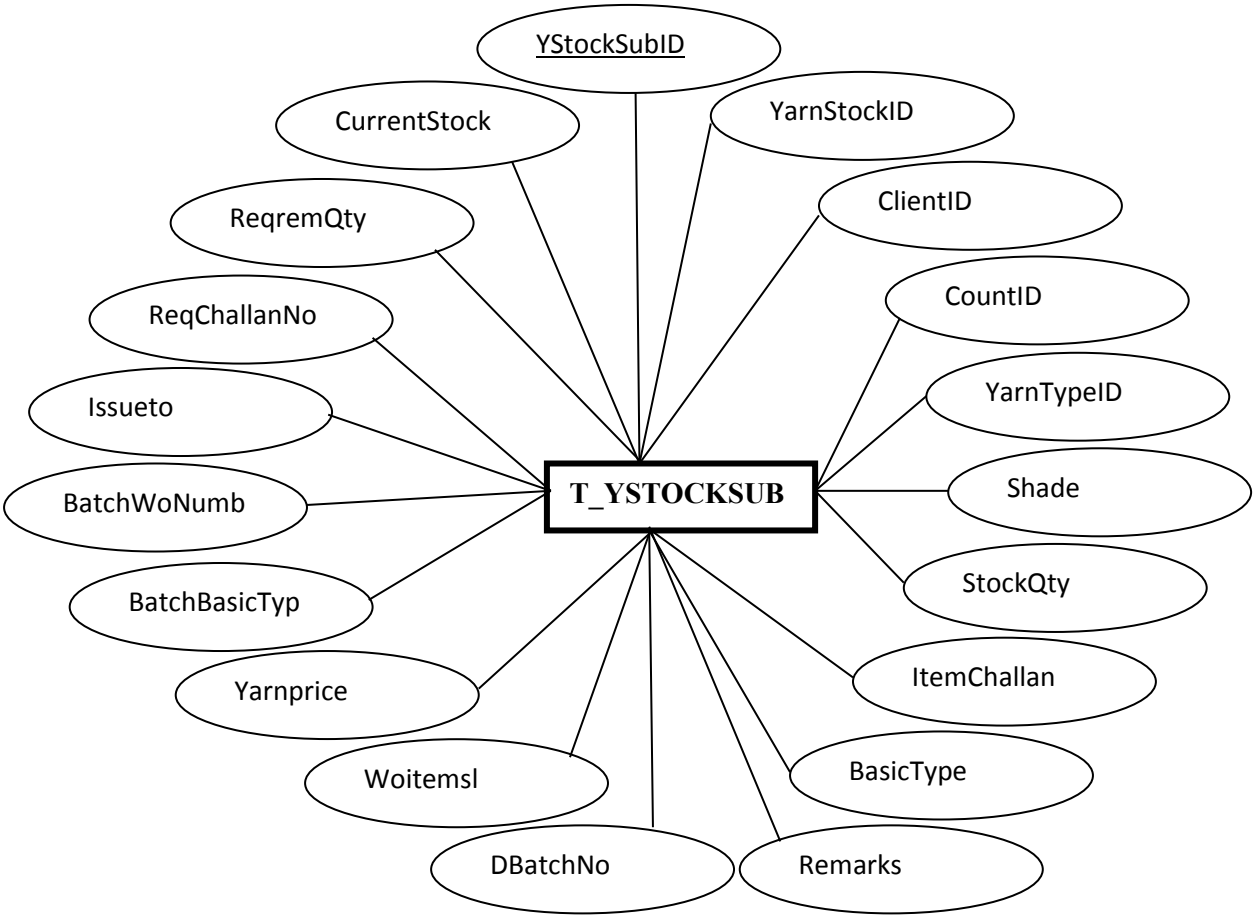


Figure 3.4.7: Attributes of Entity T_YSTOCKSUB

3.4.8 Attributes of Entity T_YARNTRANSACTIONTYPE

Figure 3.4.8 shows the Attributes of Entity **T_YARNTRANSACTIONTYPE**. StockTypeID is the primary key of the following table.

Figure 3.4.8: Attributes of Entity

3.4.9 Attributes of Entity: T_YARNTRANSACTIONTYPEDETAILS

Fig 3.4.9 shows the Attributes of Entity T_YARNTRANSACTIONTYPEDETAILS. StockTypeID is the primary key of the following table.

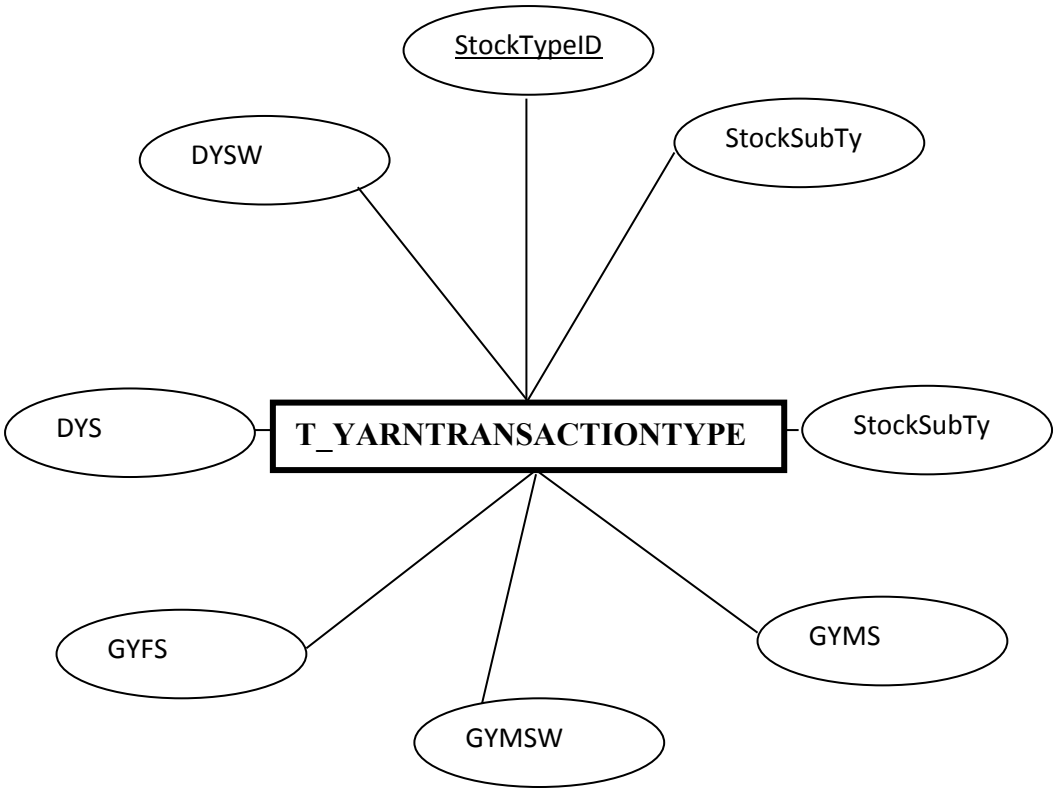


Figure 3.4.9: Attributes of Entity

3.4.10 Attributes of Entity: T_UNITOFMEAS

Figure 3.4.10 shows the Attributes of Entity: T_UNITOFMEAS. UnitofmeasID is the primary key of the following table.

Figure 3.4.10 Attributes of Entity T_UNITOFMEAS

3.5 E-R Diagram

3.5.1 E-R diagram:

An entity-relationship (E-R) diagram is a graphical representation of entities and their relationships to each other, typically used in computing in regard to the organization of data within databases or information systems.

Entity relationship diagram (E-R Diagram) can be translated into broad diversity of technical architecture data, such as relational, network, and hierarchical. An E-R data model evolves from project identification and selection throughout analysis as it becomes more precise and is validated by more detailed analysis of system needs. Several elements are to be included in order to draw the E-R diagram, such as entities, attributes, primary and foreign keys and identifiers. There are three basic elements in E-R diagram:

- Entities (tables) are the elements about which one seek information. Boxes are commonly used to represent entities.
- Attributes are the data one collect about the entities. Ovals are used to represent attributes.
- Relationships provide the structure needed to draw information from multiple entities. Diamonds are normally used to represent relationships.

Programming language is very much important because it helps to Design interface and run the application smoothly. The server and the language that used in this project are described below:

Programming language

- C#.NET
- Oracle 10.2g Database
- Crystal Report

The interface is designed with the help of C#.NET as it provides easy abstract window toolkit and used widely. C#.NET is an extremely rich programming language and it contains the basic components of developing user's interface. It is an integrated development environment in which we can develop, run, test and debug our application. The programmer can easily use the buildup tool kits for any

application software. It has evolved into a major development environment that covers every aspects of programming, from education applications to database and from financial applications to Internet components.

Figure 3.5.1 shows entire an E-R diagram.
Entity-Relationship Diagram of production planning Software (PPS) is shown to next page:

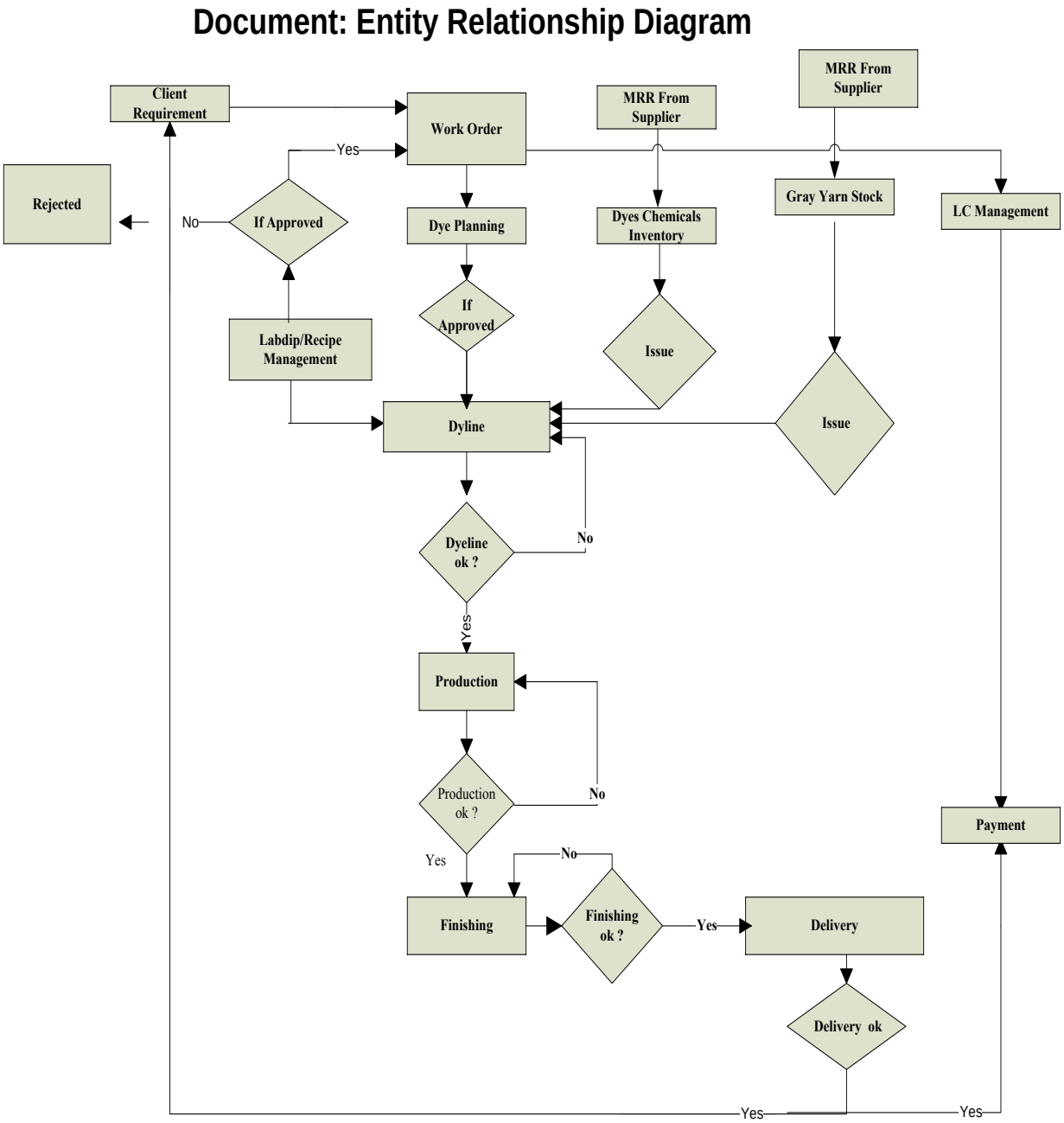


Figure 3.5.1: E-R Diagram

3.5.2 Database Schema Diagram

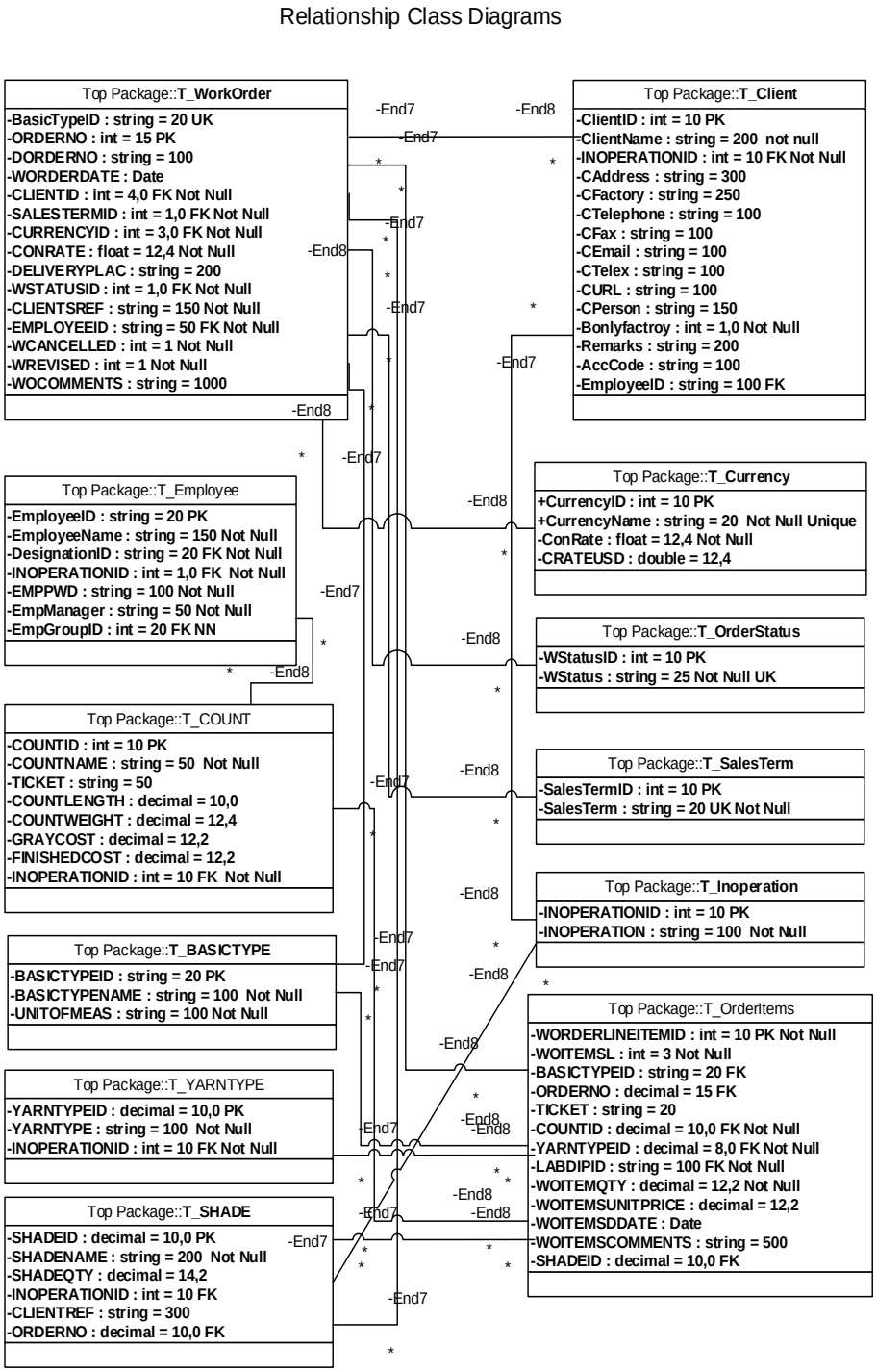


Figure 3.5.2: Schema Diagram

3.6 Data Flow Diagram

The Data Flow Diagram (DFD) is a graphical representation of the flow of data through an information system. Data flow diagrams present the logical flow of information through a system in graphical or pictorial form. Data flow diagrams (DFDs) have only for symbols, which makes useful for communication between analysts and users. Data flow diagrams (DFDs) show the data used and provided by processes within a system. The following DFD shows all the processes that comprise the overall system, show how information moves from one process to other.

