**Chapter 1**

**INTRODUCTION**

ERMS means Enterprise Resources Management System, a system that can manage all the resources of an enterprise, manage the information from different teams of a company and helps in proper gathering of complete information about the current state of the system. ERMS can be used for business management as it can manage more than one resource such as employees, clients and vendors. It has a unified system for management of products, bills, workforce, stocks, inventories and the business can predict the risks and challenges using ERMS modules. Most ventures which are successful has an ERMS software running in the background.

Without an ERMS framework, a business body will have a hard time making business decisions which may not have a real impact on his business because the data is not centralized and often leads to harm in business operations. Always an ERMS has been highly beneficial due to the following reasons:

* Reuniting various practical areas to guarantee appropriate correspondence, profitability and effectiveness.
* Customer feedbacks valuation to keep a track of the likeness of a product. This helps the enterprise to know which product the users prefer.
* Product sales and commission information can be known from the ERMS.
* Keeps synchronized between stock in hand and sale of stock.
* ERMS can adapt to any processor easily, so the enterprise need to only use the existing devices for implementing ERMS.
* Provides extra protection as authorization is also included in ERMS so that only right users access those modules.
* Data security and data integrity can be ensured so that no other member modifies the data and cause loss to the business.
* Provides easy control of invoicing and payment handling and in this manner boosting their profitability in operations.
* Reduce paper records through digitization of business data.
* Greater precision with charts that tracks down loss of business and helps to recover the losses immediately.
* Improves Universal operations by providing global expense structures, invoicing plans, various monetary forms, global balance sheets.

An ERMS has modules for financial administration, production, supplier and buyer administration, stock and material administration.

1. **Finance administration-** Using ERMS, organizations can manage the accounts receivable and accounts payable of the organization with significant advantages such as accuracy and accountability over redundant bookkeeping procedures, for example company budget plan creation, distribution and administration; income monitoring; creditor liabilities and receivable and revenue reporting are part of finance administration.
2. **Production-** It shows production plan, resources available for production and management of production with optimize production methodology that follows quality standards. With ERMS and its assembling capacities, organizations are engaged to make the most ideal item in the most productive and practical way.
3. **Supplier and Buyer Administration-** An ERMS framework suite incorporates stock management, request administration, acquiring and obtainment, product logistics, inventory network management, product returns management to help organizations facilitate and control their supply chain network. With ERMS framework, organizations can better deal with all production network operations, including the sourcing, obtaining, the booking and administration of the warehouse in supplying the products to the buyers on time.
4. **Stock and Material Administration Module-** Stock Control Module contains the following units such as stock entries with the quantity, name and price of the product. When this information is fed into the ERMS framework, stock is calculated for each product, the stock management team can have a better assurance of the stock in hand and get alerted if a product goes out of stock, the company can acquire the stock as per the need of the customers.

An ERMS model that has resource management modules, customer relationship management modules, feedback modules, purchase order modules, client modules, inventory modules, production and supply modules, finance and accounting modules, employee modules and user modules.

Modules store data in a centralized database which helps to gain business intelligence by analysing raw data as well as perform predictive analytics and data mining.

A Proposed ERMS works on all systems and devices. ERMS possess information from different fields and required information can be passed to other departments through the same system, a fine example is order purchase information passed to the product delivery team. ERMS is the most important application to any enterprise to coordinate different business operations and enhance profitability. An ERMS is used for a specific end goal to expand production, lessen costs and optimize business for profits.

**Chapter 2**

**LITERATURE SURVEY**

**2.1 Existing System**

With the manual system, the records of Stocks, Purchase Order, Suppliers and Make are maintained in a File. Most of the present systems have very few modules such as the billing system, the product information system and the customer membership system. Many tasks are not digitalized in existing systems and are recorded in register books manually by the manager. Often the collected data are affected by human errors and cause wrong stock details of products which can affect the business and the customers. Customer Resource Management (CRM) system helps to manage the customers and their preferences. Feedback system collects the feedback of clients. Customers are very important for the success of business. These modules are absent in the existing systems.

**Advantages**

* Manual Methods do not require Internet and Advanced Technology.
* Method is inexpensive and cost effective.

**Disadvantages**

* There are many problems in managing the inventory records. The records may not be kept safe and their might be a chance of records being replaced or misplaced.
* The records might not be reliable.
* Manual records are prone to have many errors which may result in major problems.

**2.2 Technical System**

The technical based system consists of a database which holds on the data of the respective fields. Each time the data is entered it is automatically added into the database. It has an ability to keep a track of items received and sent.

**Advantages**

* With the technical system the data can be kept secured using the necessary measures.
* The data cannot be altered by any user. Only admin has the privilege to do it.
* Data entered into the system can be reliable.
* Data management in the technical System can be faster and efficient.
* The System Software can be customized according to the User's Requirement.
* Data from the previous entries can be obtained easily.

**Disadvantages**

* This System requires Electricity consumption and may require additional components as per the user requirement.
* The System cost may be priced high.
* This System may lead to technical error and failure which may result into temporary breakdown of the system.
* Care must be taken so as to avoid the data to be reached into wrong Hands.

**2.3 Proposed System**

Various software are already available in the market and online which deals with Enterprise Resource Management. But this proposed system is developed according to the requirements provided by the clients. It consists of various fields integrated according to the system architecture. Due to effective passing of data between the ERMS modules the proposed system does the tasks in the best possible manner, as a result managing of the enterprise of the enterprise will be easier, efficient and is far better than the existing system.

[1] Athul Jayaram, Swati Singal, et al., “An Enterprise Resource Management Model for Business Intelligence, Data Mining and Predictive Analytics”, IEEE, 2017.

Their model consist of ERMS that has specialized modules for employees, clients, accounting, products, payrolls and inventory. The dashboard of the ERMS, where it shows the statistical data of the enterprise is in graphical form which shows new orders, total revenue, total cost, total profit as well as a graph that shows number of sales in each month. Each of the modules has submodules for inserting, updating, deleting data records with live data search and data sort feature.

[2] Yan XU, Nasrin Rahmati, and Vincent C S Lee Clayton School of IT, Monash University, Australia, et al., “A Review of Literature on Enterprise Resource Planning Systems”, 2008 , IEEE.

Their survey was based on a study of journals and conference proceeding papers. They concentrated on those articles which main theme is concerning ERP systems. The full text of each article was reviewed in order to eliminate those articles that were not really related to ERP systems. Besides case study method, there are some other methods are being used to do social science research, such as surveys, experiments, histories, and analysis of archival information. Depending on Yin’s research, generally speaking, case study is the preferred strategy by most researchers when ‘how’ and ‘why’ research questions are being posed, when the contemporary events are being examined, and when the investigator has little or no control over the events. There are two types of case studies, single-case study and multiple-case study.

“The essence of a case study, the central tendency among all types of case study, is that it tries to illuminate a decision or set of decisions: why they were taken, how they were implemented, and with what result.”(Yin) according to this definition, case studies are good strategy for the research topics such as “ERP selection”, “ERP implementation”, and “ERP post-implementation”.

**Features**

* This System can generate Purchase Order for the required materials.
* It can keep a track on the current stock in the warehouse.
* Keeps a track of the Gate Pass material (sent for the outsourcing work).
* Keeps a track on the pending items to be received.
* Implements billing of the products according to the current billing system.
* This system can generate date wise Stock Reports.
* It provides Supplier Wise details of the stocks received.
* Generates Job Order report.
* Can generate reports for stock, goods and supplier.
* Implements predictive analysis of the stock management.

**Chapter 3**

**TECHNOLOGIES USED**

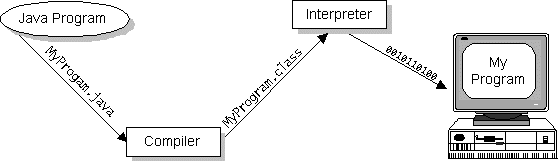
**3.1 Java**

Java is a general-purpose computer-programming language that is [concurrent](https://en.wikipedia.org/wiki/Concurrent_computing), [class-based](https://en.wikipedia.org/wiki/Class-based_programming), [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming), and specifically designed to have as few implementation dependencies as possible. It is intended to let application developers "[write once, run anywhere](https://en.wikipedia.org/wiki/Write_once,_run_anywhere)" (WORA), meaning that [compiled](https://en.wikipedia.org/wiki/Compiler) Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to [bytecode](https://en.wikipedia.org/wiki/Java_bytecode) that can run on any [Java virtual machine](https://en.wikipedia.org/wiki/Java_virtual_machine) (JVM) regardless of [computer architecture](https://en.wikipedia.org/wiki/Computer_architecture). As of 2016, Java is one of the most [popular programming languages in use](https://en.wikipedia.org/wiki/Measuring_programming_language_popularity), particularly for client-server web applications, with a reported 9 million developers. Java was originally developed by [James Gosling](https://en.wikipedia.org/wiki/James_Gosling) at [Sun Microsystems](https://en.wikipedia.org/wiki/Sun_Microsystems)(which has since been [acquired by Oracle Corporation](https://en.wikipedia.org/wiki/Sun_acquisition_by_Oracle)) and released in 1995 as a core component of Sun Microsystem's [Java platform](https://en.wikipedia.org/wiki/Java_(software_platform)). The language derives much of its [syntax](https://en.wikipedia.org/wiki/Syntax_(programming_languages)) from [C](https://en.wikipedia.org/wiki/C_(programming_language)) and [C++](https://en.wikipedia.org/wiki/C%2B%2B), but it has fewer [low-level](https://en.wikipedia.org/wiki/Low-level_programming_language) facilities than either of them.

Java is a high-level programming language which has all of the following features:

* Simple
* Object-oriented
* Distributed
* Interpreted
* Robust
* Secure
* Architecture-neutral
* Portable
* High-performance
* Multithreaded
* Dynamic

Java is also unusual in that each Java program is both compiled and interpreted. With a compiler, you translate a Java program into an intermediate language called Java byte codes--the platform-independent codes interpreted by the Java interpreter. With an interpreter, each Java byte code instruction is parsed and run on the computer. Compilation happens just once; interpretation occurs each time the program is executed. This figure illustrates how this works.



Java byte codes can be considered as the machine code instructions for the Java Virtual Machine (Java VM). Every Java interpreter, whether it's a Java development tool or a Web browser that can run Java applets, is an implementation of the Java VM. The Java VM can also be implemented in hardware.

Java byte codes help make "write once, run anywhere" possible. The Java program can be compiled into byte codes on any platform that has a Java compiler. The byte codes can then be run on any implementation of the Java VM. For example, the same Java program can run on Windows NT, Solaris, and Macintosh.

**3.1.1 Java Libraries**

The Java Class Library is a set of [dynamically loadable libraries](http://en.wikipedia.org/wiki/Library_%28computer_science%29#Dynamic_linking) that [Java](http://en.wikipedia.org/wiki/Java_%28programming_language%29) applications can call at [run time](http://en.wikipedia.org/wiki/Run_time_%28program_lifecycle_phase%29). Because the [Java Platform](http://en.wikipedia.org/wiki/Java_Platform) is not dependent on any specific operating system, applications cannot rely on any of the existing libraries. Instead, the Java Platform provides a comprehensive set of [standard class libraries](http://en.wikipedia.org/wiki/Standard_library), containing much of the same reusable functions commonly found in modern operating systems.