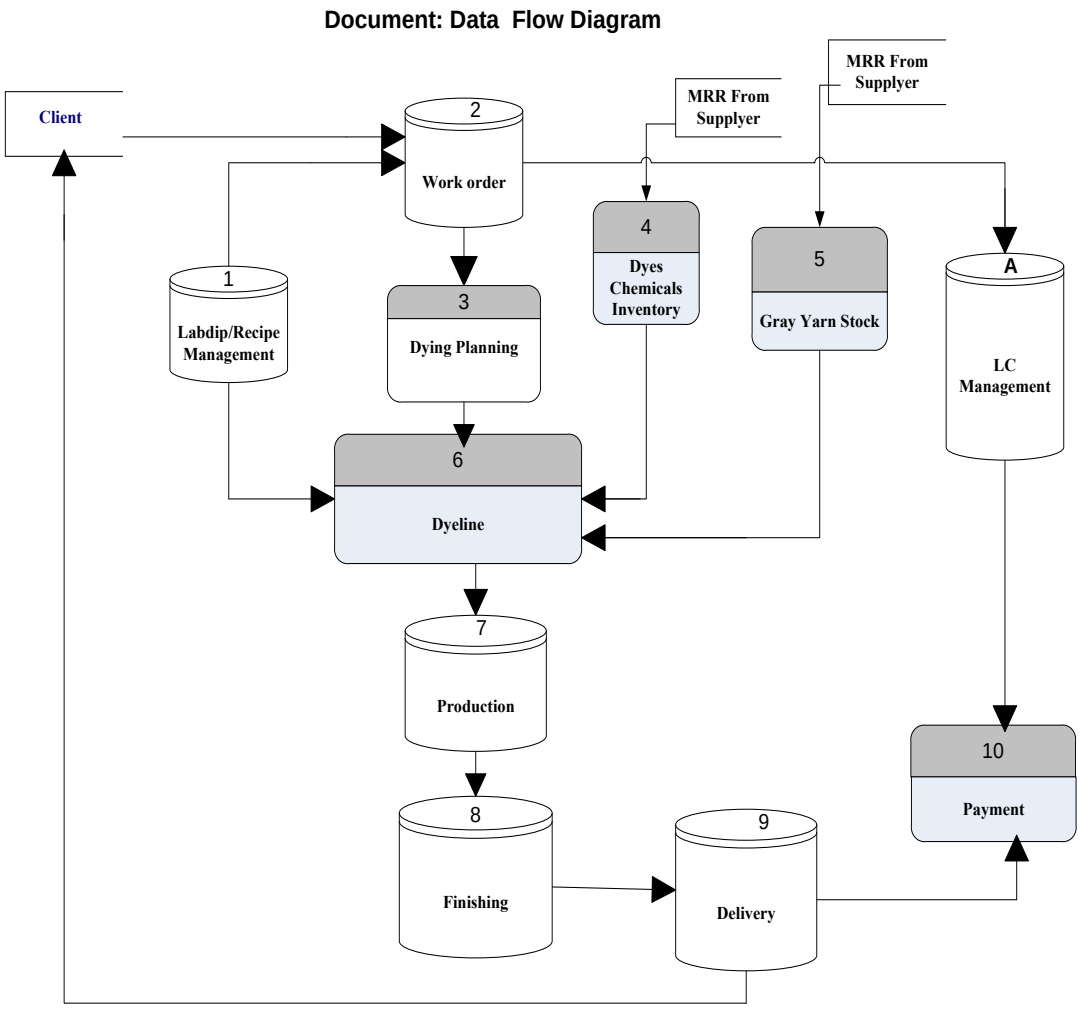


Figure 3.6: Data Flow Diagram

3.7 System Flowchart

System flowchart describes the data flow for a data processing system. It provides a logical diagram of how the system operates. It presents the flow of documents, the operations performed in data processing system. It also reflects the relationship between inputs, processing and outputs, illustrate the system flowchart of the project.

The flowchart of the production planning software (PPS) is given next page:

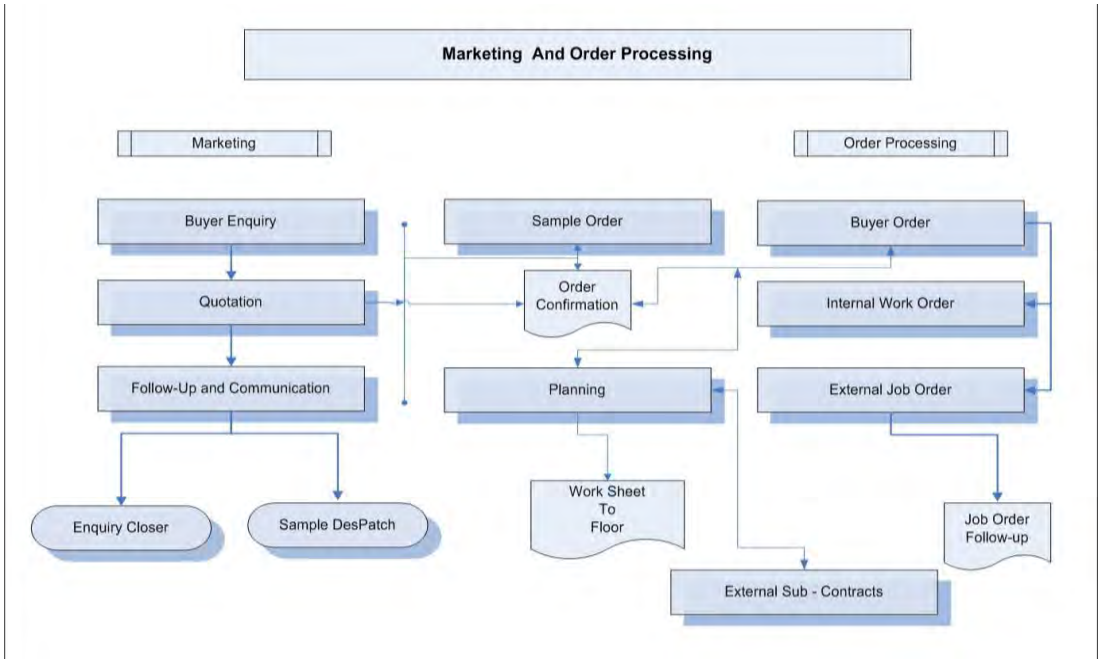


Figure 3.7: Flow Chart of PPS

3.8 Software Design

Software design is a process of problem-solving and planning for a software solution. Model-View and Controller analysis and design (MVC) are implemented during the software design. Each controller controls the flow of data and control via model to views. Views are basically controls the graphical user interface which works via the controller and models are handling the full database actions. Data are passing in view through models. There are a number of different notations for representing these flows, such as the Unified Modeling Language (UML). Different diagrams are used to help visualizing the whole development process.

3.9 UML Diagram

The Unified Modeling Language (UML) is graphical notation system for representation the analysis and design. UML is the industry-standard language for the specification, visualization, construction, and documentation of the components of software systems. UML helps to simplify the process of software design, making a model for construction with a number of different views. One of the great merits of UML is the way it helps open up the development process which is called use cases. These serve to identify principal roles (actors) in the system, boundaries, actions, and so on. UML Use Case Diagrams can be used to describe the functions of a system in a horizontal way.

3.10 Summery

The design of a database is crucial so that the database can function accurately or precisely. A good design of a database facilitates data managements and valuable information generator. A poor designed database tends to have errors, data redundancy, inaccuracy, can also be time consuming etc. proper or good design must be taken into consideration so to avoid redundancy, in accuracy, errors etc. as users would like to have a more efficient and reliable database that performs to the best or as expected.

For this project we create a database which contents 10 tables. All tables are well designed and redundant free. We use strong entity set for most of the tables. Developing a database without an ERD is as building a house without a building plan. It might be doable because you think that simply laying a brick one over another is enough to build something.

In this project we build a E-R diagram which contents details entity, entity set, relationship between entities and key constraints so that the database become easier and meaningful to everybody.

Chapter 4

System features

4.0 Introduction

This chapter designates the important functionalities and reports of the proposed production planning system software.

Functionality: The actions (operations), capabilities and usefulness of something such as a software application.

Reports: A report or account is any informational work (usually of writing, speech, television, or film) made with the specific intention of relaying information or recounting certain events in a widely presentable form. Reports are often used to display the result of an experiment, investigation, or inquiry. Reports may refer to specific periods, events, occurrences, or subjects, and may be communicated or presented in oral or written form. Written reports are documents which present focused, salient content to a specific audience.

The developed production planning system software has different essential features codes. Screen shot of some of the main features and reports are explained below.

4.1 User Login

The figure 4.20 is login form for enter the software. The user/admin/staff has to enter his username and password in the respective text box and as to click the "login" button to enter this system.

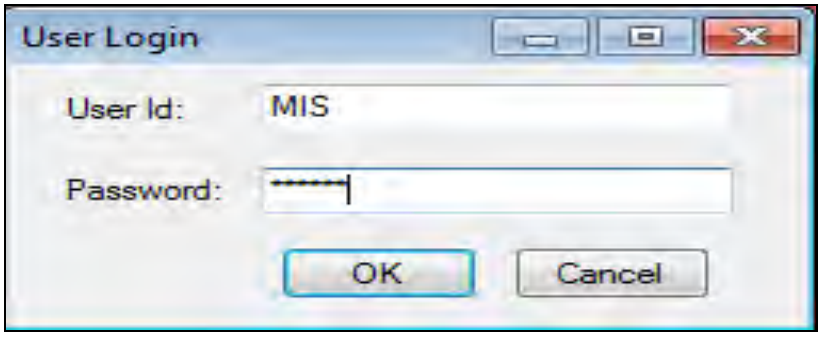


Figure 4.1: User Login (Production Planning)

4.2 Production Planning Entry Form:

The whole production is showed in one chart the machines and the time frame is presented as a graph. This is a chart in which a series of horizontal lines shows the amount of work done or production completed in certain periods of time in relation to the amount planned

for those periods. We select a machine and use the time frame to allocate it for certain production for a certain time. Production plan could be approved, not approved and reject.

The status of an order could be understood by color like

Green Color signifies Completed Work order

Yellow Color signifies running Work order

If we right click on a running work order we can observe it's status and details like which machine it is assigned, batch no ,duration, order no , status , production planning system for next 30 days machine availability etc.

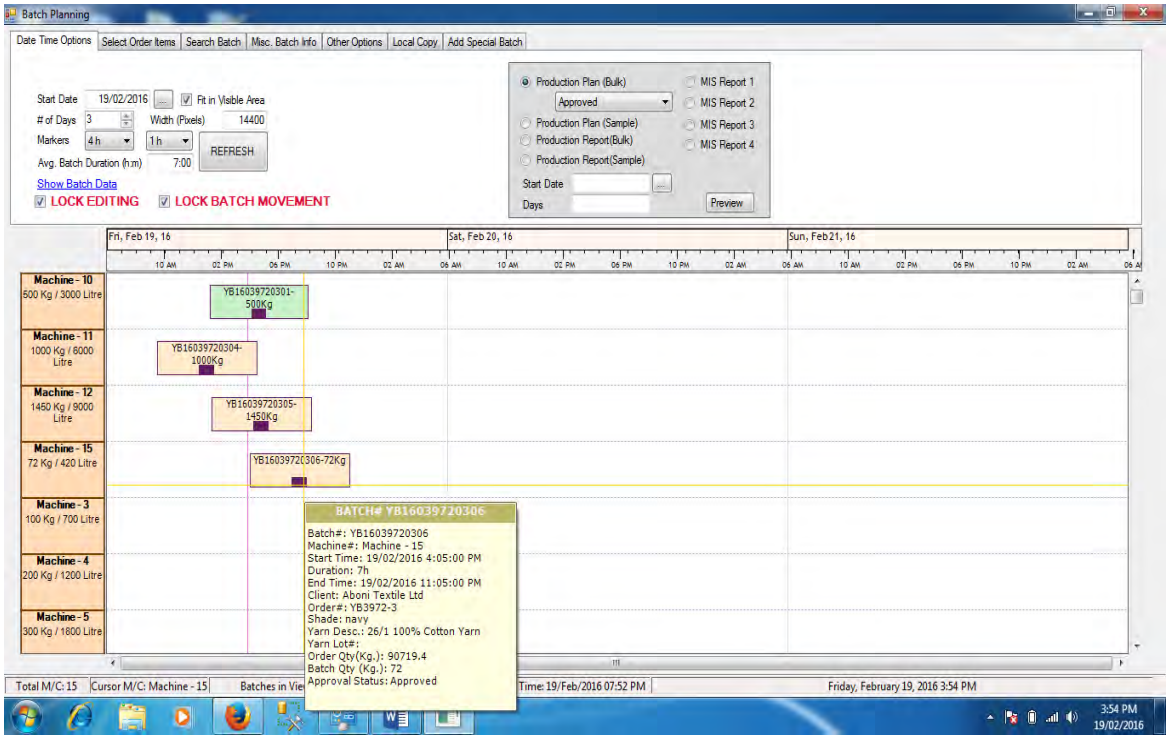


Figure 4.2: Production Planning Entry Form

4.3 Production Planning Entry (Select Order Items):

Features –

1. Browse all data at a time.
2. Drag & drop any field for Group view.
3. Print out of users' custom view.

We can select a work order and its order line items will be displayed in the grid then we can drag the order into the time slot for a following machine.

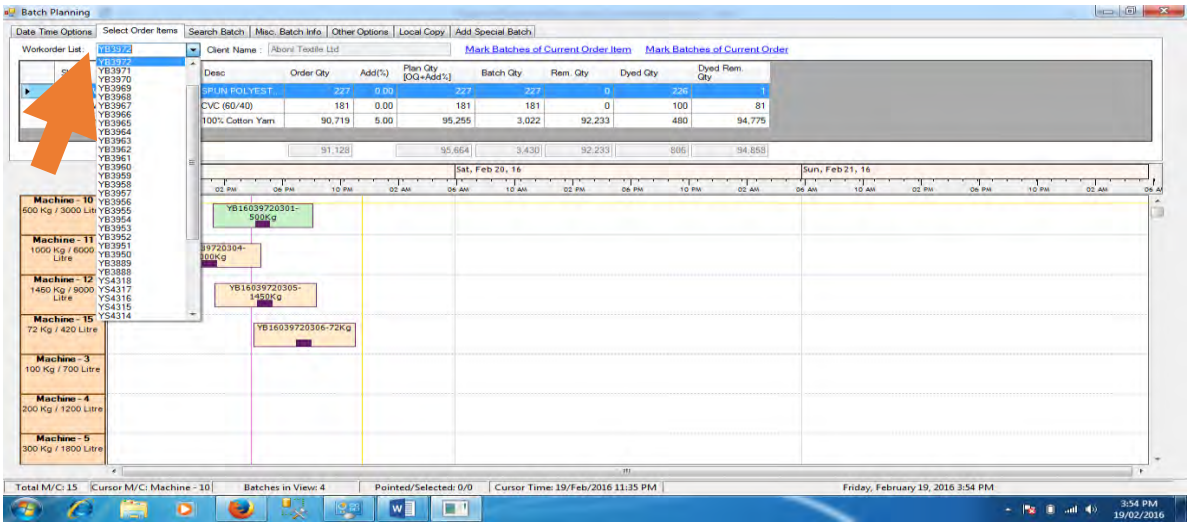


Figure 4.3: Production Planning Entry (Select Order Items)

4.4 Production Planning (Batching From Order Line Items):

We can also batch the work order of the grid by orderline items.

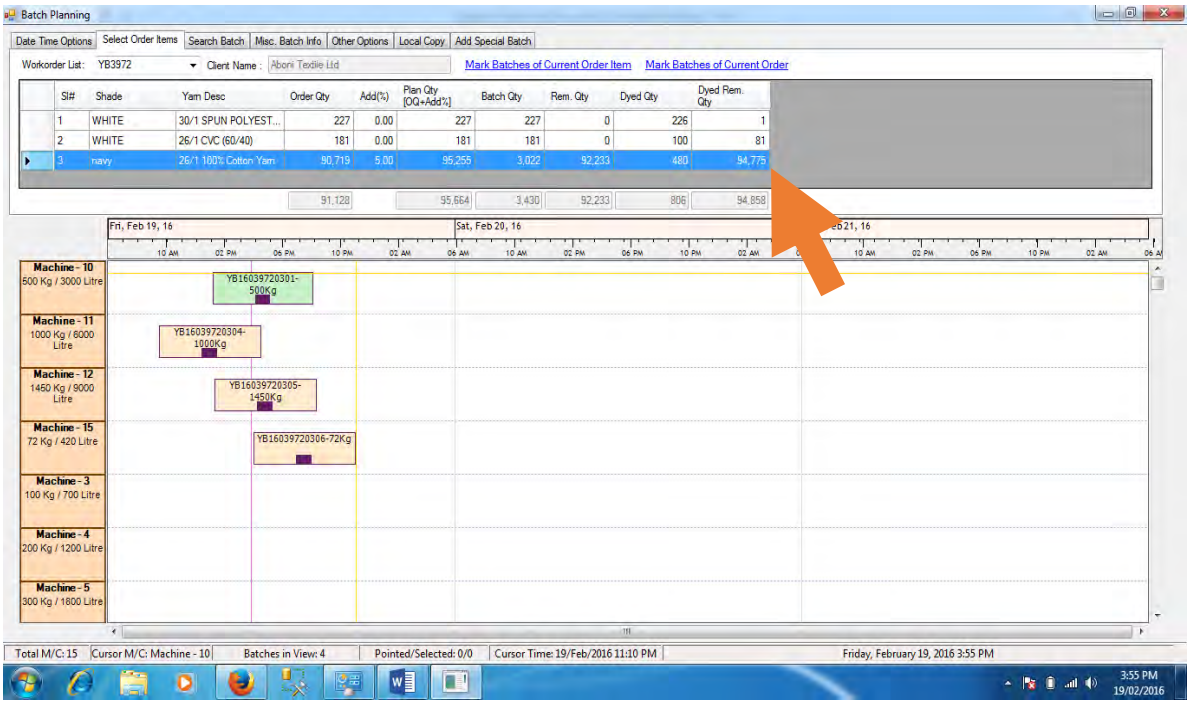


Figure 4.4: Production Planning (Batching From Order Line Items)

4.5 Actual Production Planning entry form with finished quantity

This is an actual production entry form with finished quantity. We can also approve or not approve or reject it .By clicking on a work order in Gantt chart we can modify or construct a batch planning.

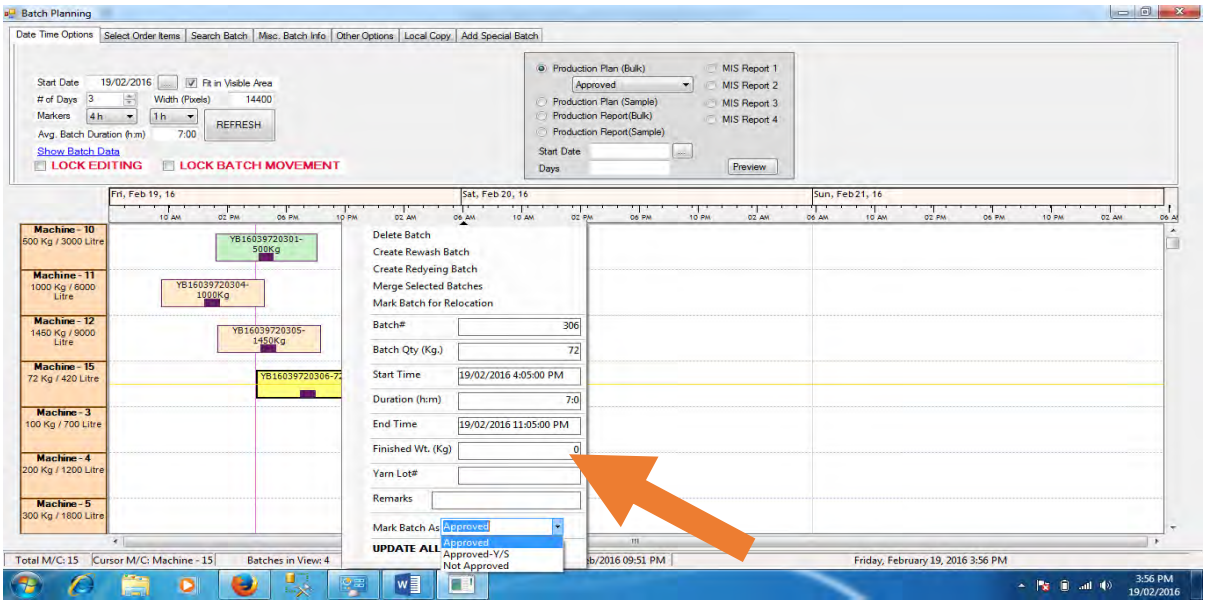


Figure 4.5: Actual Production Planning entry form with finished quantity

4.6 Actual Production Planning (Batch Status)

This is an actual production entry form with finished quantity. We can see Batch status of a certain work order.

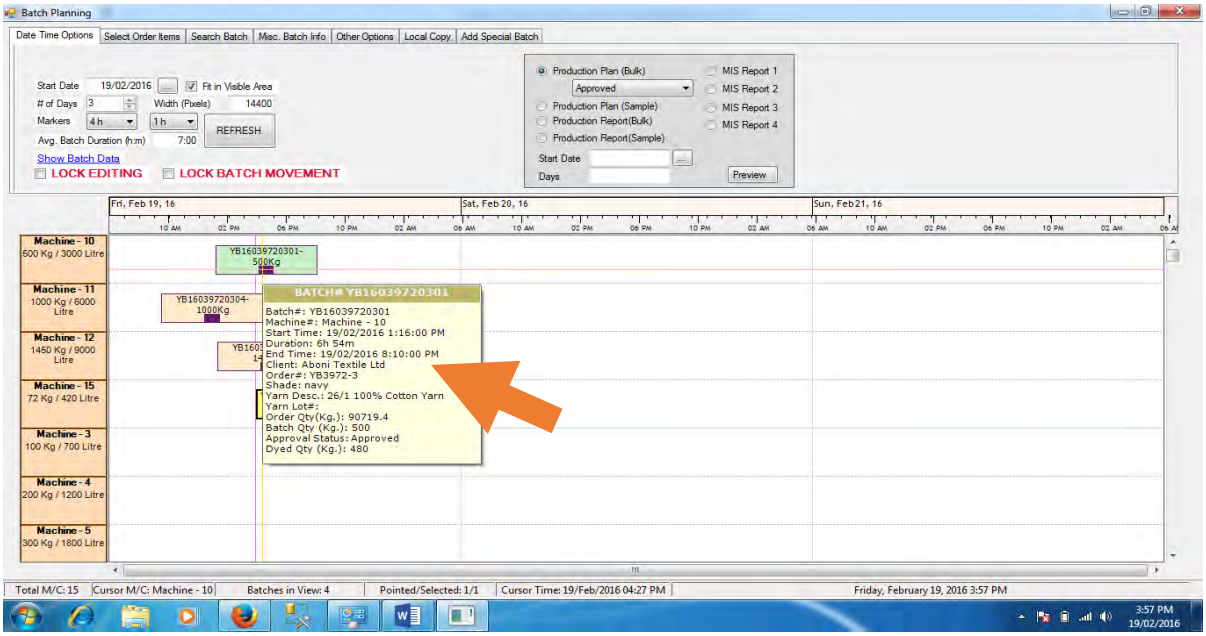


Figure 4.6: Actual Production Planning (Batch Status)

4.7 Production Planning (Batching Status with time line):

We can also see actual time line(Current time).

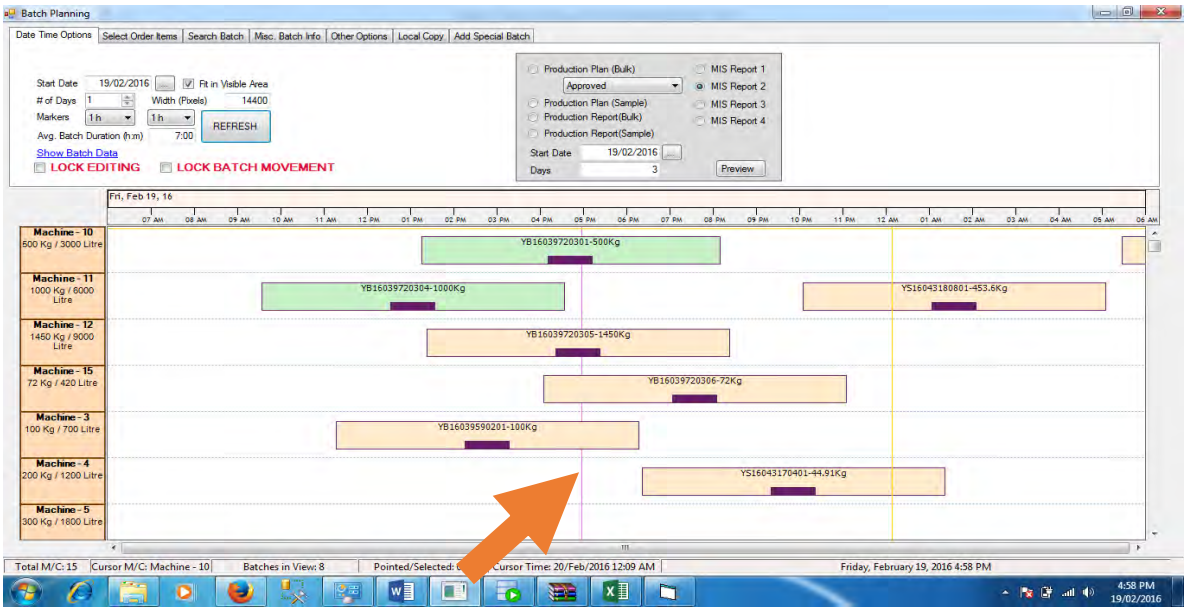


Figure 4.7: Actual Production Planning (Batch Status with time line)

4.8 Production Planning (Batching Status):

We can also batch see batch status and lock the batch from editing , number of days etc.We can show the order line items of an order and their batch status by clicking on the **Hide Batch Data** we can supress the details of an work order. We can also batch see batch status in different format . The orderlineitems can be displayed in different order . In the below figure we are seeing orders which is ordered by client .

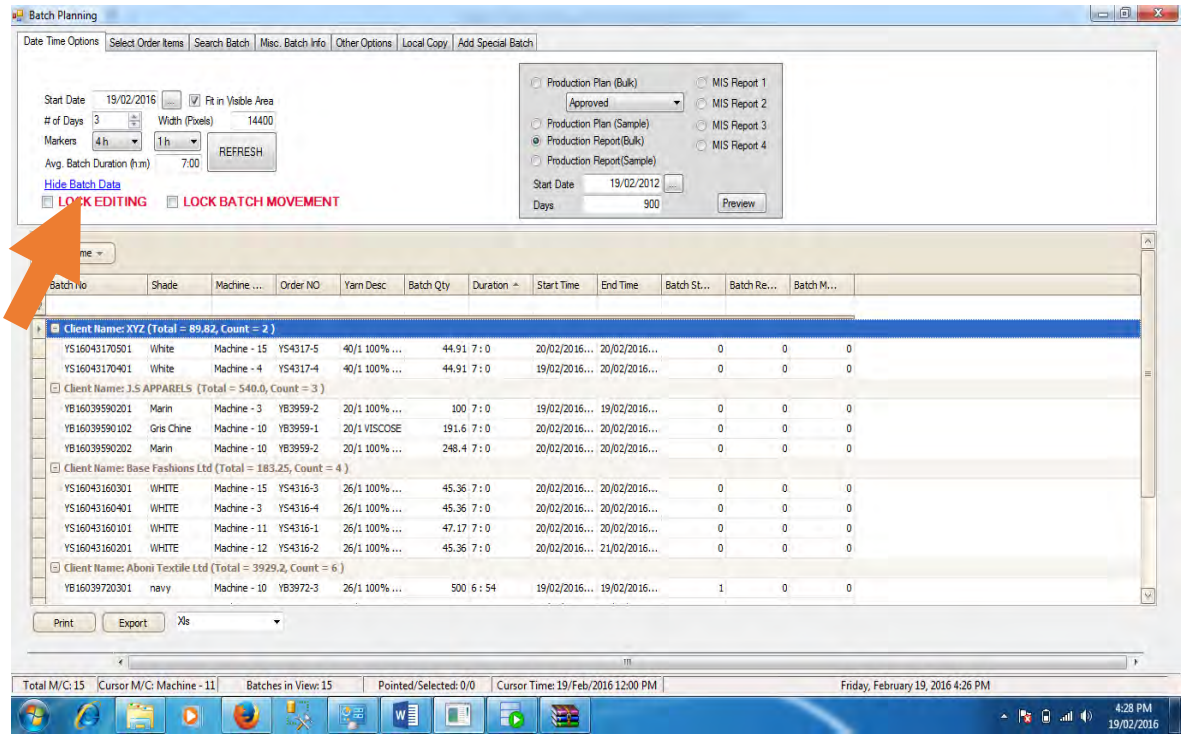


Figure 4.8.1: Production Planning Dynamic Report by Client Name(Batch Status)

In the Textile factory shade plays an important role so we decided to order the

orders by shade .

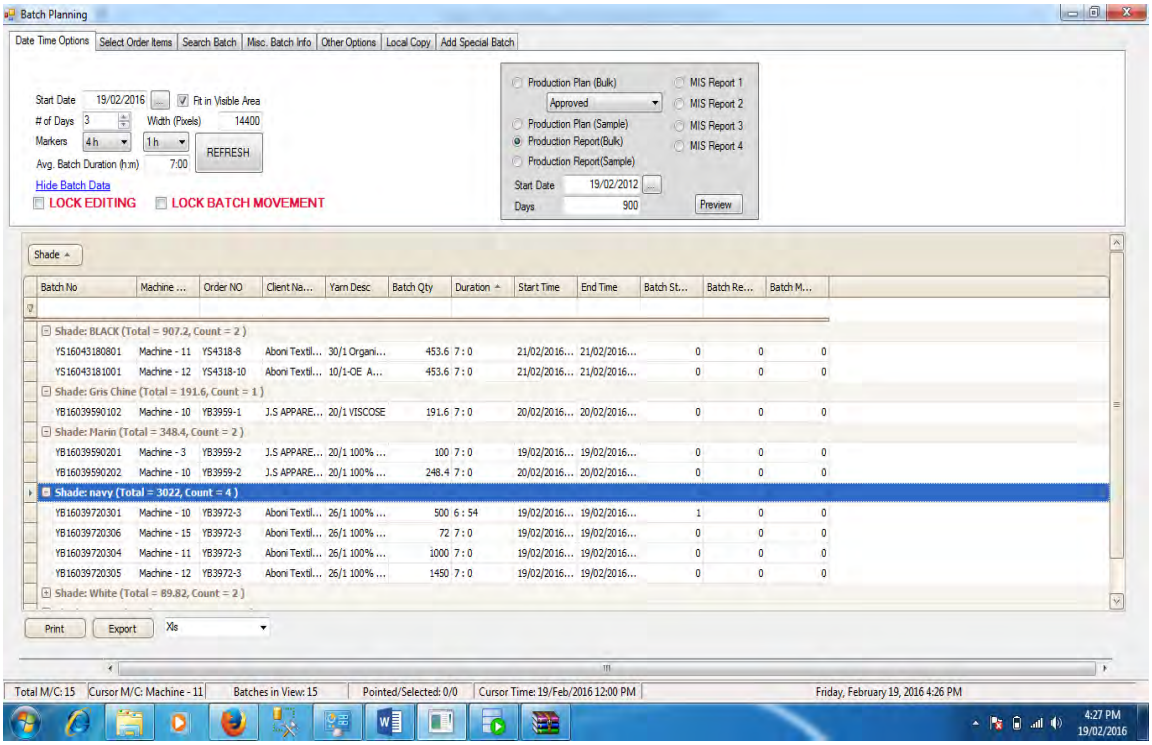


Figure 4.8.2: Production Planning Dynamic Report by Shade Name(Batch Status)

This report can be filtered by any column which makes it more flexible to use.