The Java class libraries serve three purposes within the Java Platform:

Like other [standard code libraries](http://en.wikipedia.org/wiki/Standard_library), they provide the programmer a well-known set of useful facilities, such as [container classes](http://en.wikipedia.org/wiki/Collection_class) and [regular expressions](http://en.wikipedia.org/wiki/Regular_expression).

* In addition, the class libraries provide an abstract interface to tasks that would normally depend heavily on the hardware and operating system. Tasks such as [network](http://en.wikipedia.org/wiki/Computer_networking) access and [file](http://en.wikipedia.org/wiki/Computer_file) access are often heavily dependent on the native capabilities of the platform.
* Finally, some underlying platforms may not support all of the features a Java application expects. In these cases, the class libraries can either emulate those features using whatever is available, or provide a consistent way to check for the presence of a specific feature.

**3.2 Swing (Java)**

Swing is a [GUI](https://en.wikipedia.org/wiki/Graphical_user_interface) [widget toolkit](https://en.wikipedia.org/wiki/Widget_toolkit) for [Java](https://en.wikipedia.org/wiki/Java_(programming_language)). It is part of [Oracle](https://en.wikipedia.org/wiki/Oracle_Corporation)'s [Java Foundation Classes](https://en.wikipedia.org/wiki/Java_Foundation_Classes) (JFC) – an [API](https://en.wikipedia.org/wiki/Application_programming_interface) for providing a [graphical user interface](https://en.wikipedia.org/wiki/Graphical_user_interface) (GUI) for Java programs.

Swing was developed to provide a more sophisticated set of GUI [components](https://en.wikipedia.org/wiki/Software_component) than the earlier [Abstract Window Toolkit (AWT)](https://en.wikipedia.org/wiki/Abstract_Window_Toolkit). Swing provides a [look and feel](https://en.wikipedia.org/wiki/Look_and_feel) that emulates the look and feel of several platforms, and also supports a [pluggable look and feel](https://en.wikipedia.org/wiki/Pluggable_look_and_feel) that allows applications to have a look and feel unrelated to the underlying platform. It has more powerful and flexible components than AWT. In addition to familiar components such as buttons, check boxes and labels, Swing provides several advanced components such as tabbed panel, scroll panes, trees, tables, and lists.

Unlike AWT components, Swing components are not implemented by platform-specific code. Instead, they are written entirely in Java and therefore are platform-independent. The term "lightweight" is used to describe such an element.

Swing is a platform-independent, "[model-view-controller](https://en.wikipedia.org/wiki/Model-view-controller)" [GUI](https://en.wikipedia.org/wiki/GUI) framework for Java, which follows a single-[threaded](https://en.wikipedia.org/wiki/Thread_(computing)) programming model. Additionally, this framework provides a layer of abstraction between the code structure and graphic presentation of a Swing-based GUI.

**3.3 JDBC:**

In an effort to set an independent database standard API for Java, Sun Microsystems developed Java Database Connectivity, or JDBC. JDBC offers a generic SQL database access mechanism that provides a consistent interface to a variety of RDBMS. This consistent interface is achieved through the use of “plug-in” database connectivity modules, or *drivers*. If a database vendor wishes to have JDBC support, he or she must provide the driver for each platform that the database and Java run on.

To gain a wider acceptance of JDBC, Sun based JDBC’s framework on ODBC. As you discovered earlier in this chapter, ODBC has widespread support on a variety of platforms. Basing JDBC on ODBC will allow vendors to bring JDBC drivers to market much faster than developing a completely new connectivity solution.

### 3.3.1 JDBC Goals:

Few software packages are designed without goals in mind. JDBC is one that, because of its many goals, drove the development of the API. These goals, in conjunction with early reviewer feedback, have finalized the JDBC class library into a solid framework for building database applications in Java.

The goals that were set for JDBC are important. They will give you some insight as to why certain classes and functionalities behave the way they do. The eight design goals for JDBC are as follows:

* **Provide a Java interface that is consistent with the rest of the Java system**  
  Because of Java’s acceptance in the user community thus far, the designers feel that they should not stray from the current design of the core Java system.
* **Keep it simple**  
  This goal probably appears in all software design goal listings. JDBC is no exception. Sun felt that the design of JDBC should be very simple, allowing for only one method of completing a task per mechanism. Allowing duplicate functionality only serves to confuse the users of the API.
* **Use strong, static typing wherever possible**  
  Strong typing allows for more error checking to be done at compile time; also, less errors appear at runtime.

**3.4 SQL Server 2005**

It is powerful RDBMS software developed by Microsoft Corporation. Various DBMS software can be used as back end for VB without any external support. Database design development is very easy in SQL Server Express 2005 as it provides GUI based environment. Backup and restoring is easy in SQL Server, since the whole database is stored in single database.

A data base is simply a collection of useful data. SQL Server is a fully featured database management system lets you collect, organize, find, display and print information about your personal and business life or organization. SQL Server comes with wizards and variety of pre-defined data elements that lets you keep track of some common information with only a few clicks of your mouse. SQL Server Express is Relational Database, which means that it allows data that has been stored in different places to be linked.

SQL Server Express 2005 helps to manage information in three important ways.

* Reduces redundancy.
* Facilitate the sharing of information.
* Keeps the data accurate.

SQL Server Express 2005 can help to create

* + - Tables
    - Queries
    - Reports
    - Controls
    - Backup / Restore

**DATABASE:**

A database is simply a collection of useful data.

**TABLES:**

In SQL Server tables are similar collection of data. There may be many tables in the database, all these tables would be organized differently and contain mostly different information but they all be in the same database file and may relate to one another. These tables are stored in the same database file because they are often used together to create reports.

**RECORDS:**

A record is all the information contained in one row of an SQL Server database table.

**FIELDS:**

Fields are places in the table where you store individual chunks of information. SQL Server users key fields and indexing to help many database operation. You can tell SQL Server which fields should be key fields.

**QUERIES:**

Queries are requested to access the information and respond to users request and it is called as dynast.

**FEATURES OF SQL SERVER EXPRESS 2005:**

* GUI based environment.
* Supports almost all type of SQL statements.
* Data security by providing encryption/ password.
* Very easy to understand and work with.
* Provides various properties for tables to improve performances.
* Provides various objects such as tables, queries and reports.

**Chapter 4**

**REQUIREMENTS SPECIFICATION**

* 1. **Functional Requirements:-**

Functional requirements are the statements of the services that the system should provide, how the system should react to particular inputs and how the system should behave in particular situations .It depends on the type of the software, expected users and the type of system where the software is used

The functional requirements of the software are:

* **USER REGISTRATION :**

This module helps the user to register, or the admin to register on to the application. Where, then the details of the user/admin is recorded and whenever new stock is ordered it adds up to previous stock

* **USER LOGIN:**

The user and the admin can only login if the registered password and the name is entered properly and then it verifies. If matched only then the user can access to the services.

* **SEARCH:**

With the help of the reference number or the name of the company or the user particular information about the stock purchased by the user can be known. For many other functions also we search for particular product and do the required search.

* **UPDATE:**

With the help of the update if the stock order is entered less or more with the update changes can be made. Automatically, the stock is updated

* **DELETE**

If the user wants to delete the order made by him then he can delete the order. By cancelling the order being placed by him with the function present.

* **INSERT**

The order that is to be placed by the vendor may be to the supplier with all the required information is added to the fields and whenever required it is being inserted. The insertion of the values and the stock is being updated accordingly.

The functional requirements include the **raw material management** as follows:-

* Raw material manufacturers and suppliers definition.
* List of the warehouse for storage.
* Control of orders and received items.
* Easily locate inventory items with user-defined items.
* Consumable supplying for calibrations and maintenances.
* Management of supplies assigned to product development.
* Costs control.
* Occurrence management.
* Association of multiple files controlled documents.

And **warehouse control** as follows:-

* Levels definition access per user.
* Events control (entries and exits).
* Definition of minimum stock replacement points.
* Stock level alerts.
* Manage multiple inventory locations.
* Support integration with internal systems.

**4.2 Non Functional Requirements:-**

Non-functional requirements are the constraints on the services or functions offered by the system such as the timing constraints, constraints on the development procedure, standards etc.

Non-functional requirements are called as the “quality attributes” of the system

Non-functional requirements are classified into 2 types

1. Execution qualities that are usability and security that are executable at run time
2. Evolution qualities such as testability, maintainability, extensibility and scalability which are embodied in the static structure of the software system.

* **PERFORMANCE**

Performance is the testing process that is involved as to how fast the system works under a particular workload. It serves different purposes like it can meet particular requirement. Example: providing the particular information of the user stock purchased and the available stock in the inventory in an effective way.

* **RESPONSIVENESS**

This includes the response of the system at any external input provided by the user. Despite, of all the work it is currently working on.

* **SCALABILITY**

This includes the architecture that can grow variably with the growing traffic and the user base. It helps the user experience remain the same irrespective of the number of users

Example: there may be admin and the manager and the user using the application. It should not affect the system experience

* **USABILITY**

The user is the simple user with only the requirements for his company. The user should be familiar with the software where he can use the software without any hindrance.

* **RELIABILITY**

The amount of time required that the system will not fail during the working. It is “probability of success with failure” in the working mode of the system

* **SECURITY**

It is the use of hardware, software and the functions that helps us to protect the system from the external threats. It is process of protecting the data modification, unauthorized access

Example: there is an active user in the master column which helps to operate at the time of activity

* **MAINTAINAIBILITY**

The ease of use and the fixing of the bugs of the system must of prior importance. Maintenance should cost us less.

**4.3 System Requirements**

A system requirement is a detailed description of the system services and a structured document setting.

**4.3.1 Hardware Requirements**:

* Processor 1.7 GHz or more.
* RAM min 1 GB or more
* Hard Disk min 5GB free or more.

**4.3.2 Software** **Requirements**:

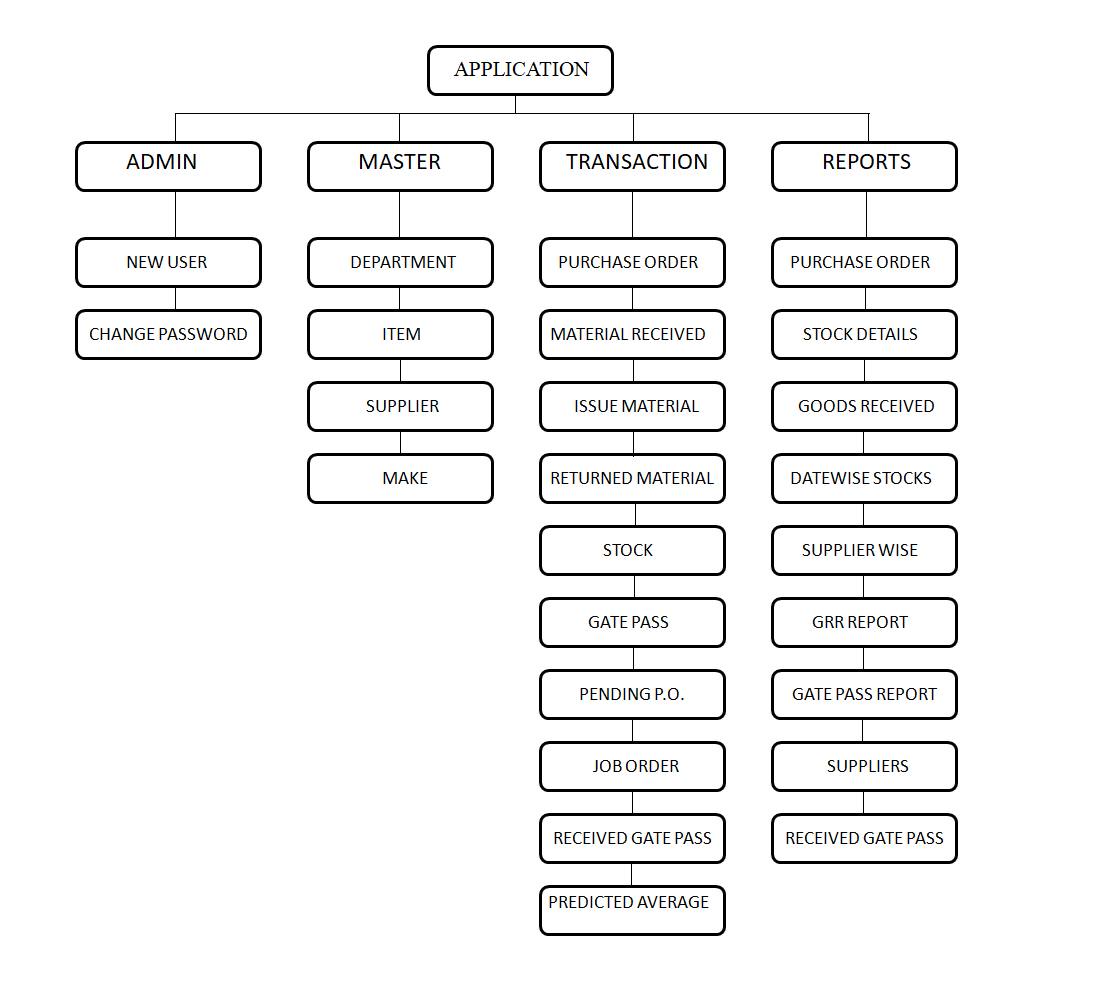
* Front End Java Swings
* Back End SQL Server 2005
* IDE NetBeans
* Generating reports Jasper

**Chapter 5**

**SYSTEM DESIGN AND IMPLEMENTATION**

**5.1 System Architecture**

The Fig 5.1 shows the system architecture of the inventory management application.

****

The application consists of four important modules:

* **Admin Module**

The admin of the application can create new users or change the login password for accessing the application. These features are under the Admin section of the application.

* **Master Module**

The master module allows the admin of the application to add the various items, departments, suppliers and make of various items into the database. There are options to also modify, delete and view the data from the database.

* **Transaction Module**

This module does most of the essential tasks required. The transaction module provides various important services like generating purchase orders, issuing gate passes, updating stocks and keeping track of the purchase orders.

* **Reports Module**

The user of the application can directly generate existing reports using the services of this module. It also provides options to generate all the supplier details and the entire stock of the inventory.

**5.2 General working of the application**

Initially the admin/store manager has to feed all the items, departments, make and suppliers into the database. This is a onetime task except for if there are some variations in the attributes of the various data.

The following steps describe the general working of the application system:

1. First the login page is initialized and the user has to enter his credentials to gain access to the application.
2. The admin then studies the requirements of the enterprise and understands various tasks required to perform.
3. If there is requirement of any raw materials then the admin can generate the purchase order in accordance to the supplier’s quotation.
4. Other tasks to be performed like adding stocks, issuing materials to various departments of the industry or generating gate passes are carried out if required.

**5.3 Use Case Diagram**

A use case diagram is a representation of a user’s interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different uses cases and will often be accompanied by other types of diagrams as well.

The purpose of the use case diagrams is simply to provide the high level view of the system and convey the requirements in terms of stakeholders.

The storekeeper in charge performs the following actions:

* **View the stock**

The storekeeper views and tracks the stock levels of various items required.

* **Study requirements**

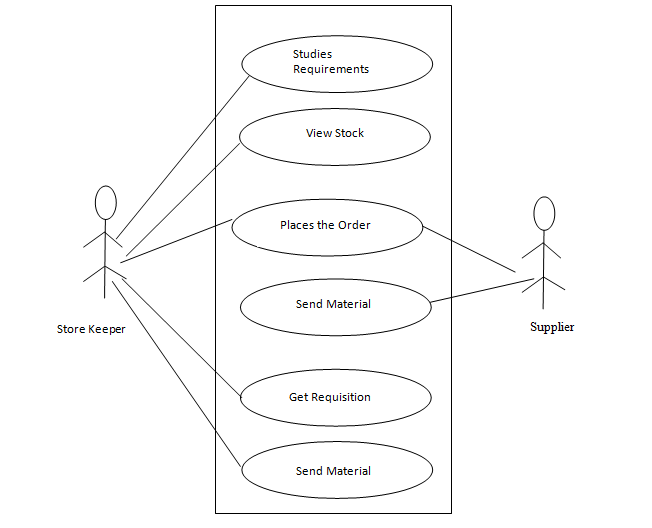
All the requirements are studied by the storekeeper to perform the required daily activities.

* **Placing Order**

The purchase orders are generated by the storekeeper and sent to the respective supplier.

* **Get Requisition**

The storekeeper receives requests for material or items from the internal departments of the industry for manufacturing.



**Fig 5.3 Use Case diagram for Inventory Management application**

**5.4 E-R Diagrams**

E-R Diagram is a visual representation of data that describes how data is related to each other. The E-R model defines the conceptual view of a database. It works around real-world entities and the associations among them. At view level, the E-R model is considered a good option for designing databases.

The components of an E-R Diagram are:

* **Entity**

An Entity can be any object, place, person or class. In E-R diagram, an entity is represented using rectangles.

**Weak Entity:** An Entity that depends on another entity. They don’t have key attribute of their own. Double rectangle represents weak entity.

* **Attributes**

An attribute describes a property or characteristic of an entity. For example, Name, Age, Address etc. can be attributes of a Student. An attribute is represented using ellipse.

**Key attributes:** They represent the main characteristic of an Entity. It is used to represent primary key. Ellipses with underlying lines represent key attribute.

**Composite attributes:** An attribute also have their own attributes. These attributes are known as composite attributes.

* **Relationship**

A relationship describes relations between entities. Relationship is represented using diamonds.

There are three types of relationship that exist between entities.

* **Binary Relationship**

A relationship where two entities are participating is called a binary relationship. Cardinality is the number of instance of an entity from a relation that can be associated with the relation.

It is further divided into 4 types:

* One-One Binary Relationship
* One-Many Binary Relationship
* Many-One Binary Relationship
* Many-Many Binary Relationship
* **Recursive Relationship**

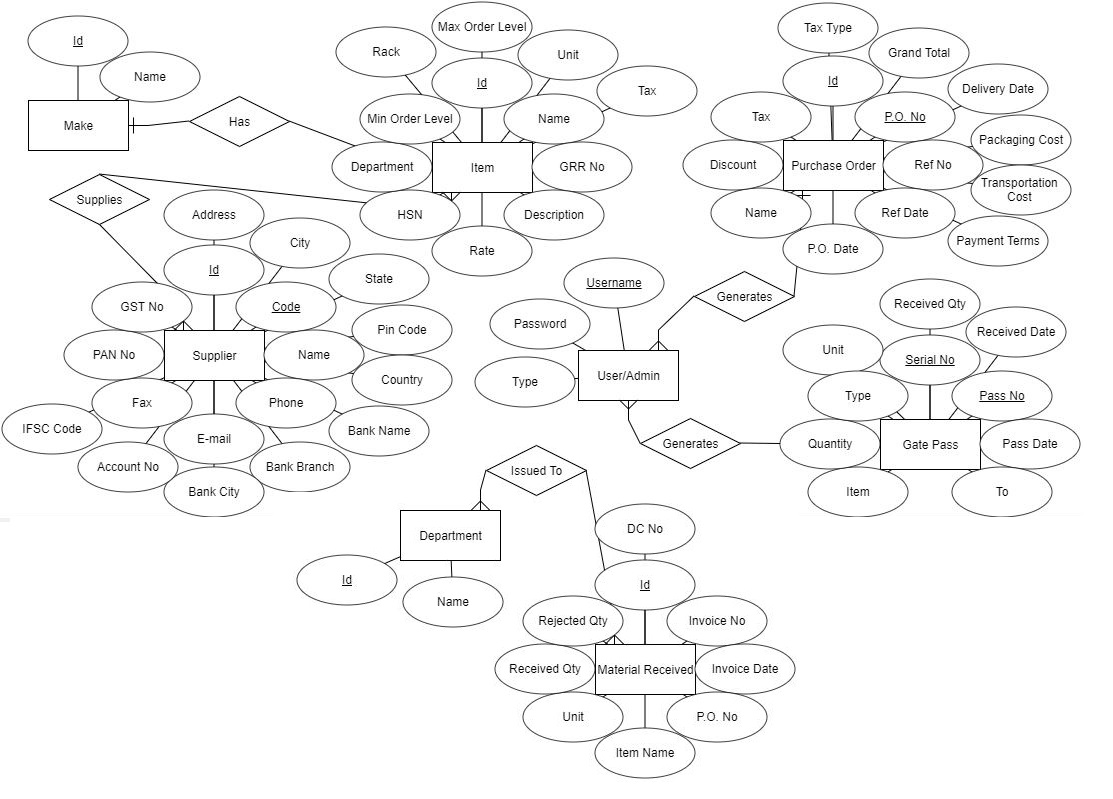
When an Entity is related with itself it is known as Recursive Relationship.

* **Ternary Relationship**

Relationship of degree three is called as Ternary Relationship.

Entities and attributes of inventory management application are described in the table below:

|  |  |
| --- | --- |
| **Entity** | **Attributes** |
| User | * Username * Password * Member Type |
| Department | * Id * Name |
| Make | * Id * Name |
| Item | * Id * Name * GRR No * Description * Rate * HSN * Department * Min Order Level * Max Order Level * Unit * Tax * Rack |
| Supplier | * Id * Code * Name * Phone * E-mail * Fax * PAN No * GST No * Address * City * State * Pin Code * Country * Bank Name * Bank Branch * Bank City * Account No * IFSC Code |
| Purchase Order | * Id * P.O. No * Reference No * Reference Date * P.O. Date * Name * Discount * Tax * Tax Type * Grand Total * Delivery Date * Packaging Cost * Transportation Cost * Payment Terms * GRR No |
| Issue Material | * Serial No * From * To * Requisition No * Requisition Date * Item Name * Issued Quantity * Remarks |
| Gate Pass | * Serial No * Pass No * Pass Date * To * Unit * Item * Quantity * Type * Received * Received Quantity * Received Date |
| Material Received | * Id * Invoice No * Invoice Date * P.O. No * Item Name * Unit * Received Quantity * Rejected Quantity * DC No |



**Fig 5.4 E-R Diagram for Inventory Management application**

**5.5 Database Design**

Database design is the process of producing a detailed data model of a database. This data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. Database design can also be thought of as the logical design of the base data structures used to store the data.

**5.6.1 Database Schema**

A database schema is the skeleton structure that represents the logical view of the entire database. It defines how the data is organized and how relations among them are associated. It formulates all the constraints that are to be applied on the data.

A database schema defines it entities and the relationship among them. It contains a descriptive detail of the database, which can be depicted by means of schema diagrams. It’s the database designers who design the schema to help programmers understand the database and make it useful.