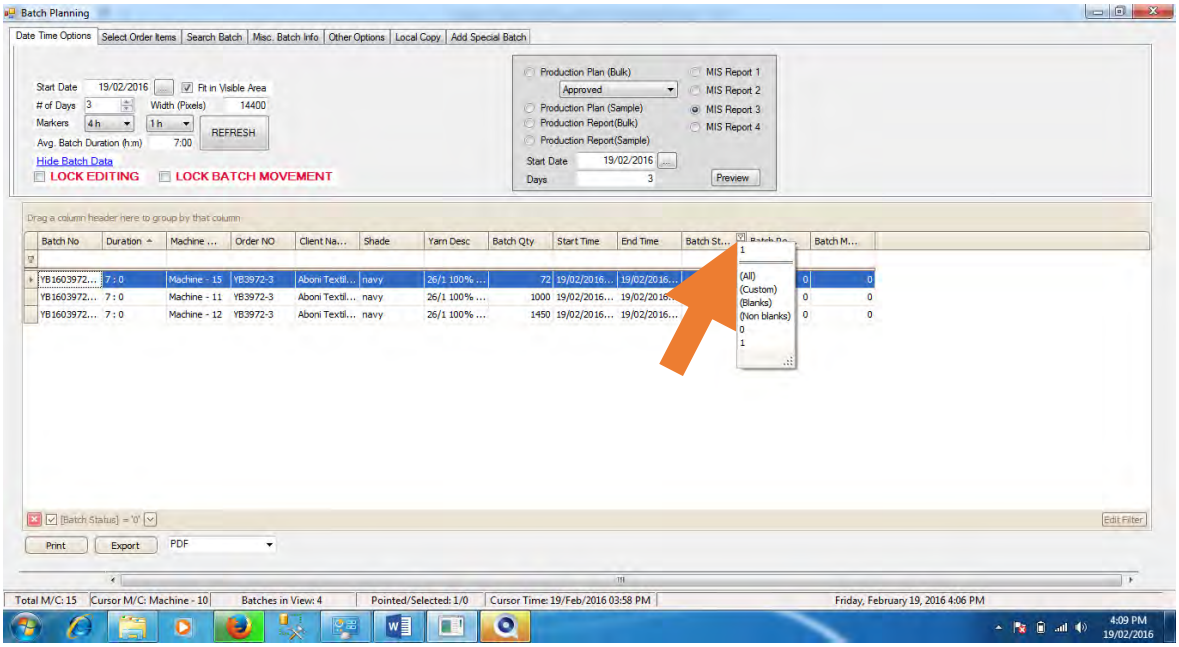


Figure 4.8.3: Production Planning Dynamic Report Filter by any field(Batch Status)

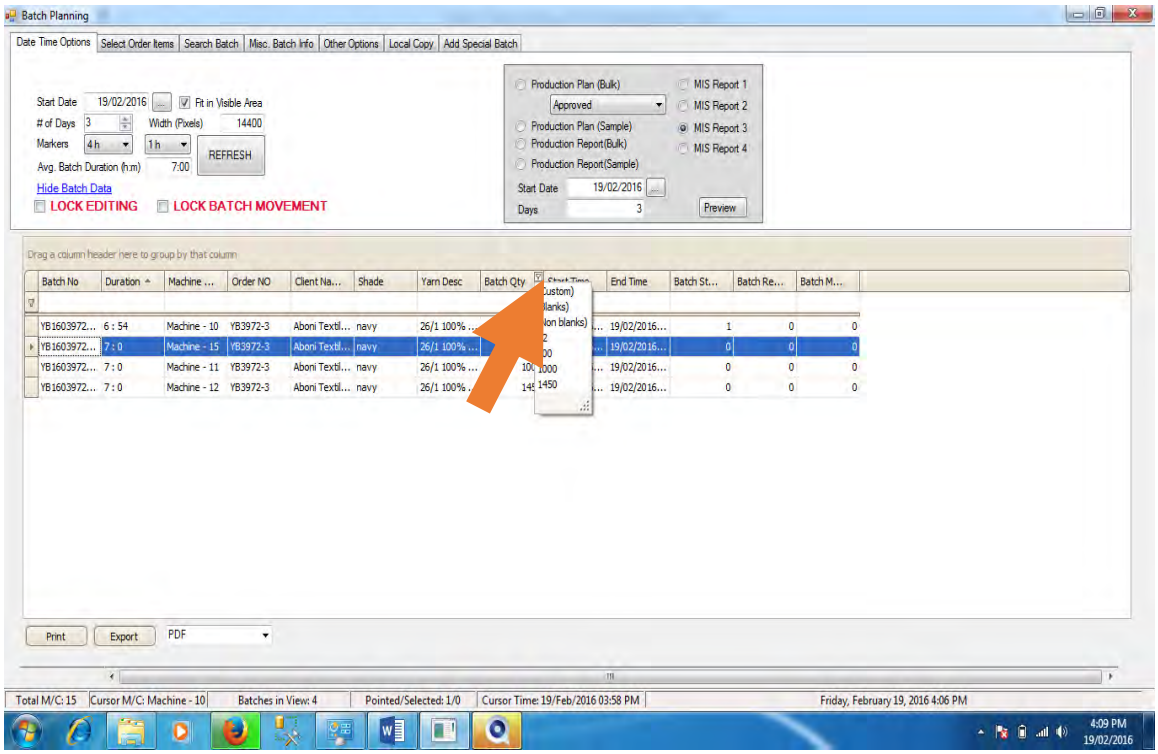


Figure 4.8.4: Production Planning Dynamic Report Filter by any field(Batch Status)

In one floor there are a several number of machines, to see the orders assigned to different machines and their details so we also keep the option of sorting the orders by machines.

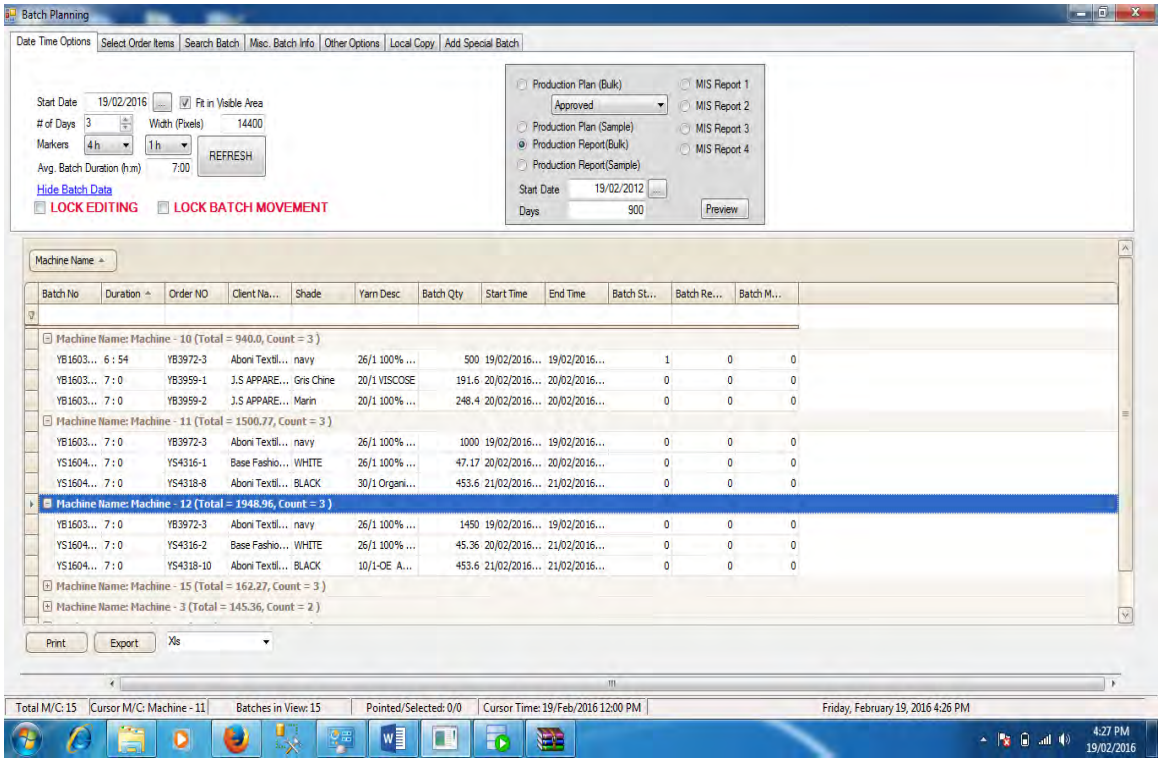


Figure 4.8.5: Production Planning Dynamic Report by Machine Name(Batch Status)

We can preview the reports dynamically. In the below figure we are seeing dynamic

report and the print option is also available.

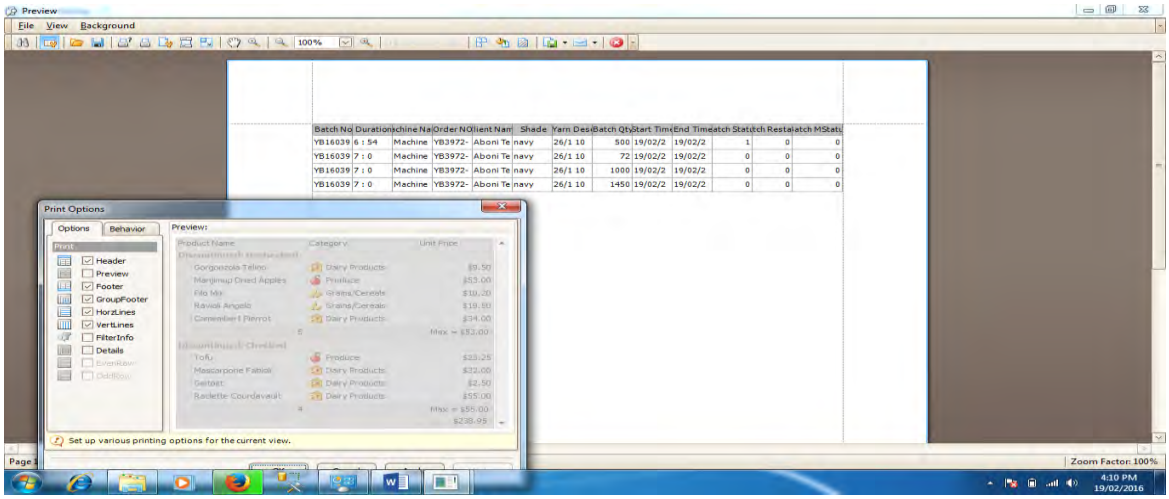


Figure 4.8.6: Production Planning Dynamic Report preview(Batch Status)

It's also possible to export the production planning in different file format. Which is shown in the below figure marked by an arrow.

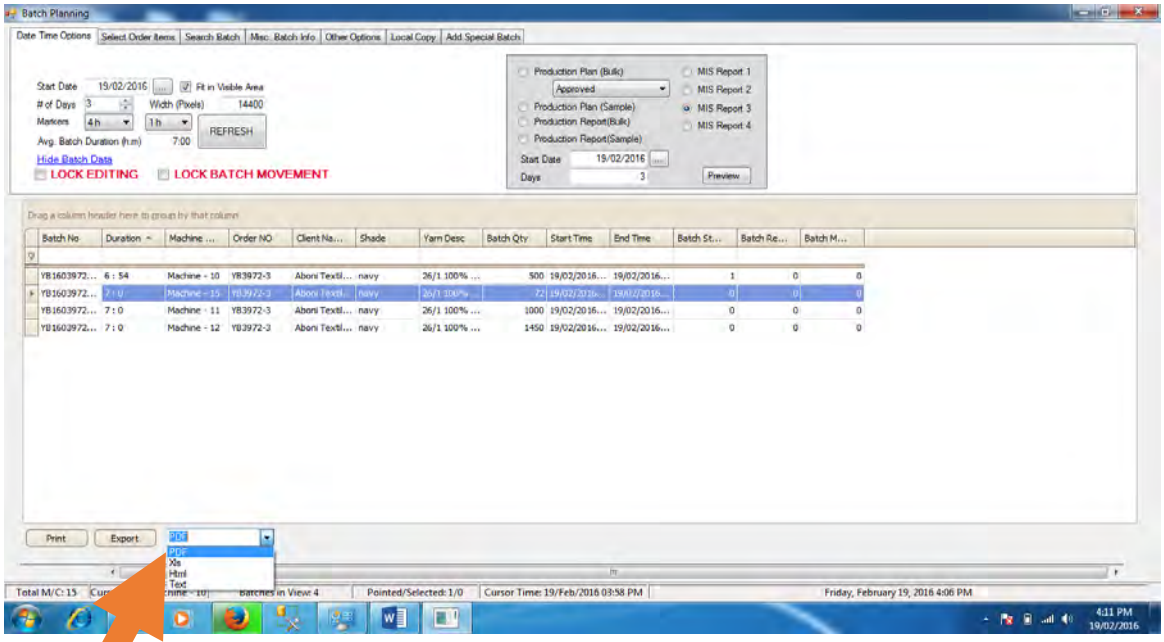


Figure 4.8.7: Production Planning Dynamic exported by pdf, excel, html and text format(Batch Status)

4.9 : Production Planning (Finish Batch Report):

We can see the entire completed batch by this report. This report is presented machine wise production.

Report: Production Plan (Bulk)

1 / 1 100%

XYZ.

Production Plan (Bulk)

Plan Date: 19/Feb/2016 to 21/Feb/2016

Approved

Machine Name	Batch Status	Priority	ClientName	OrderNo	Description	Yarn Lot#	Shade	Batch Qty(Kg)	Batch#	Approval	Dyeline Status	Load Time	Unload Time	Remarks
Machine - 10 (500 kgs)			Aboni Textile Ltd	YB3,972	26/1100% Cotton Yarn		navy	500.00	YB16039720301					
Machine - 11 (1,000 kgs)			Aboni Textile Ltd	YB3,972	26/1100% Cotton Yarn		navy	1,000.00	YB16039720304					
Machine - 12 (1,450 kgs)			Aboni Textile Ltd	YB3,972	26/1100% Cotton Yarn		navy	1,450.00	YB16039720305					
Machine - 15 (72 kgs)			Aboni Textile Ltd	YB3,972	26/1100% Cotton Yarn		navy	72.00	YB16039720306					

Figure 4.9: Production Planning (Finish Batch Report)

4.10 : **Production Planning (Batch wise Report)**

We can see the entire completed batch by this report. This report is presented batch wise production. Batching status for a current machine, loading time, unloading, quantity, dyed qty etc.

Report: Production Report

XYZ.

Production Report (Package)

Production Date: 19/Feb/2016 to 06/Aug/2018

MC No.	Batch#	Order#	Shade	Yarn Description	Dyed Qty (Kg)	Load	Unload	Length	Idle Time	Customer	Remarks
Friday, February 19, 2016											
Machine - 10 (500 kgs)	YB16039720301	YB3,972	navy	26/1 100% Cotton Yarn	480.00	13:16	20:10	6:54	9:18	Aboni Textile Ltd	
MC Total	1				480.00			6:54			
Machine - 11 (1,000 kgs)	YB16039720304	YB3,972	navy	26/1 100% Cotton Yarn		9:33				Aboni Textile Ltd	R
MC Total	1							7:0			
Machine - 12 (1,450 kgs)	YB16039720305	YB3,972	navy	26/1 100% Cotton Yarn		13:23				Aboni Textile Ltd	R
MC Total	1							7:0			
Machine - 15 (72 kgs)	YB16039720306	YB3,972	navy	26/1 100% Cotton Yarn		18:05				Aboni Textile Ltd	R
MC Total	1				0.00			7:0			
Machine - 3 (100 kgs)	YB16039590201	YB3,959	Marin	20/1 100% Cotton Yarn		11:17				J.S APPARELS	R
MC Total	1							7:0			
Machine - 4 (200 kgs)	YS16043170401	YS4,317	White	40/1 100% Cotton Yarn		18:22				XYZ	R
MC Total	1							7:0			
Bulk Total	6				480.00			41:54			
Day Total	6				480.00			41:54			
Saturday, February 20, 2016											
Machine - 10 (500 kgs)	YB16039590102	YB3,959	Gris Chine	20/1 VISCOSE		5:28				J.S APPARELS	R

DailyProductionReport.rpt 1

Figure 4.10: Production Planning (Batch wise Report)

4.11: **Production Planning (Batch)**

We can see the entire completed batch by this report. This report is presented by batch wise production and its details.