A database schema can be divided broadly into two categories:

* **Physical Database Schema**

Schema pertains to the actual storage of the data and its form of storage like files, indices etc. It defines how the data will be stored in a secondary storage.

* **Logical Database Schema**

Schema defines all the logical constraints that need to be applied on the data stored. It defines tables, views and integrity constraints.

* **Entity-Set and Keys**

Key is an attribute or collection of attributes that uniquely identifies an entity among entity set. For example, the item name of a item makes it identifiable among all items.

* **Super Key**

A set of attributes that collectively identifies an entity in an entity set.

* **Candidate Key**

A minimal super key is called a candidate key. An entity set can contain more than one candidate key.

* **Primary Key**

A primary key is one of the candidate keys chosen by the database designer to uniquely identify the entity set.

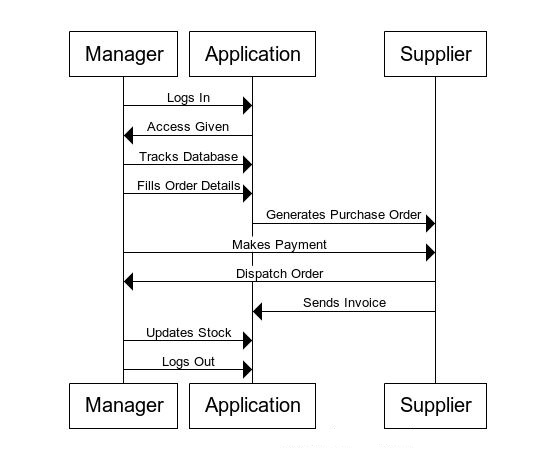
* **Foreign Key**

A foreign key is a field in one table that uniquely identifies a row of another table or the same table.

**5.6 Sequence Diagrams**

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.

A sequence diagram shows, as parallel vertical lines, different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

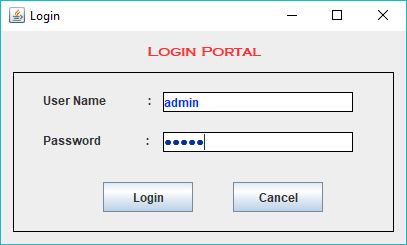


**Fig 5.5 Sequence Diagram for Inventory Management application**

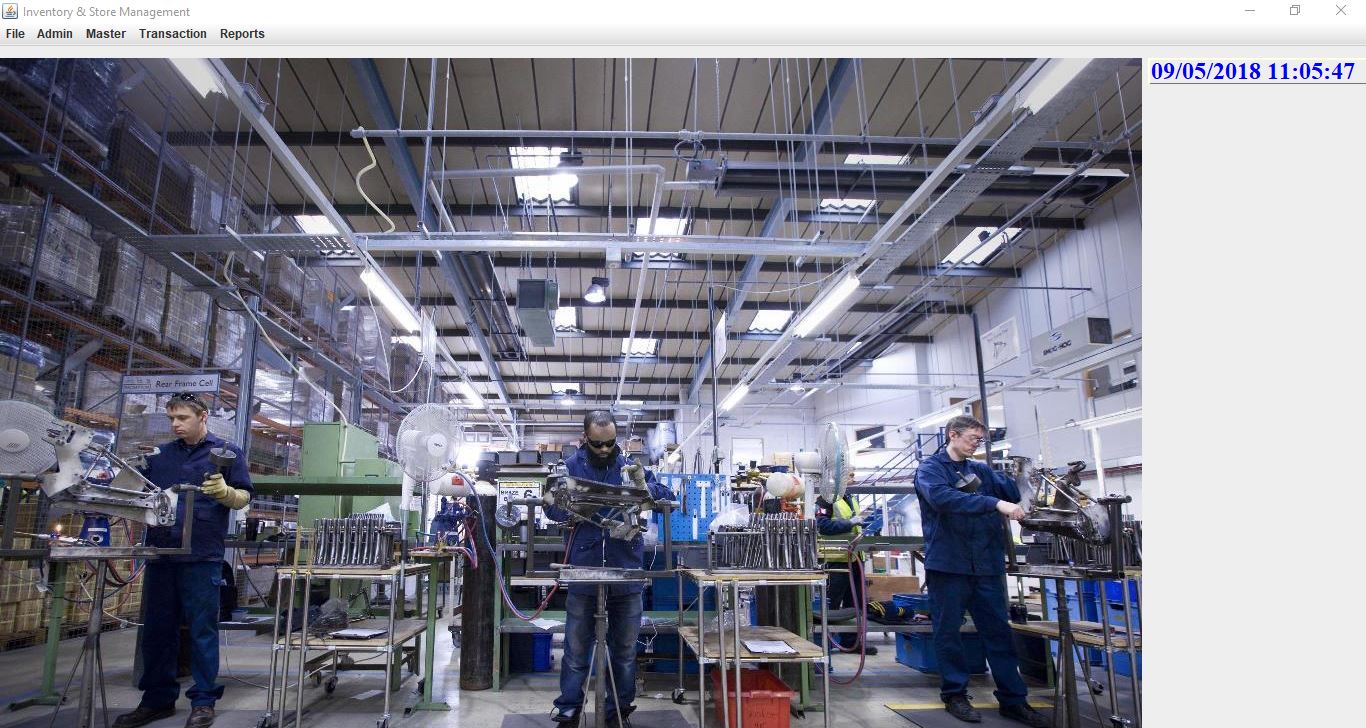
**Chapter 6**

**SNAPSHOTS**

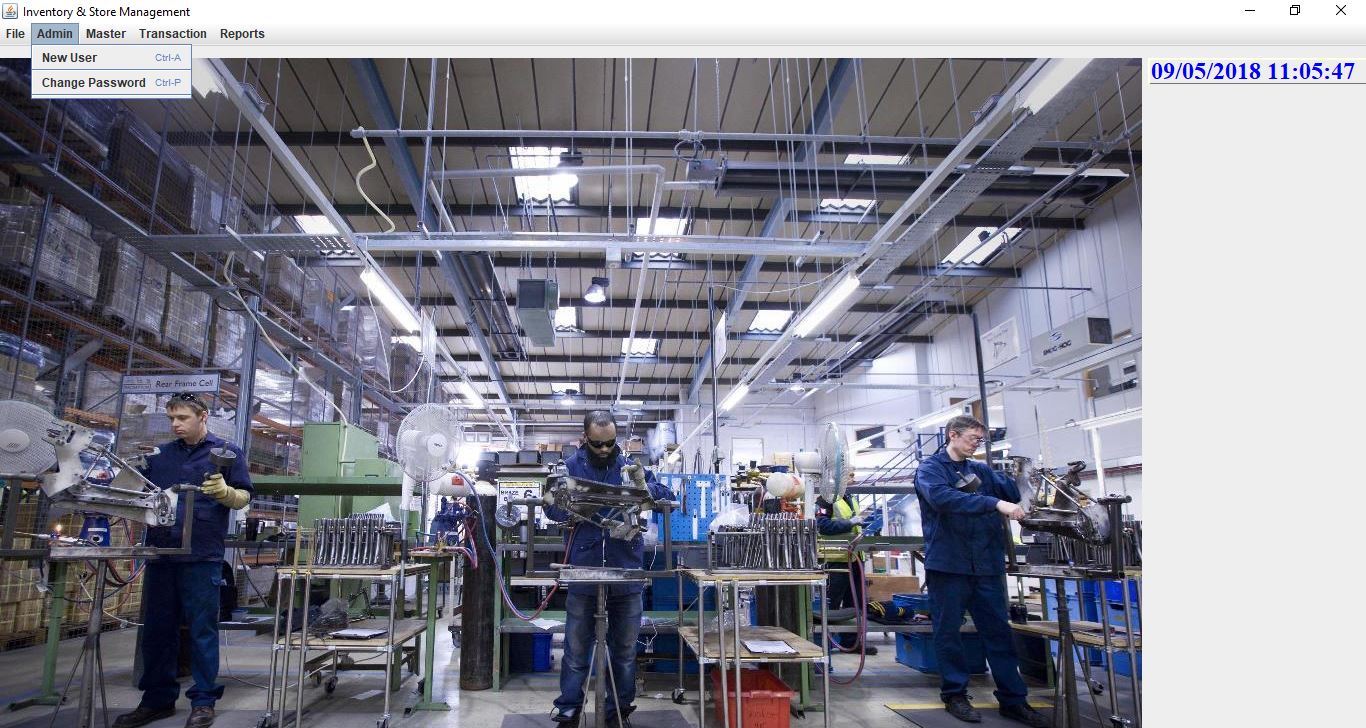
1. When user opens the application he has to login to gain access.



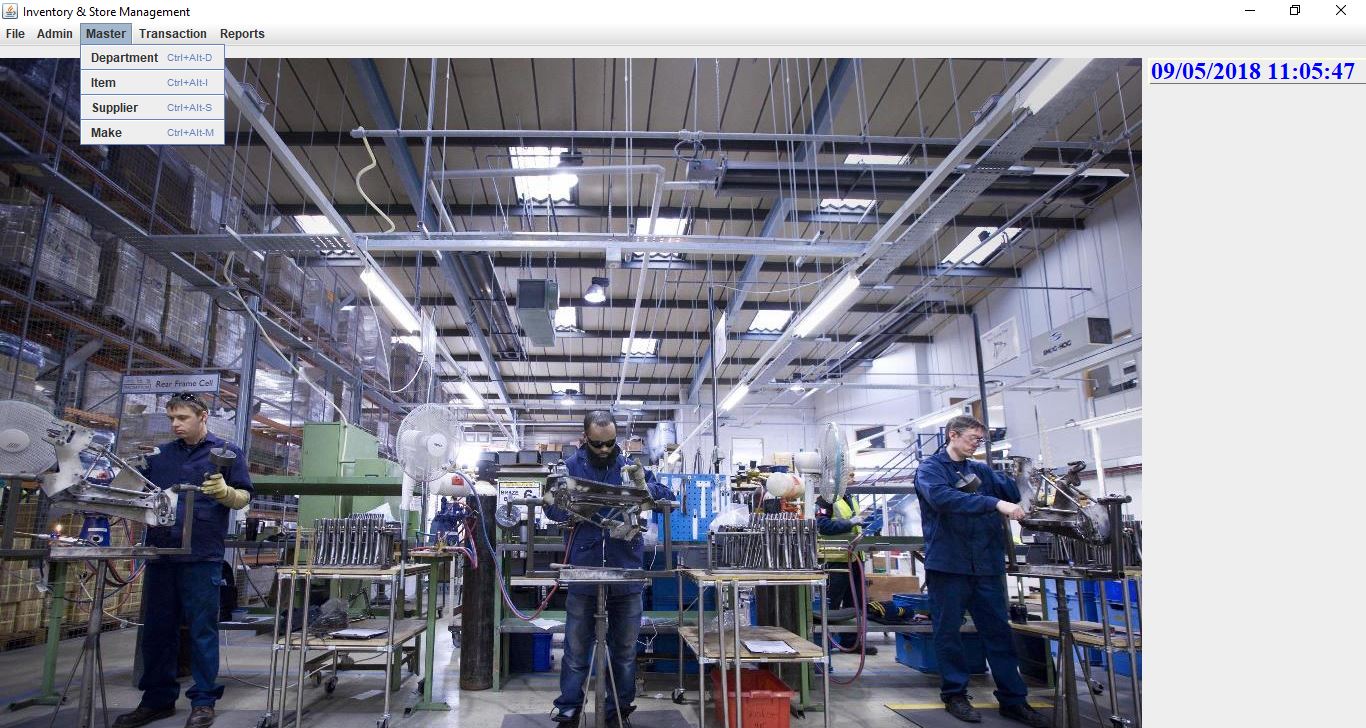
1. The user sees the below image after logging in.