Report: Production Plan (MIS 1)

1 / 1 100%

XYZ

DHAKA

Quantity(Kg)								
Production (Dyeing) of 19/02/2016				Production(Winding) of 19/02/2016			Delivery of 19/02/2016	
Yarn Quantity	Bulk	Sample	Total	Yarn Quantity	Soft	Random	Yarn Quantity	
Single	480.00		480.00	Single			Single	
Total	480.00		480.00	Total			Total	
Grand Total			480.00	Up to Work Order			Except Work Order	

Figure 4.11: Production Planning (Batch)

4.12: **Production Planning (Batch)**

We can see the batch details by this report. This report is presented by batch planning. Like how many batches is approved, not approved and rejected is shown in this report.

Report: Production Plan (MIS 2)

1 / 1 100%

XYZ

DHAKA

	Bulk									Sample		
	Sweater			Yarn			Grand Total			Sweater	Yarn	Grand Total(Kg.)
	Approved	Appr-Y/S	Not App.	Approved	Appr-Y/S	Not App.	Approved	ApprBut-Y/S	Not App.			
	0.00	0.00	0.00	1072.00	0.00	0.00	1072.00	0.00	0.00	0.00	1,450.00	1450.00
	0.00			1,072.00			1,072.00	0.00	0.00	0.00	1,450.00	1,450.00

Figure 4.12: Production Planning (Batch)

4.13: **Production Planning (Batch Status)**

This report is presented by batch status. It shows machine status wise batches and their details like client information, machine information. So is a batch is rejected it could be revised and helps to undertake other production related decision.

Report: Production Plan (MIS 3)

1 / 1 100%

XYZ

DHAKA

Bulk									
MC No	Sweater			Yarn			Grand Total		
	Approved	Appr-Y/S	Not App.	Approved	Appr-Y/S	Not App.	Approved	ApprBut-Y/S	Not Appr.
Machine - 11				1			1		
Machine - 15				1			1		
TOTAL				2			2		

Sample									
MC No	Sweater			Yarn			Grand Total		
	Approved	Appr-Y/S	Not App.	Approved	Appr-Y/S	Not App.	Approved	ApprBut-Y/S	Not Appr.
Machine - 12				1			1		
TOTAL				1			1		

Figure 4.13: Production Planning (Batch Status)

4.14 Summary

In this chapter we have described about various important functionalities such as auto batching system, production planning system for next 30 days machine availability etc. We have generate various reports like total machine status report, production reports also Pivot report that user can generate dynamically by using Dev Express.

Chapter 5

Conclusion

5.1 Conclusion:

Production planning software system is a demand of this day for any textile industry in this era of Information and Communication Technology (ICT). In order to fulfill this demand, this project presented a prototype of a production planning software for the textile industry with different types of features so that user can easily posted transactional data by visual mode such as auto batch creation, batch drag and drop into plan board. In this system we used three tier architecture i.e. business layer and presentation layer and database layer for managing data easily, graphical environments is faster and more secure. We also used dynamic report so that user can view actual production, production process tracking, production plan, dynamic actual production report etc. Due to the different types of unique features, it is more user friendly to the users. The system is self descriptive. Users can easily access the system and view the information they need. Production planning software has been developed aiming at inventory reduction, increased productivity, improved customer service and control of the business in a textile industrial unit. This system is expected to improve the life style of user in terms of real production flavor of their necessary plan. Administrator of the software can control the all work order information, production planning, add, delete, modify and operator view only. However here are some limitations of this software. It follows client server based approach although web based approach is the norm of this day.

5.2 Future works

Author suggests the following future work for enhancement of the project.

This system should be web-based so that user can perform their task anywhere by using their devices. Designed software can be enhanced and some more new features can be added to facilitate user needs.

References

[1] A.C. Hax, Aggregate production planning, in: J. Moder and S. Elmaghraby (eds.) Handbook of Operations Research, Van Nostrand Reinhold, New York (1978).

[2] Veeke, H.P.M., and Lodewijks, G. (2005). Simulation application framework for production planning and scheduling. 3rd Intl Industrial Simulation Conference, Berlin, Germany.

[3] Wight, O.W.(1984). Production and inventory management in the computer age. New York, U.S.A: Van Nostrand Reinhold Company.

[4] Apress.Expert.C.Sharp.2005.Business.Objects.2nd.Edition.Mar.2006

[5] oreilly.oracle.pl.sql.programming.6th.edition. Jan 2014

[6] http://www.oracle.com ; Last accessed on March 2016

[7] http://www.w3school.com; Last accessed on March 2016

[8] http://www.devexpress.com; Last accessed on March 2016

[9] http://www.crystalreports.com; Last accessed on March 2016

[10] http://en.wikipedia.org/wiki/Web_application; Last accessed on March 2016

[11] http://www.ovum.com/mocca/content/tt2/pay.pdf; Last accessed on March 2016

[12] http://www.toletdhakabd.com/tolet.php; Last accessed on October 2015

[13] Expert C# 2008 Business Objects. Copyright © 2009 by Rockford Lhotka

[14] http://en.wikipedia.org/wiki/Software_development_methodology; Last accessed on March 2016

[15] http://en.wikipedia.org/wiki/V-Model_ (software_development); Last accessed on March 2016

Appendix A

5.3 User Login

The bellow figure 4.1 is login form for enter the software. The user/admin/staff has to enter his username and password in the respective text box and as to click the "login" button to enter this system.

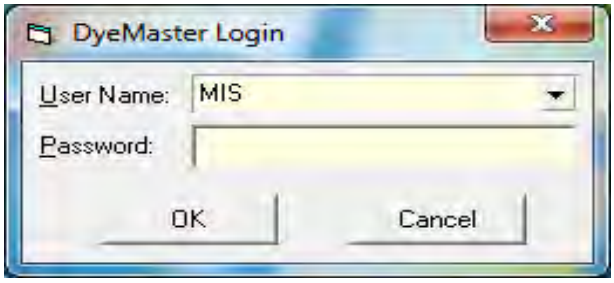


Figure 5.3: Login Form

5.4 Master Form

Figure 5.4 is the Master Form of the system. After login to the system, the bellow figure should be displayed. User can entry work order, client information, employee information, batch entry etc. as per permission.

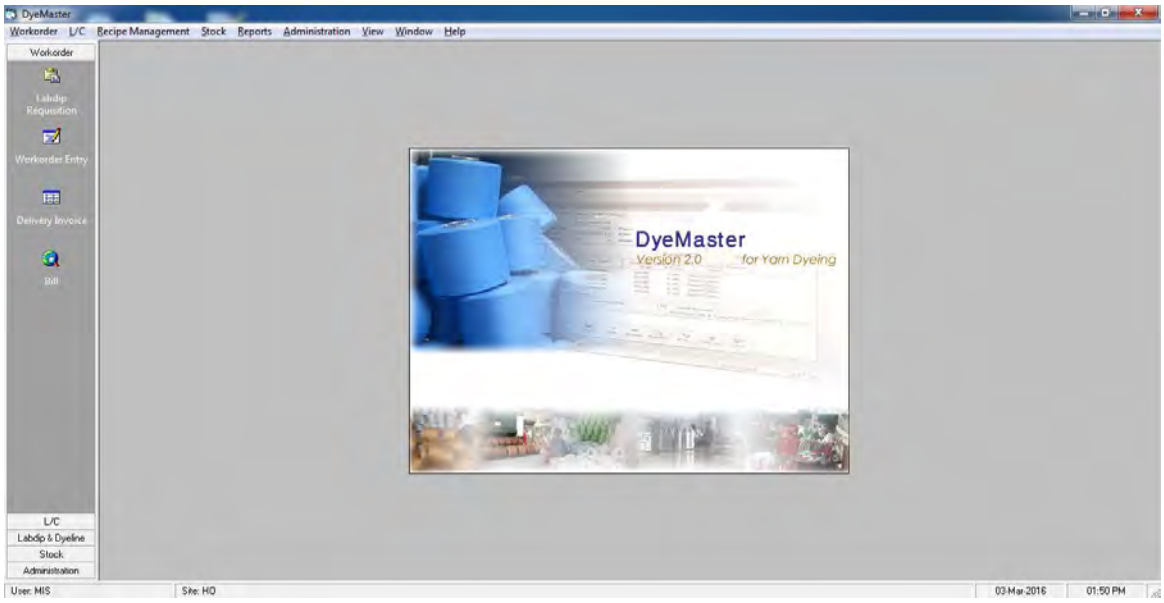


Figure 5.4: Master Form

5.5 Work Order Form:

The figure 5.5 is work order entry form. The user can input necessary information of a work order.

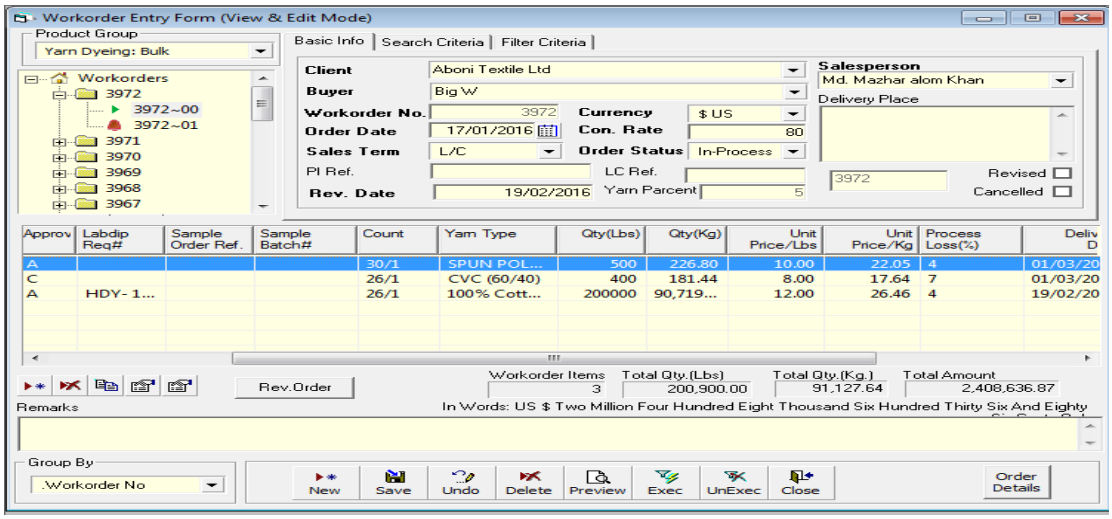


Figure 5.5: Admin/User (Full Access) Work Order Entry Form

5.6 Work Order Form (Read Only):

The figure 5.6 is work order entry form (Read Only). The user only views this form.

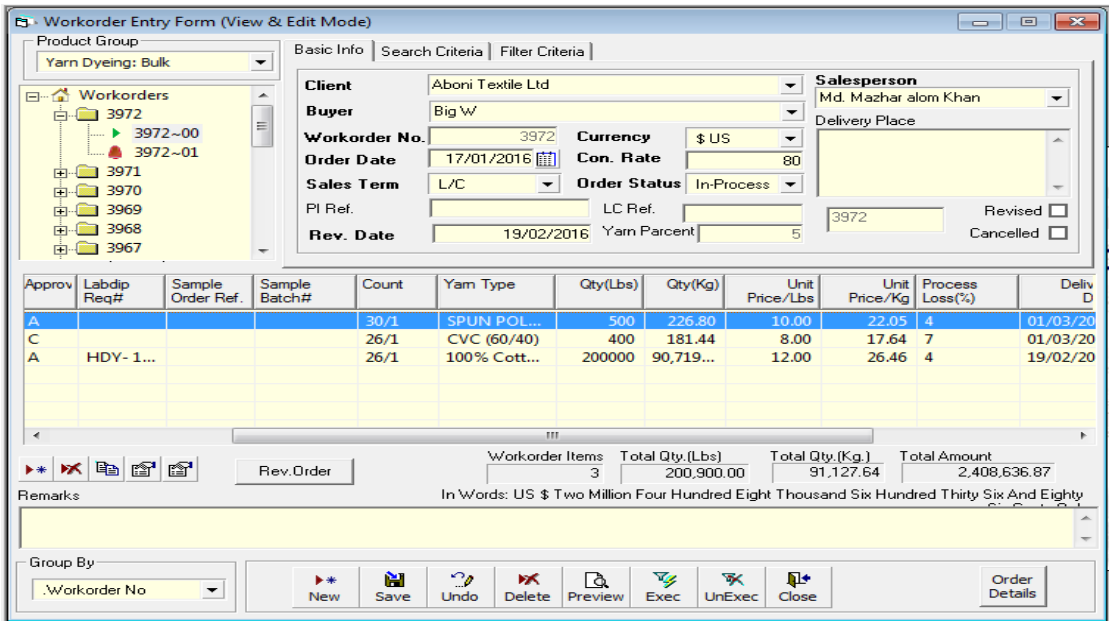


Figure 5.6: User (Read Only Access) Entry Form

5.7 Work Order Form (No Access):

The figure 5.7 is work order entry form (**No Access**). The unauthorized user cannot access this form.

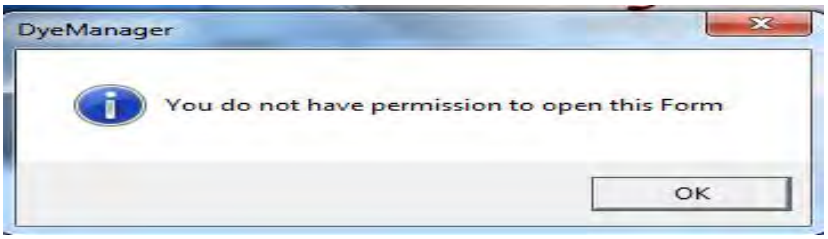


Figure 5.7: User (No Access) Entry Form

5.8 Employee or user Form:

The figure 5.8 is employee or user entry form. The user can input necessary information of an employee or user.

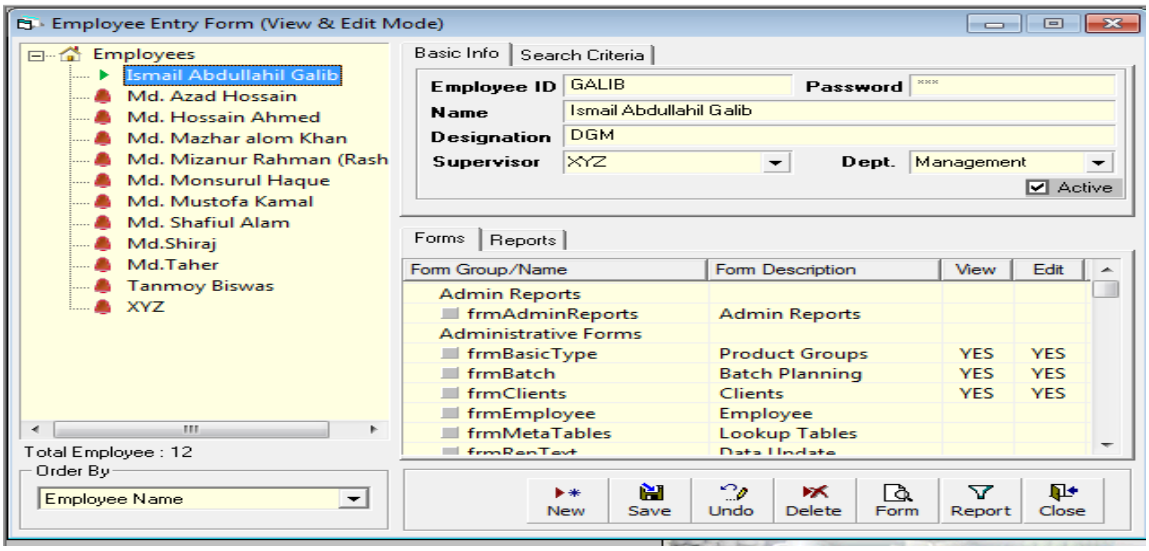


Figure 5.8: User/Registration

5.9 Employee or user Form:

The figure 5.9 is employee or user entry form. The user can input necessary information of an employee or user . Below that there is permission list for forms and reports of that following employee which he/she could access.