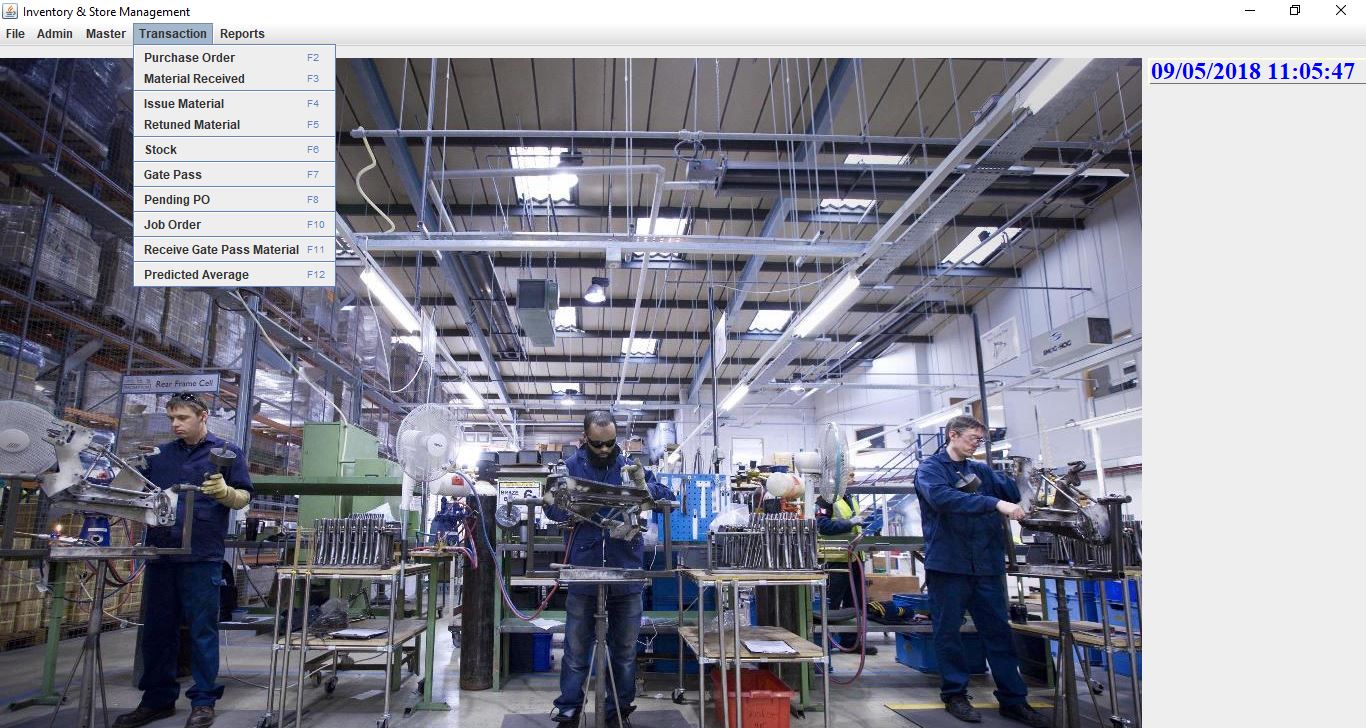
1. The first menu option is Admin and allows the user to create a new user or change the current password.



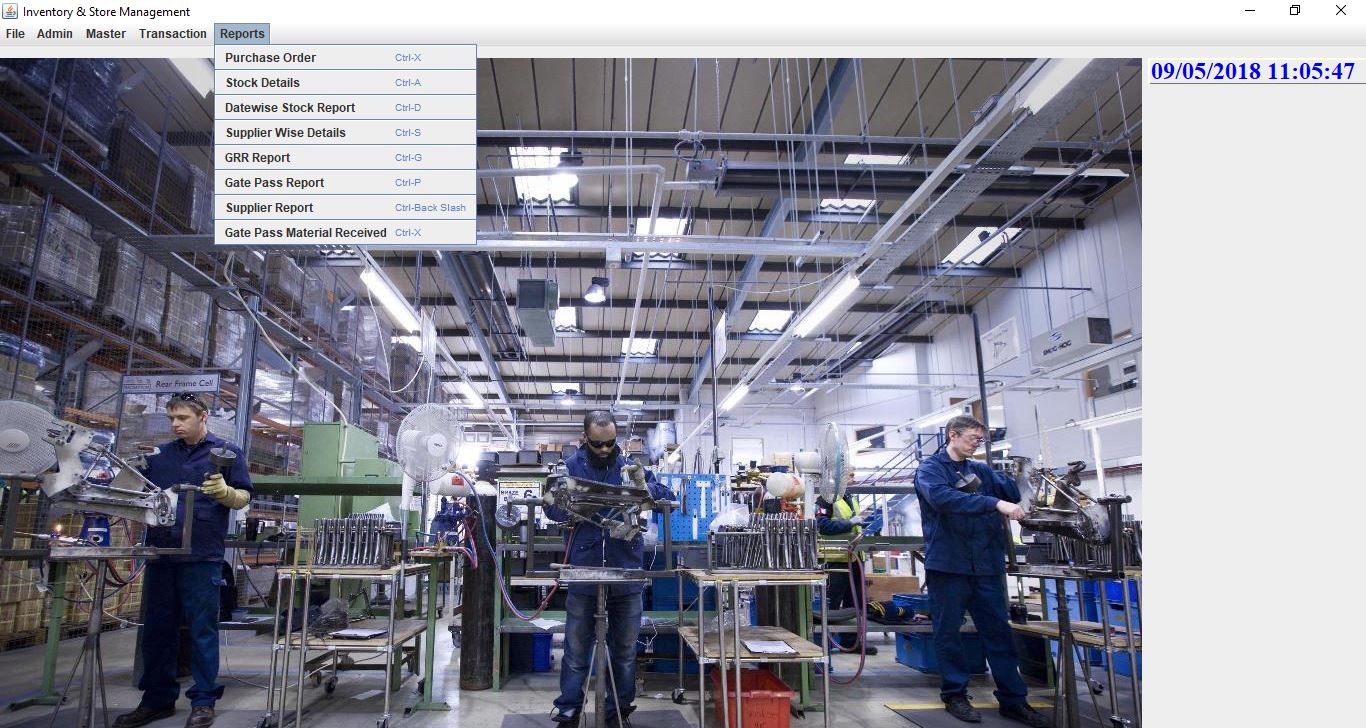
1. The second menu option is Master which provides the admin to feed the various entries required into the database.



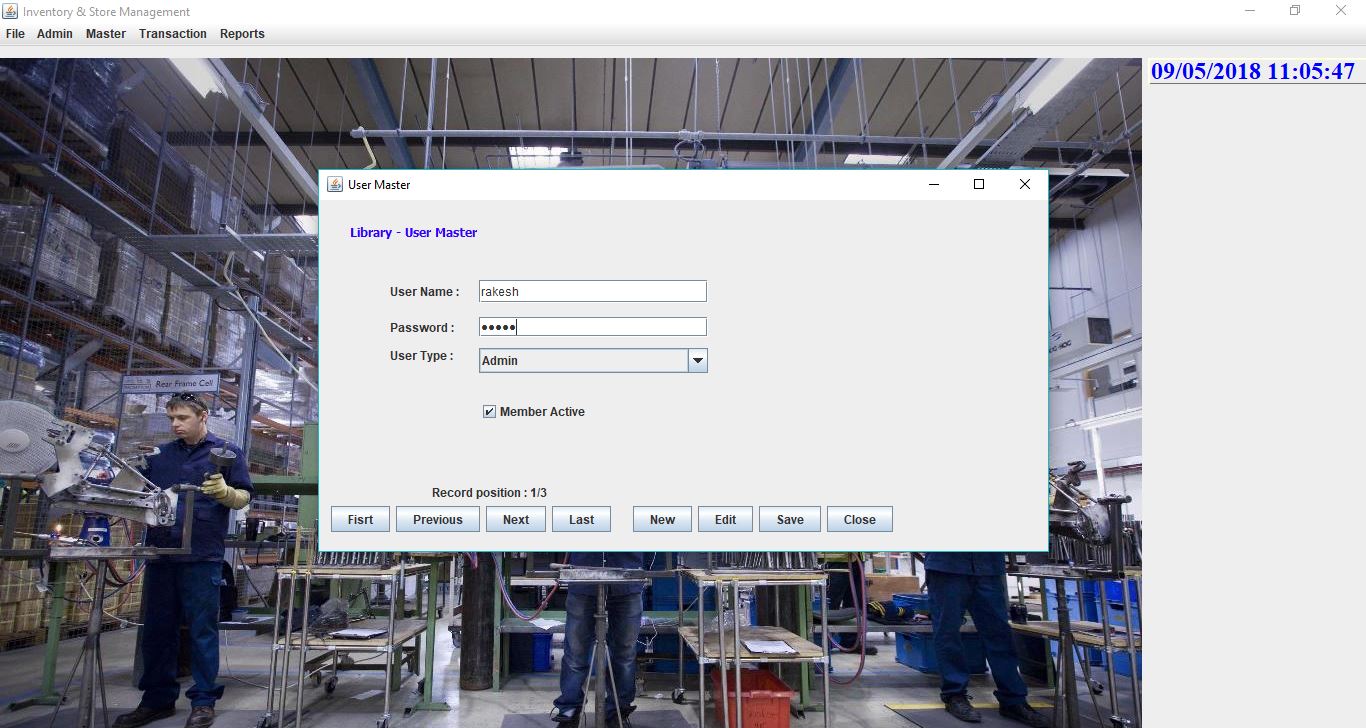
1. The third option allows the admin to carry out all the important activities related to the functioning of the application.