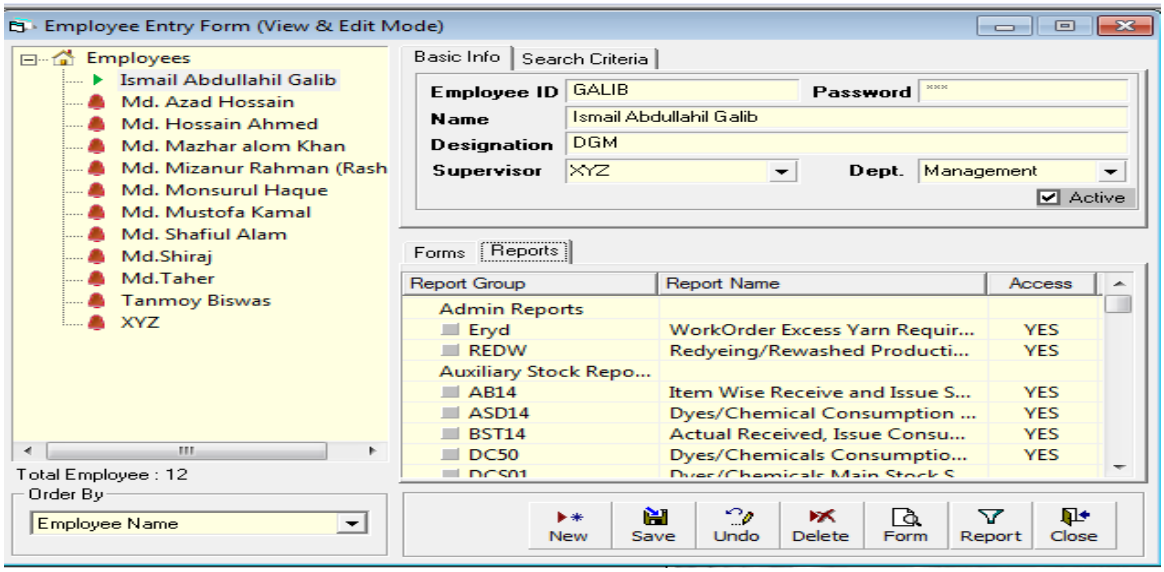


Figure 5.9: User/Registration Entry Form with form Permission list

5.10 Work Order Entry Form (View & Edit Mode):

The figure 5.10 is work order entry form (View & Edit Mode). The user uses this form for edit a specific work order and view it.

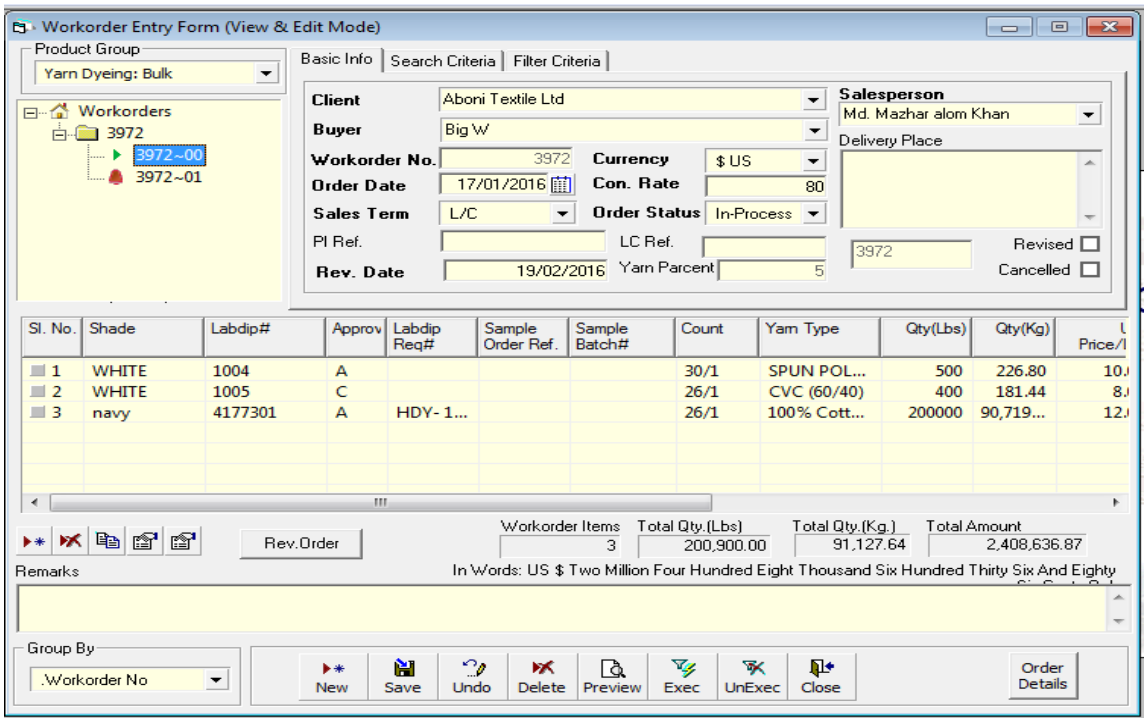


Figure 5.10: Work Order Entry Form

5.11 Work Order Entry Form (View & Edit Mode):

The figure 5.11 is Gray yarn issue to floor Entry Form. The user uses this form for issuing Gray yarn to dyeing section and completed yarn issue to finished store.

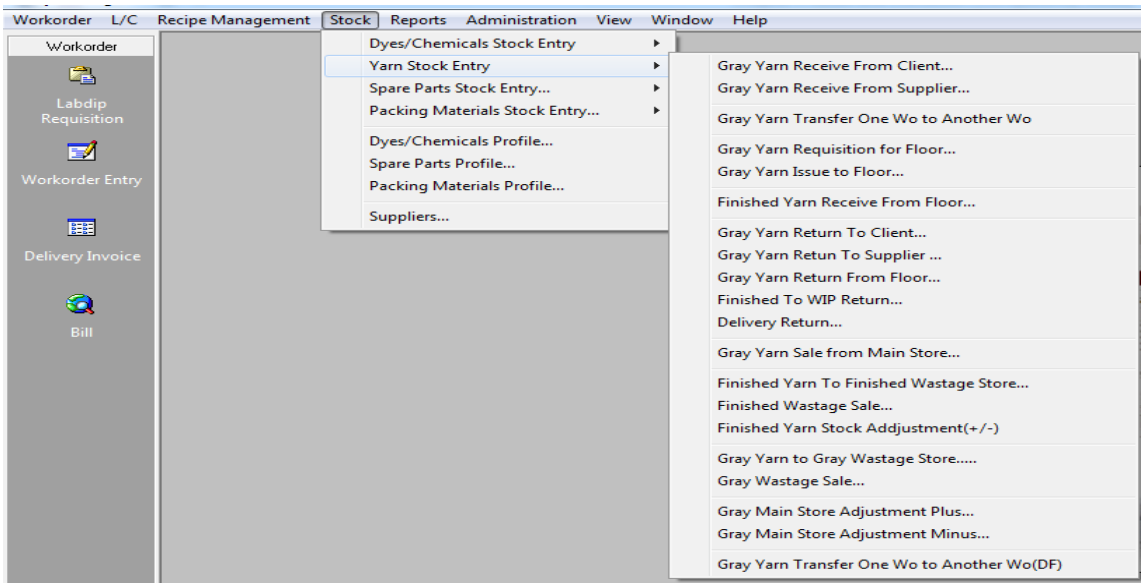


Figure 5.11: Gray yarn issue to batching floor Entry Form

5.12 Gray yarn received from client Entry Form:

The figure 5.12 is Gray yarn received from client Entry Form. The user uses this form for received from client for dyeing.

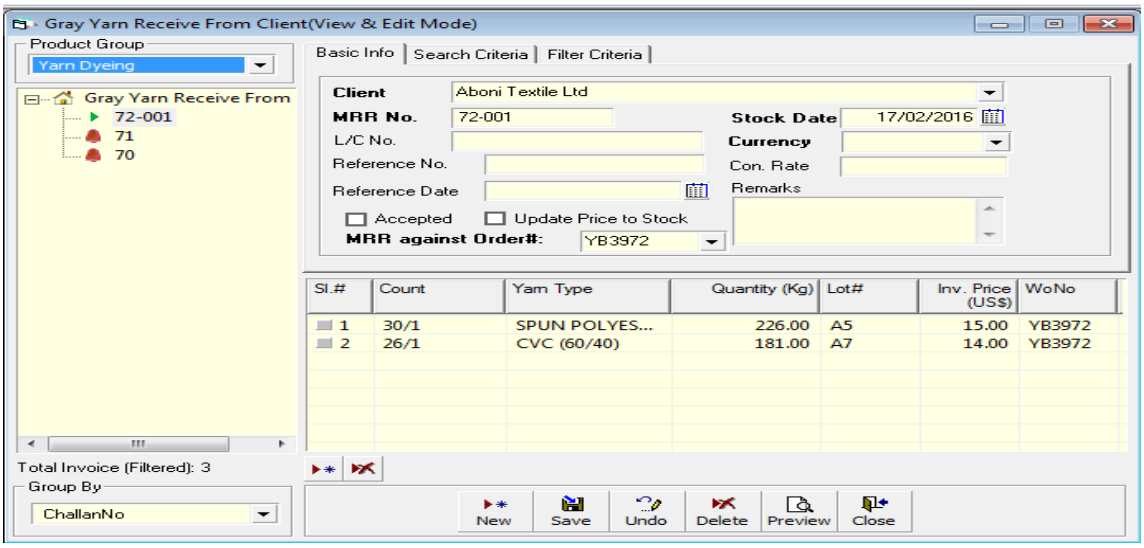


Figure 5.12: Gray yarn received from client

5.13 Gray Yarn Requisition Entry Form:

The figure 5.13 is Gray Yarn Requisition Entry Form. The user uses this form for giving entry to WIP Yarn Stock Entry Form in the left side we saw others entry by Challan No.

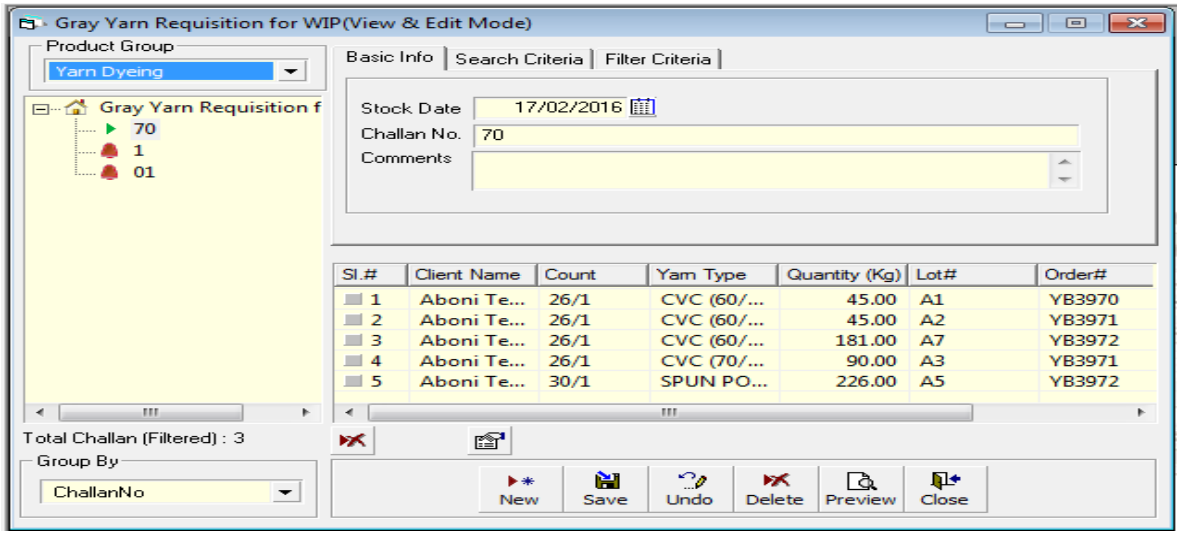


Figure 5.13: Gray Yarn Requisition Entry Form

5.14 WIP Yarn Stock Entry Form:

The figure 5.14 is WIP Yarn Stock Entry Form. The user uses this form for giving entry to WIP Yarn Stock Entry Form in the left side we saw others entry by Challan No.

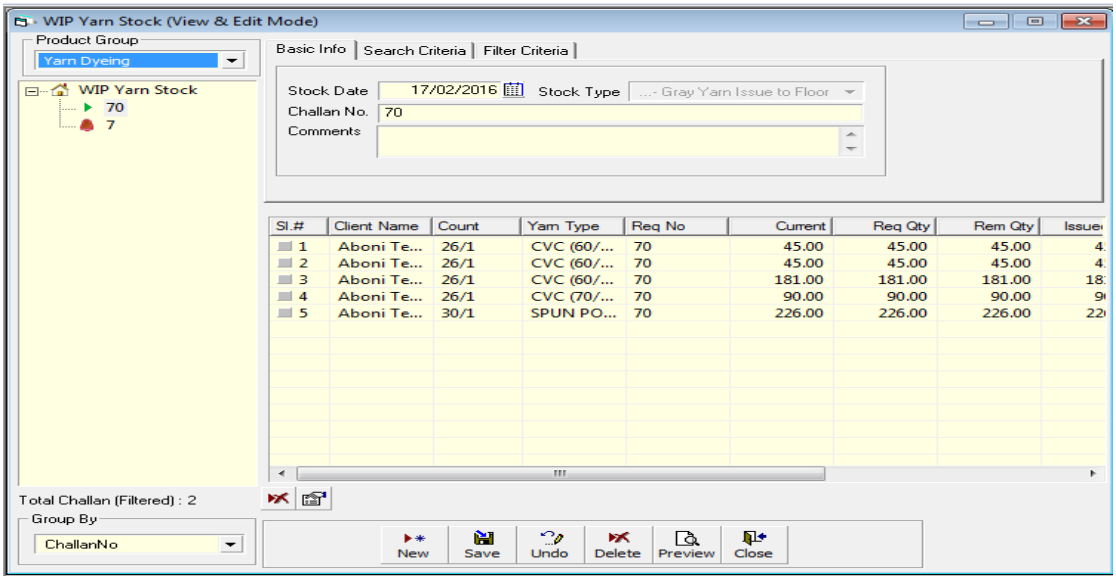


Figure 5.14: WIP Yarn Stock Entry Form

5.15 Dyeline Entry Form:

This form will be used for giving entry of Dyeline.

How to Open Dyed Yarn Entry Form

- Select Recipe Management> Dyeline Entry Form.
- The Dyeline Entry Form is displayed.
- We can view; give entry, and editing an existing entry.

- Below basic info section we can see Dyeline heads in left side and amount of used dyes and chemical right side of the screen.

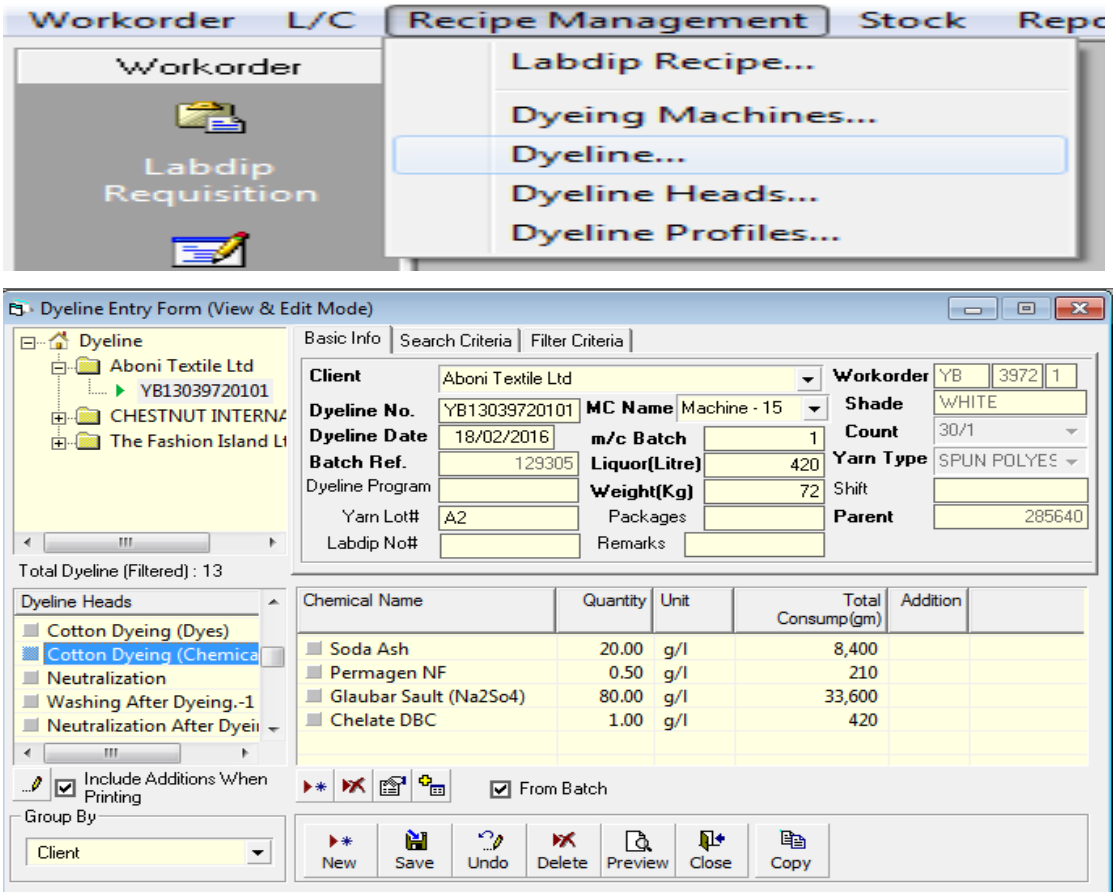


Figure 5.14: Dyeline Entry Form

5.16 Finished Yarn Stock Entry Form:

This form will be used for giving entry of Finishing yarn stock. View and editing was enabled in this form.