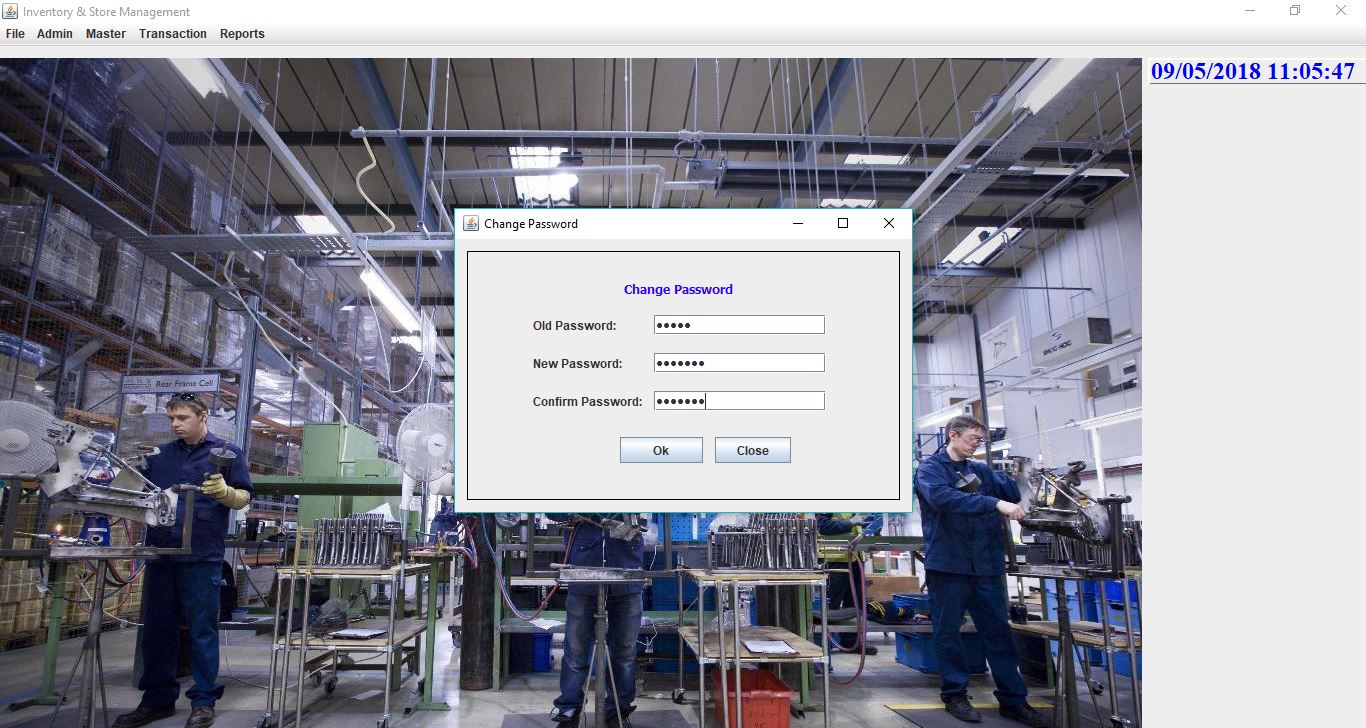
1. The final option helps the admin to directly access the existing generated reports and also print them.



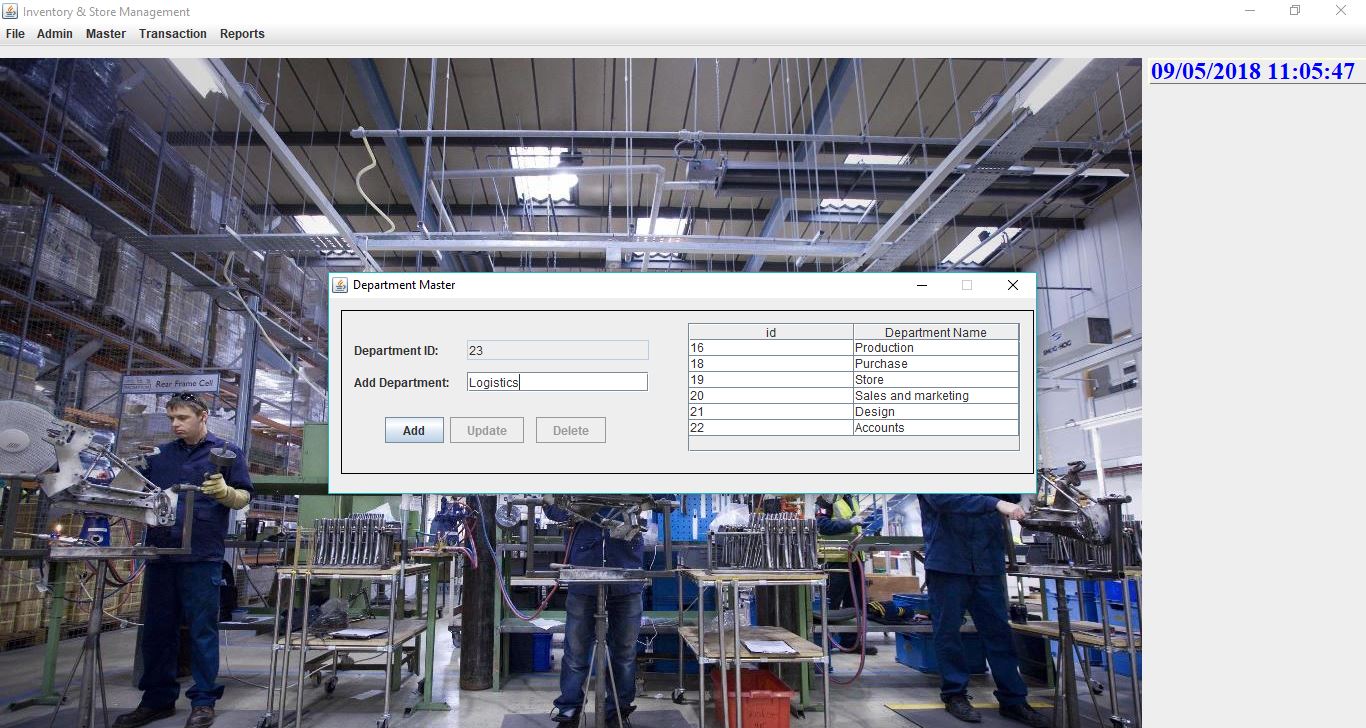
1. The New User option helps in creating a new user for the application by creating a new username and password.



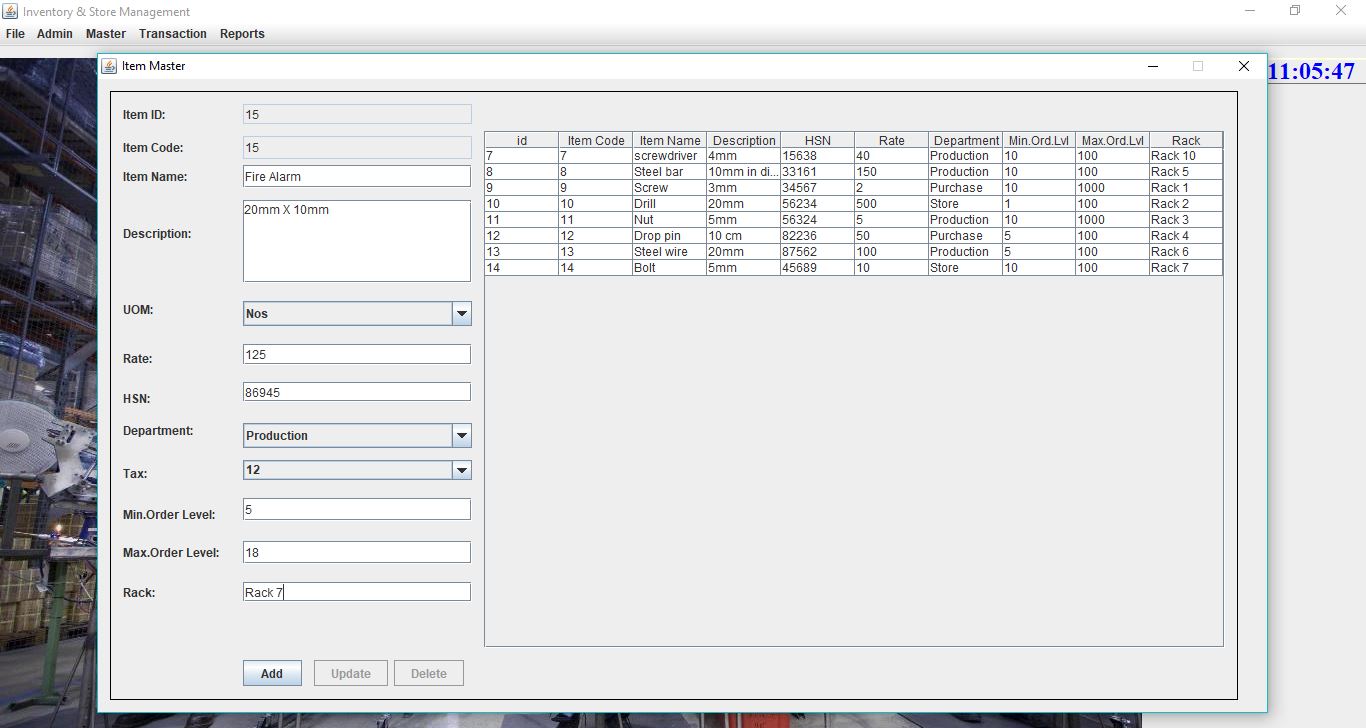
1. The Change Password option allows to change the password of the current logged in user.



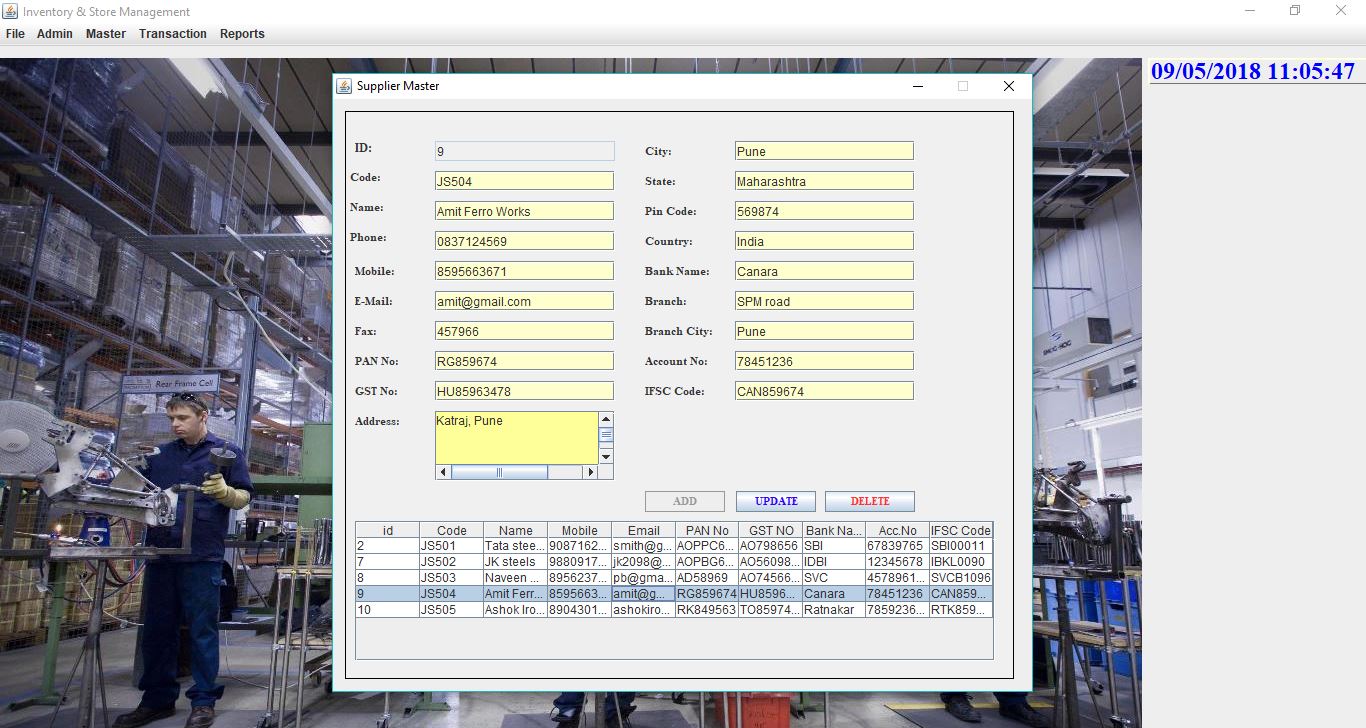
1. The Department option helps in storing the various departments of the industry.



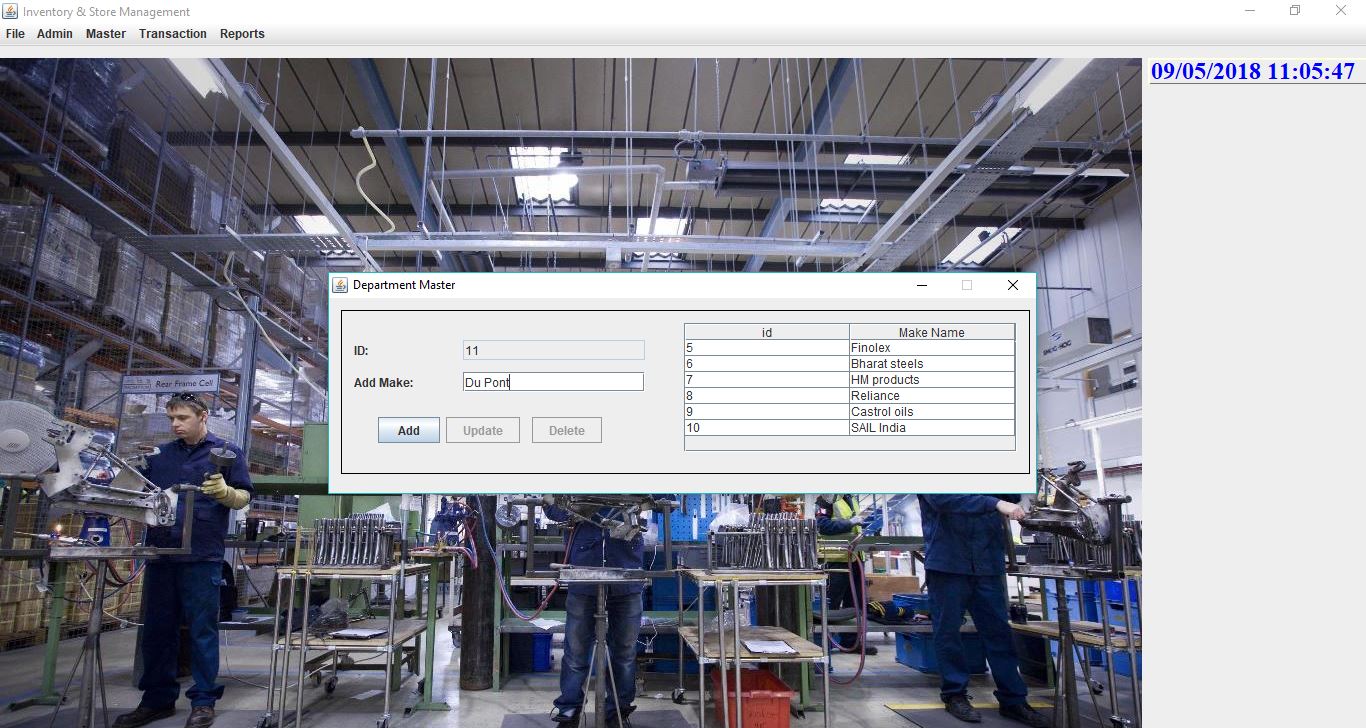
1. The Item option helps in storing the various items used by the industry.



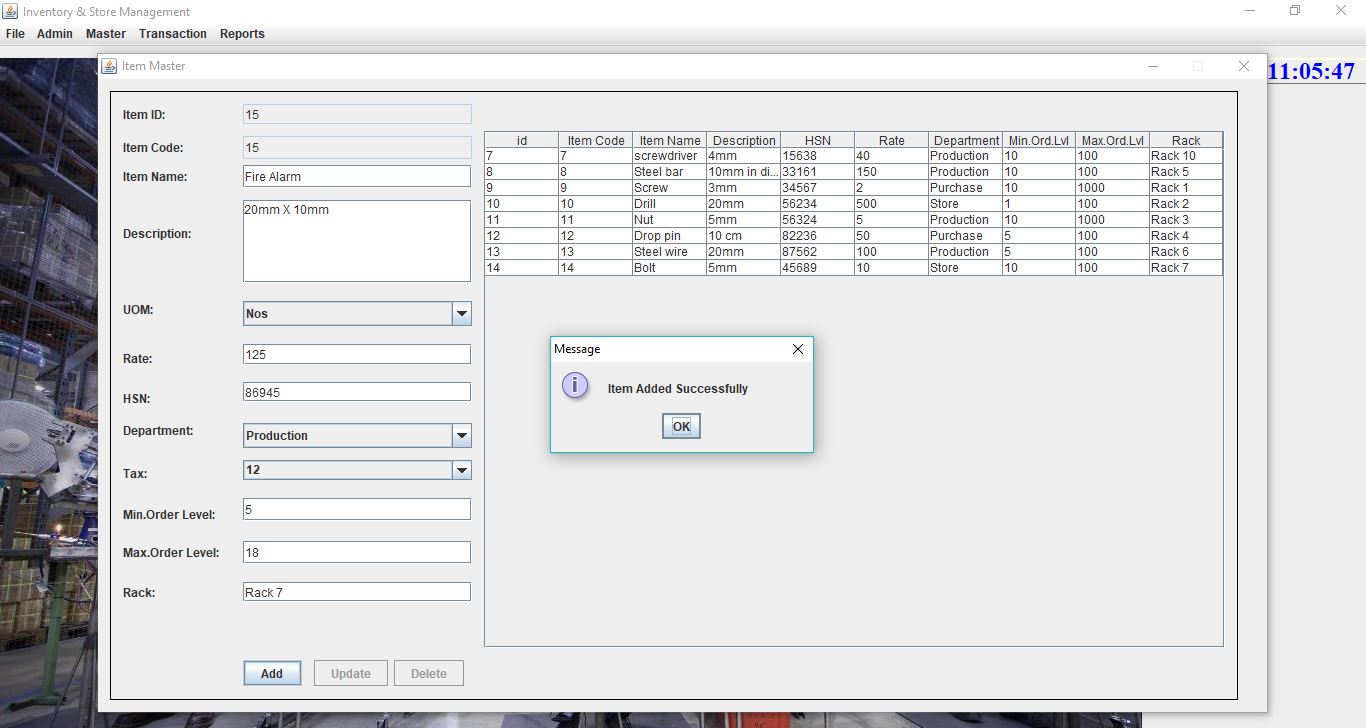
1. The Supplier option helps in storing the supplier details.



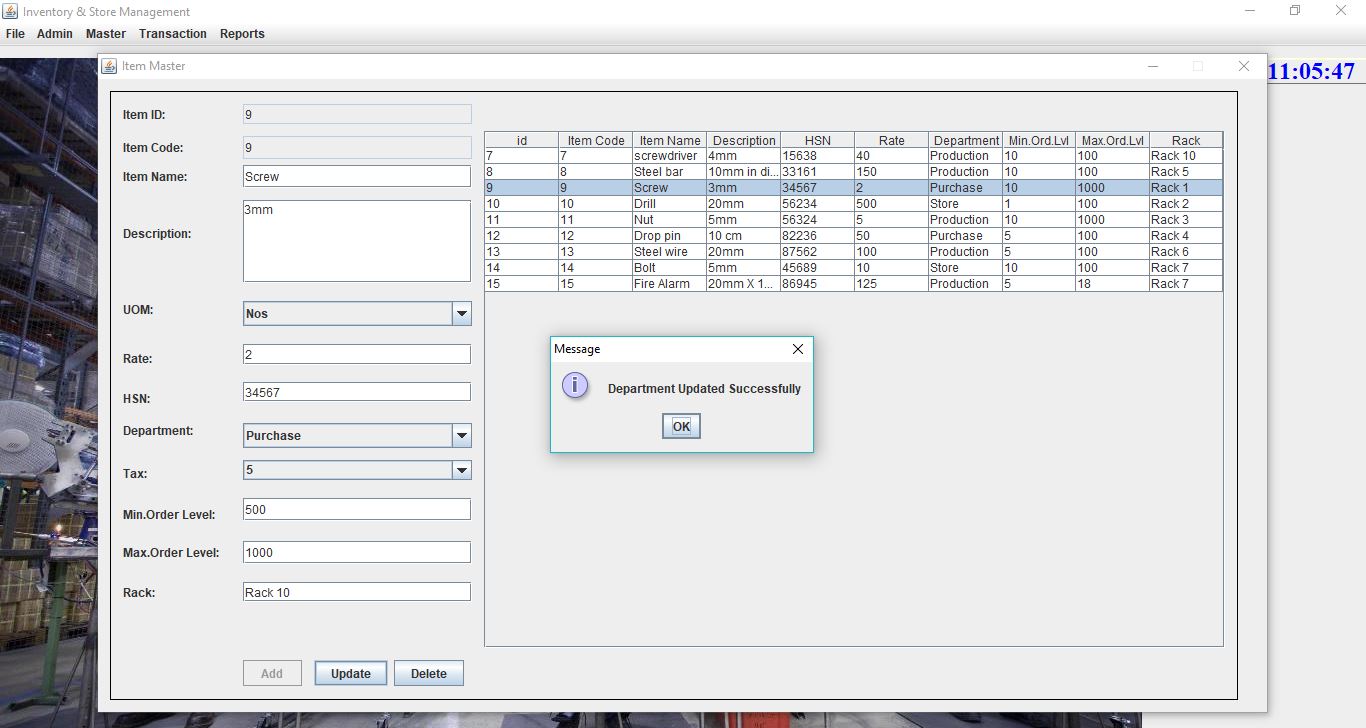
1. The Make option allows storing the make of the items used by the industry.