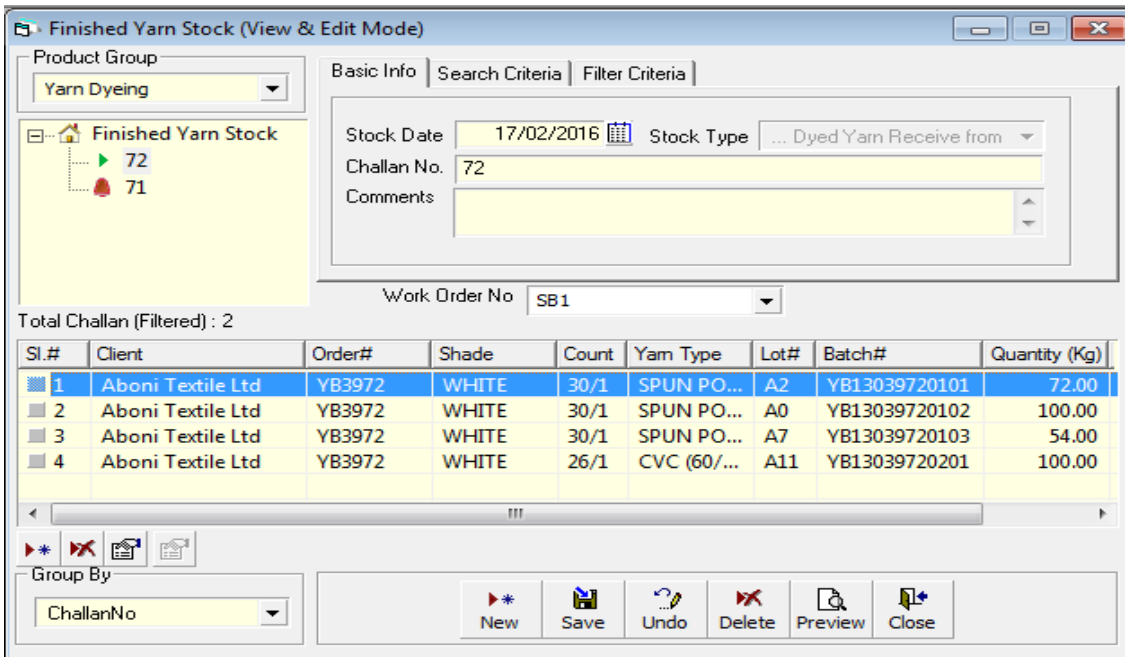


Figure 5.16: Finished Entry Form

5.17 Delivery Entry Form:

This form will be used for giving entry of Delivery Entry Form. View and editing was enabled in this form. When a yarn dyeing order is completed entry is given here of that completed work.

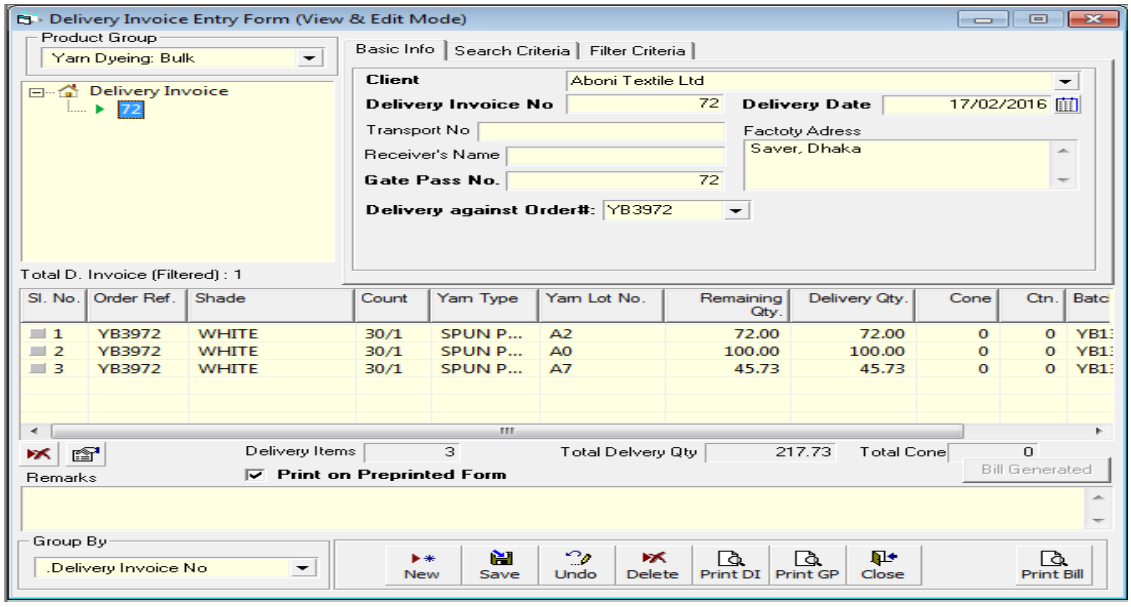


Figure 5.17: Delivery Entry Form

5.18 Work Order Report:

We can see all information including client information of a selected work order.

We also can print thing information.

1 / 1 100%

XYZ

Yarn Dyeing: Bulk Workorder

Rev. No : 0 Main Order Rev. Date : 19/02/2016

Clients Name :	Aboni Textile Ltd	*Work Order No* :	YB3972
Buyer Name :	Big W	*Sales Term* :	L/C
Contact Person :		*Date* :	17/01/2016
Delivery Place :		*PI Ref* :	
		LC Ref :	

SL NO	*Shade*	*Labdip#*	*Labdip Req# / Sample Batch#*	*Type Of Yarn*	*Quantity (Kg)*	*Price/ Kg*	*Delivery Date*	*Client Ref.*	*Remarks*
1	WHITE	1004-A		30/1 SPUN POLYESTER	226.80	22.05	01/03/2016		
2	WHITE	1005-C		26/1 CVC (60/40)	181.44	17.64			
3	navy	4177301-A	HDY- 130012	26/1 100% Cotton Yarn	90,719.40	26.46	19/02/2016		
			US $ @ Tk.80.00	*Total:*	*91,127.64*	*US $ 2,408,636.87*			

In Words: US $ Two Million Four Hundred Eight Thousand Six Hundred Thirty Six And Eighty Six Cents Only

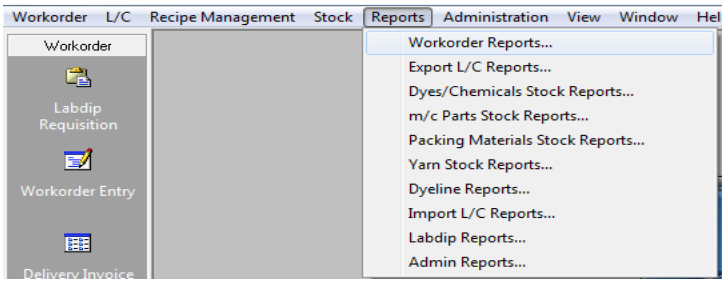
Remarks :

Figure 5.18: Work Order Report

5.19 Work Order Report(View):

How to Open Work Order Report

- Select Reports> Work order Reports Form.
- We have to give client's name, order status and a time frame then we can show information of all the work order in that time frame.



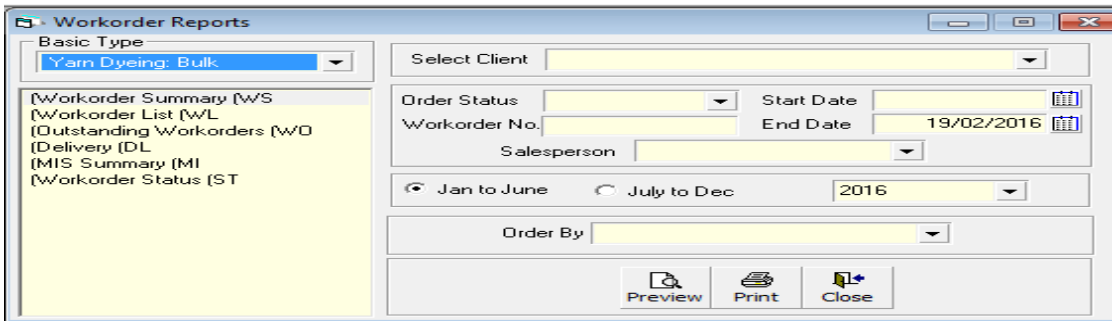


Figure 5.19: Work Order Report(View)

5.20 Yarn Stock Reports (View):

How to Open Yarn Stock Reports

- Select Reports> Yarn Stock Reports Form.
- We have to give client's name, order status, count, yarn type and a time frame then we can show information of all yarn stock in that time frame.
-

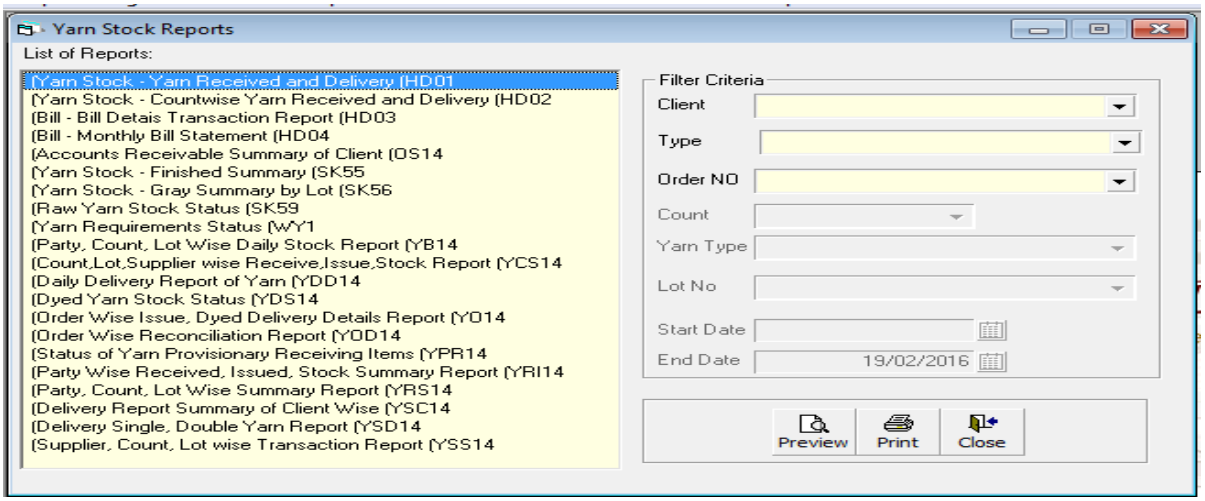


Figure 5.20: Yarn Sock Report

5.21 Dyeline Reports (View):

How to Open Dyeline Reports

- Select Reports> Dyeline Reports Form.
- We have to give a time frame then we can show information of all Dyeline in that time frame.

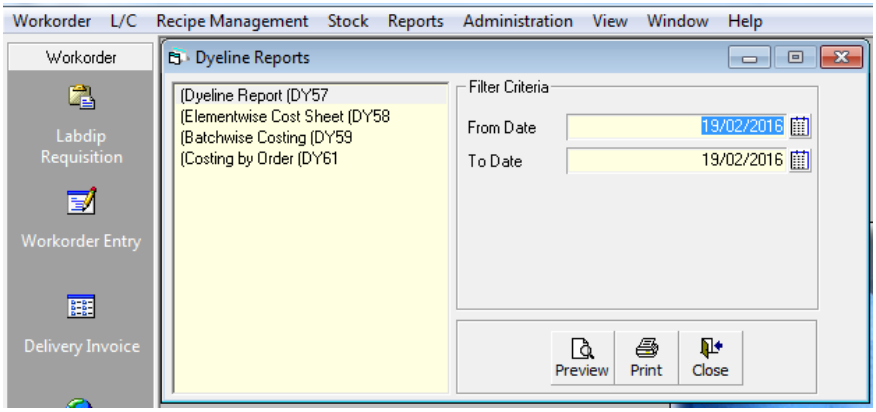


Figure 5.21: Dyeline Report

Appendix B

Data used for the Development

For simplicity, thirty tables have been Designed namely T_EMPGROUP, T_EMPLOYEE, T_SQLLOG, T_WORDER, T_WOITEMS, T_UNITOFMEAS, T_YARNTYPE, T_COUNT, T_CURRENCY, T_CLIENT, T_DBATCH, T_DBATCHTEMP, T_DINVOICE, T_DI NVOICEITEMS. Brief description of the tables is given below:

```
 CREATE TABLE "DYPROHDNEW"."T_EMPGROUP"
  (        "EMPGROUPID" NUMBER(2,0),
           "EMPGROUP" VARCHAR2(20 BYTE)
);

 CREATE TABLE "DYPROHDNEW"."T_EMPLOYEE"
  (        "EMPLOYEEID" VARCHAR2(8 BYTE),
           "EMPLOYEE" VARCHAR2(100 BYTE) CONSTRAINT "NN_T_EMPLOYEE_EMPLOYEE" NOT
NULL ENABLE,
           "DESIGNATION" VARCHAR2(100 BYTE) CONSTRAINT
"NN_T_EMPLOYEE_DESIGNATION" NOT NULL ENABLE,
           "INOPERATIONID" NUMBER(1,0) CONSTRAINT "NN_T_EMPLOYEE_INOPERATIONID"
NOT NULL ENABLE,
           "EMPPWD" VARCHAR2(8 BYTE) CONSTRAINT "NN_T_EMPLOYEE_EMPPWD" NOT NULL
ENABLE,
           "EMPMANAGER" VARCHAR2(8 BYTE) CONSTRAINT "NN_T_EMPLOYEE_EMPMANAGER"
NOT NULL ENABLE,
           "EMPGROUPID" NUMBER(2,0) CONSTRAINT "NN_T_EMPLOYEE_EMPGROUP" NOT NULL
ENABLE,
            CONSTRAINT "PK_T_EMPLOYEE" PRIMARY KEY ("EMPLOYEEID"));

 CREATE TABLE "DYPROHDNEW"."T_SQLLOG"
  (        "LOGSEQ" NUMBER(20,0),
           "USERNAME" VARCHAR2(30 BYTE) CONSTRAINT "NN_T_SQLLOG_USERNAME" NOT
NULL ENABLE,
           "TERMINAL" VARCHAR2(30 BYTE) CONSTRAINT "NN_T_SQLLOG_TERMINAL" NOT
NULL ENABLE,
           "TSTAMP" DATE DEFAULT sysdate CONSTRAINT "NN_T_SQLLOG_TSTAMP" NOT NULL
ENABLE,
           "COPIED" NUMBER(1,0) DEFAULT 0 CONSTRAINT "NN_T_SQLLOG_COPIED" NOT NULL
ENABLE,
           "SQLTEXT" VARCHAR2(4000 BYTE) CONSTRAINT "NN_T_SQLLOG_SQLTEXT" NOT NULL
ENABLE,
            CONSTRAINT "PK_T_SQLLOG" PRIMARY KEY ("LOGSEQ")
);
```

```
CREATE TABLE "DYPROHDNEW"."T_WORDER"
(       "BASICTYPEID" VARCHAR2(3 BYTE),
        "WONUMBER" NUMBER(6,0),
        "WODATE" DATE CONSTRAINT "NN_T_WORDER_WODATE" NOT NULL ENABLE,
        "CLIENTID" NUMBER(4,0) CONSTRAINT "NN_T_WORDER_CLIENTID" NOT NULL
ENABLE,
        "SALESTERMID" NUMBER(1,0) CONSTRAINT "NN_T_WORDER_SALESTERMID" NOT
NULL ENABLE,
        "CURRENCYID" NUMBER(3,0) CONSTRAINT "NN_T_WORDER_CURRENCYID" NOT NULL
ENABLE,
        "CONRATE" NUMBER(12,4) CONSTRAINT "NN_T_WORDER_CONRATE" NOT NULL
ENABLE,
        "DELIVERYPLACE" VARCHAR2(200 BYTE),
        "WSTATUSID" NUMBER(1,0) CONSTRAINT "NN_T_WORDER_WSTATUSID" NOT NULL
ENABLE,
        "CLIENTSREF" VARCHAR2(100 BYTE),
        "EMPLOYEEID" VARCHAR2(8 BYTE) CONSTRAINT "NN_T_WORDER_EMPLOYEEID" NOT
NULL ENABLE,
        "WCANCELLED" NUMBER(1,0) DEFAULT 0 CONSTRAINT
"NN_T_WORDER_WCANCELLED" NOT NULL ENABLE,
        "WREVISED" NUMBER(1,0) DEFAULT 0 CONSTRAINT "NN_T_WORDER_WREVISED" NOT
NULL ENABLE,
        "WOCOMMENTS" VARCHAR2(200 BYTE),
        "BUYERID" NUMBER(4,0) CONSTRAINT "NN_T_CLIENT_BUYERID" NOT NULL ENABLE,
        "PIORLCREF" VARCHAR2(200 BYTE),
        "REVISEDCOUNT" NUMBER(3,0),
        "REVISEDDATE" DATE,
        "DWONUMBER" NUMBER(7,0),
        "MAXREVISEDCOUNT" NUMBER(3,0),
        "LCREF" VARCHAR2(200 BYTE),
         CONSTRAINT "C_T_WORDER_WCANCELLED" CHECK (WCancelled in(0,1)) ENABLE,
         CONSTRAINT "C_T_WORDER_WREVISED" CHECK (WRevised in(0,1)) ENABLE,
         CONSTRAINT "PK_T_WORDER" PRIMARY KEY ("BASICTYPEID", "WONUMBER")
USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
STORAGE(INITIAL 524288 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "USERS"  ENABLE,
         CONSTRAINT "FK_T_WORDER_T_EMPLOYEE" FOREIGN KEY ("EMPLOYEEID")
          REFERENCES "DYPROHDNEW"."T_EMPLOYEE" ("EMPLOYEEID") ENABLE,
         CONSTRAINT "FK_T_WORDER_T_CURRENCY" FOREIGN KEY ("CURRENCYID")
          REFERENCES "DYPROHDNEW"."T_CURRENCY" ("CURRENCYID") ENABLE,
         CONSTRAINT "FK_T_WORDER_T_CLIENT" FOREIGN KEY ("CLIENTID")
          REFERENCES "DYPROHDNEW"."T_CLIENT" ("CLIENTID") ENABLE,
         CONSTRAINT "FK_T_WORDER_T_BASICTYPE" FOREIGN KEY ("BASICTYPEID")
```

```
 REFERENCES "DYPROHDNEW"."T_BASICTYPE" ("BASICTYPEID") ENABLE,
CONSTRAINT "FK_T_WORDER_T_WSTATUS" FOREIGN KEY ("WSTATUSID")
 REFERENCES "DYPROHDNEW"."T_WSTATUS" ("WSTATUSID") ENABLE,
CONSTRAINT "FK_T_WORDER_T_SALESTERM" FOREIGN KEY ("SALESTERMID")
 REFERENCES "DYPROHDNEW"."T_SALESTERM" ("SALESTERMID") ENABLE
);

 CREATE TABLE "DYPROHDNEW"."T_WOITEMS"
 (       "BASICTYPEID" VARCHAR2(3 BYTE),
         "WONUMBER" NUMBER(6,0),
         "WOITEMSL" NUMBER(3,0),
         "TICKET" VARCHAR2(10 BYTE),
         "SHADE" VARCHAR2(40 BYTE) CONSTRAINT "NN_T_WOITEMS_SHADE" NOT NULL
ENABLE,
         "COUNTID" NUMBER(3,0) CONSTRAINT "NN_T_WOITEMS_COUNTID" NOT NULL
ENABLE,
         "YARNTYPEID" NUMBER(3,0) CONSTRAINT "NN_T_WOITEMS_YARNTYPEID" NOT NULL
ENABLE,
         "LABDIPID" VARCHAR2(30 BYTE),
         "WOITEMSQTY" NUMBER(12,2) CONSTRAINT "NN_T_WOITEMS_WOITEMSQTY" NOT
NULL ENABLE,
         "WOITEMSUNITPRICE" NUMBER(12,2),
         "WOITEMSDDATE" DATE CONSTRAINT "NN_T_WOITEMS_WOITEMSDDATE" NOT NULL
ENABLE,
         "WOITEMSCOMMENTS" VARCHAR2(200 BYTE),
         "LABDIPNO" VARCHAR2(100 BYTE),
         "APPROVED" VARCHAR2(1 BYTE),
         "QTYLBS" NUMBER(12,2),
         "SAMPLEORDERREF" VARCHAR2(9 BYTE),
         "SAMPLEBATCHREF" VARCHAR2(50 BYTE),
         "UNITPRICELBS" NUMBER(12,2),
         "ADDYARNPERC" NUMBER(12,2) DEFAULT 0,
         "CLIENTSREF" VARCHAR2(150 BYTE),
         "PROCESSLOSS" NUMBER(12,2) DEFAULT 0,
          CONSTRAINT "PK_T_WOITEMS" PRIMARY KEY ("BASICTYPEID", "WONUMBER",
"WOITEMSL")
 USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
 STORAGE(INITIAL 2097152 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
 TABLESPACE "USERS"  ENABLE,
          CONSTRAINT "FK_T_WOITEMS_T_COUNT" FOREIGN KEY ("COUNTID")
           REFERENCES "DYPROHDNEW"."T_COUNT" ("COUNTID") ENABLE,
          CONSTRAINT "FK_T_WOITEMS_T_YARNTYPE" FOREIGN KEY ("YARNTYPEID")
           REFERENCES "DYPROHDNEW"."T_YARNTYPE" ("YARNTYPEID") ENABLE,
```

```
	CONSTRAINT "FK_T_WOITEMS_T_WORDER" FOREIGN KEY ("BASICTYPEID",
"WONUMBER")
	 REFERENCES "DYPROHDNEW"."T_WORDER" ("BASICTYPEID", "WONUMBER") ENABLE
  );

  CREATE TABLE "DYPROHDNEW"."T_UNITOFMEAS"
  (	"UNITOFMEASID" NUMBER(3,0),
	"UNITOFMEAS" VARCHAR2(20 BYTE) CONSTRAINT
"NN_T_UNITOFMEAS_UNITOFMEAS" NOT NULL ENABLE,
	 CONSTRAINT "U_T_UNITOFMEAS_UNITOFMEAS" UNIQUE ("UNITOFMEAS"));

  CREATE TABLE "DYPROHDNEW"."T_YARNTYPE"
  (	"YARNTYPEID" NUMBER(3,0),
	"YARNTYPE" VARCHAR2(30 BYTE) CONSTRAINT "NN_T_YARNTYPE_YARNTYPE" NOT
NULL ENABLE,
	"INOPERATIONID" NUMBER(1,0) CONSTRAINT "NN_T_YARNTYPE_INOPERATIONID"
NOT NULL ENABLE
);
  CREATE TABLE "DYPROHDNEW"."T_COUNT"
  (	"COUNTID" NUMBER(3,0),
	"COUNTNAME" VARCHAR2(10 BYTE) CONSTRAINT "NN_T_COUNT_COUNTNAME" NOT
NULL ENABLE,
	"TICKET" VARCHAR2(10 BYTE),
	"COUNTLENGTH" NUMBER(10,0),
	"COUNTWEIGHT" NUMBER(12,4),
	"GRAYCOST" NUMBER(12,2),
	"FINISHEDCOST" NUMBER(12,2),
	"INOPERATIONID" NUMBER(1,0) CONSTRAINT "NN_T_COUNT_INOPERATIONID" NOT
NULL ENABLE,
	 CONSTRAINT "U_T_COUNT_COUNTNAME" UNIQUE ("COUNTNAME")
  );

  CREATE TABLE "DYPROHDNEW"."T_CURRENCY"
  (	"CURRENCYID" NUMBER(3,0),
	"CURRENCYNAME" VARCHAR2(15 BYTE) CONSTRAINT
"NN_T_CURRENCY_CURRENCYNAME" NOT NULL ENABLE,
	"CRATE" NUMBER(12,4) CONSTRAINT "NN_T_CURRENCY_CRATE" NOT NULL ENABLE,
	"CRATEUSD" NUMBER(12,4),
	 CONSTRAINT "PK_T_CURRENCY" PRIMARY KEY ("CURRENCYID")
  );

  CREATE TABLE "DYPROHDNEW"."T_CLIENT"
  (	"CLIENTID" NUMBER(4,0),
	"CLIENTNAME" VARCHAR2(75 BYTE) CONSTRAINT "NN_T_CLIENT_CLIENTNAME" NOT
NULL ENABLE,
```

```
	"INOPERATIONID" NUMBER(1,0) CONSTRAINT "NN_T_CLIENT_INOPERATIONID" NOT
NULL ENABLE,
	"CADDRESS" VARCHAR2(250 BYTE),
	"CFACTORY" VARCHAR2(200 BYTE),
	"CTELEPHONE" VARCHAR2(100 BYTE),
	"CFAX" VARCHAR2(50 BYTE),
	"CEMAIL" VARCHAR2(50 BYTE),
	"CTELEX" VARCHAR2(20 BYTE),
	"CURL" VARCHAR2(50 BYTE),
	"CPERSON" VARCHAR2(100 BYTE),
	"BONLYFACTORY" NUMBER(1,0) DEFAULT 0 CONSTRAINT
"NN_T_CLIENT_BONLYFACTORY" NOT NULL ENABLE,
	"REMARKS" VARCHAR2(50 BYTE),
	"ACCCODE" VARCHAR2(20 BYTE),
	 CONSTRAINT "U_T_CLIENT_CLIENTNAME" UNIQUE ("CLIENTNAME")
  );

 CREATE TABLE "DYPROHDNEW"."T_DBATCH"
  (	"DBATCHID" NUMBER(20,0),
	"BASICTYPEID" VARCHAR2(3 BYTE),
	"WONUMBER" NUMBER(6,0),
	"WOITEMSL" NUMBER(3,0),
	"BATCHNO" NUMBER(6,0) CONSTRAINT "NN_T_DBATCH_BATCHNO" NOT NULL
ENABLE,
	"DMACHINENO" NUMBER(6,0) CONSTRAINT "NN_T_DBATCH_DMACHINE" NOT NULL
ENABLE,
	"PLANDATE" DATE CONSTRAINT "NN_T_DBATCH_PLANDATE" NOT NULL ENABLE,
	"BDURATION" NUMBER(6,0) CONSTRAINT "NN_T_DBATCH_BDURATION" NOT NULL
ENABLE,
	"BATCHQTY" NUMBER(12,2),
	"FINISHEDQTY" NUMBER(12,2),
	"BATCHSTATUS" NUMBER(3,0) DEFAULT 0,
	"BATCHRESTATUS" NUMBER(3,0) DEFAULT 0,
	"BATCHPARENT" NUMBER(20,0),
	"BATCHTYPE" NUMBER(3,0) DEFAULT 0,
	"BSTATIONARY" NUMBER(1,0) DEFAULT 0,
	"DESCRIPTION" VARCHAR2(300 BYTE),
	"YARNLOTNO" VARCHAR2(100 BYTE),
	"BATCHAPPSTATUS" NUMBER(2,0) DEFAULT 0,
	"HASDYELINE" NUMBER(1,0) DEFAULT 0,
	"BATCHMPARENTID" NUMBER(20,0),
	"CBATCHTEXT" VARCHAR2(200 BYTE),
	"CMERGETEXT" VARCHAR2(1000 BYTE),
	"MCHILDBATCHIDS" VARCHAR2(200 BYTE),
	"BATCHMSTATUS" NUMBER(2,0) DEFAULT 0,
```

```
        CONSTRAINT "PK_T_DBATCH" PRIMARY KEY ("DBATCHID")
USING INDEX PCTFREE 10 INITRANS 2 MAXTRANS 255 COMPUTE STATISTICS
STORAGE(INITIAL 1048576 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS 2147483645
PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT)
TABLESPACE "USERS"  ENABLE,
        CONSTRAINT "FK_T_DBATCH_T_MACHINES" FOREIGN KEY ("DMACHINENO")
);

CREATE TABLE "DYPROHDNEW"."T_DBATCHTEMP"
(       "DBATCHID" NUMBER(20,0),
        "BASICTYPEID" VARCHAR2(3 BYTE),
        "WONUMBER" NUMBER(6,0),
        "WOITEMSL" NUMBER(3,0),
        "BATCHNO" NUMBER(6,0) CONSTRAINT "NN_T_DBATCHTEMP_BATCHNO" NOT NULL
ENABLE,
        "DMACHINENO" NUMBER(6,0) CONSTRAINT "NN_T_DBATCHTEMP_DMACHINE" NOT
NULL ENABLE,
        "PLANDATE" DATE CONSTRAINT "NN_T_DBATCHTEMP_PLANDATE" NOT NULL
ENABLE,
        "BDURATION" NUMBER(6,0) CONSTRAINT "NN_T_DBATCHTEMP_BDURATION" NOT
NULL ENABLE,
        "BATCHQTY" NUMBER(12,2),
        "FINISHEDQTY" NUMBER(12,2),
        "BATCHSTATUS" NUMBER(3,0) DEFAULT 0,
        "BATCHRESTATUS" NUMBER(3,0) DEFAULT 0,
        "BATCHPARENT" NUMBER(20,0),
        "BATCHTYPE" NUMBER(3,0) DEFAULT 0,
        "BSTATIONARY" NUMBER(1,0) DEFAULT 0,
        "DESCRIPTION" VARCHAR2(300 BYTE),
        "YARNLOTNO" VARCHAR2(100 BYTE),
        CONSTRAINT "PK_T_DBATCHTEMP" PRIMARY KEY ("DBATCHID")   );.

CREATE TABLE "DYPROHDNEW"."T_DINVOICE"
(       "BASICTYPEID" VARCHAR2(3 BYTE),
        "DINVOICENO" NUMBER(6,0),
        "DINVOICEDATE" DATE CONSTRAINT "NN_T_DINVOICE_DINVOICEDATE" NOT NULL
ENABLE,
        "CLIENTID" NUMBER(4,0) CONSTRAINT "NN_T_DINVOICE_CLIENTID" NOT NULL
ENABLE,
        "GATEPASSNO" NUMBER(6,0),
        "EFFECTIVEDDATE" DATE,
        "DINVOICECOMMENTS" VARCHAR2(200 BYTE),
        "LCNO" VARCHAR2(100 BYTE),
        "PINO" VARCHAR2(100 BYTE),
        CONSTRAINT "PK_T_DINVOICE" PRIMARY KEY ("BASICTYPEID", "DINVOICENO"));
```

```
CREATE TABLE "DYPROHDNEW"."T_DINVOICEITEMS"
(	"BASICTYPEID" VARCHAR2(3 BYTE),
	"DINVOICENO" NUMBER(6,0),
	"DITEMSL" NUMBER(3,0),
	"WBASICTYPEID" VARCHAR2(3 BYTE) CONSTRAINT
"NN_T_DINVOICEITEMS_BASICTYPEID" NOT NULL ENABLE,
	"WONUMBER" NUMBER(6,0) CONSTRAINT "NN_T_DINVOICEITEMS_WONUMBER" NOT
NULL ENABLE,
	"WOITEMSL" NUMBER(3,0) CONSTRAINT "NN_T_DINVOICEITEMS_WOITEMSL" NOT
NULL ENABLE,
	"DITEMSQTY" NUMBER(12,2) CONSTRAINT "NN_T_DINVOICEITEMS_DITEMSQTY" NOT
NULL ENABLE,
	"DITEMSQTYRET" NUMBER(12,2) DEFAULT 0,
	"DITEMSCTN" NUMBER(10,0) CONSTRAINT "NN_T_DINVOICEITEMS_DITEMSCTN" NOT
NULL ENABLE,
	"DBATCHNO" VARCHAR2(50 BYTE),
	"DITEMSCOMMENTS" VARCHAR2(100 BYTE),
	 CONSTRAINT "PK_T_DINVOICEITEMS" PRIMARY KEY ("BASICTYPEID", "DINVOICENO",
"DITEMSL")
);
```