1. All the options in Master module show a message box after successfully adding the attributes in the database.



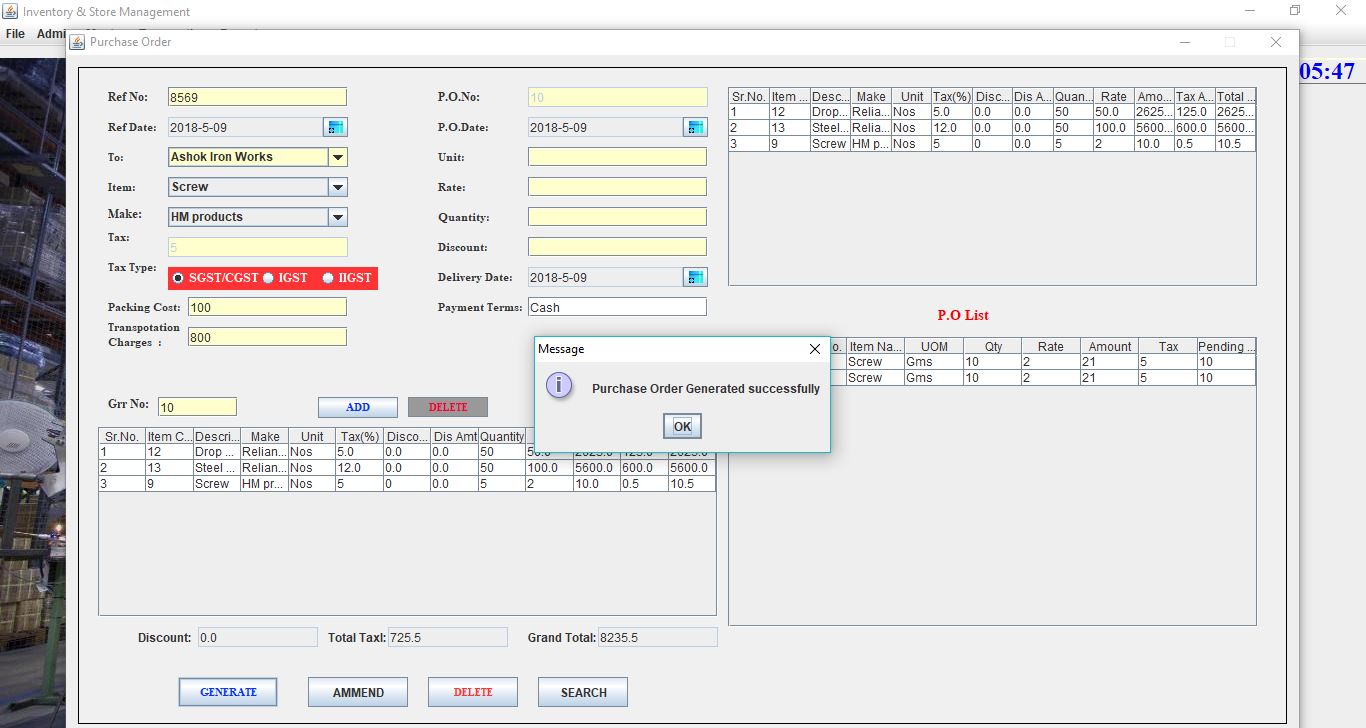
1. All the options in Master module allow editing the attributes in the database.



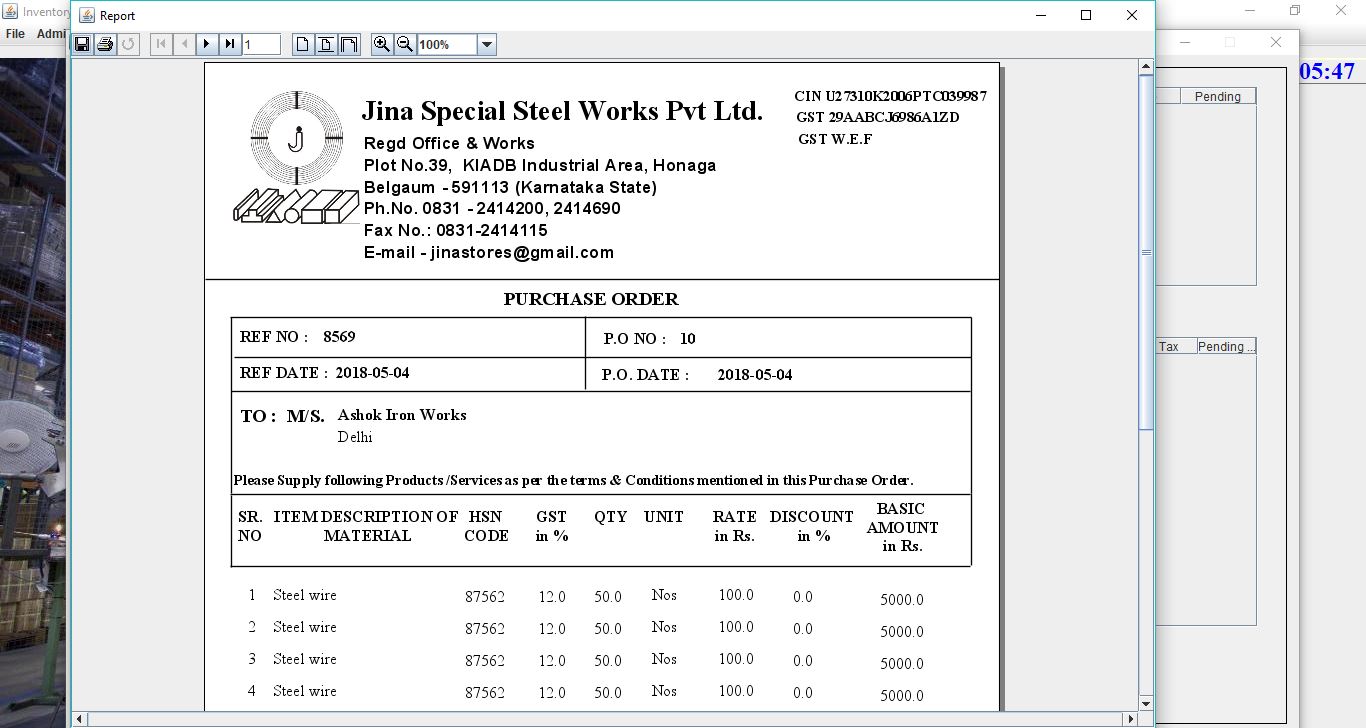
1. All the options in Master module allow deleting the attributes in the database.



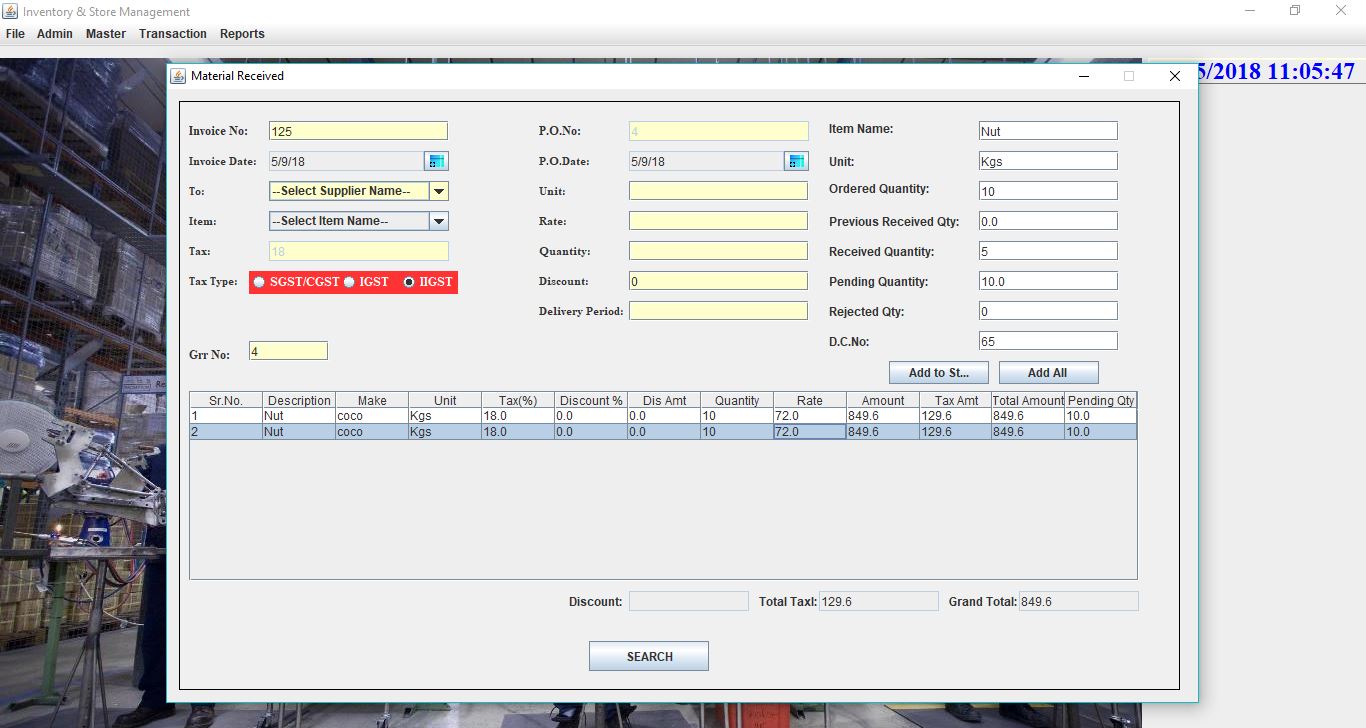
1. The purchase order option helps in generating purchase orders to suppliers.



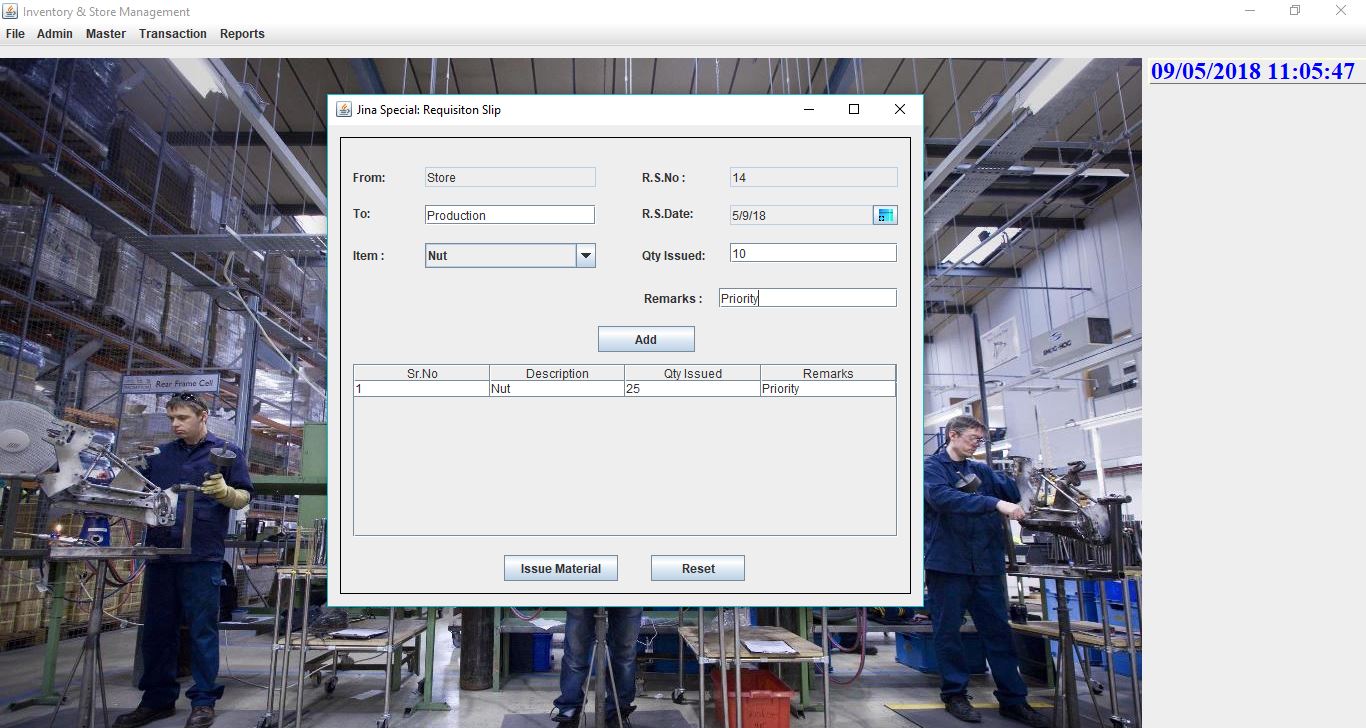
1. A sample report of the generated purchase order.



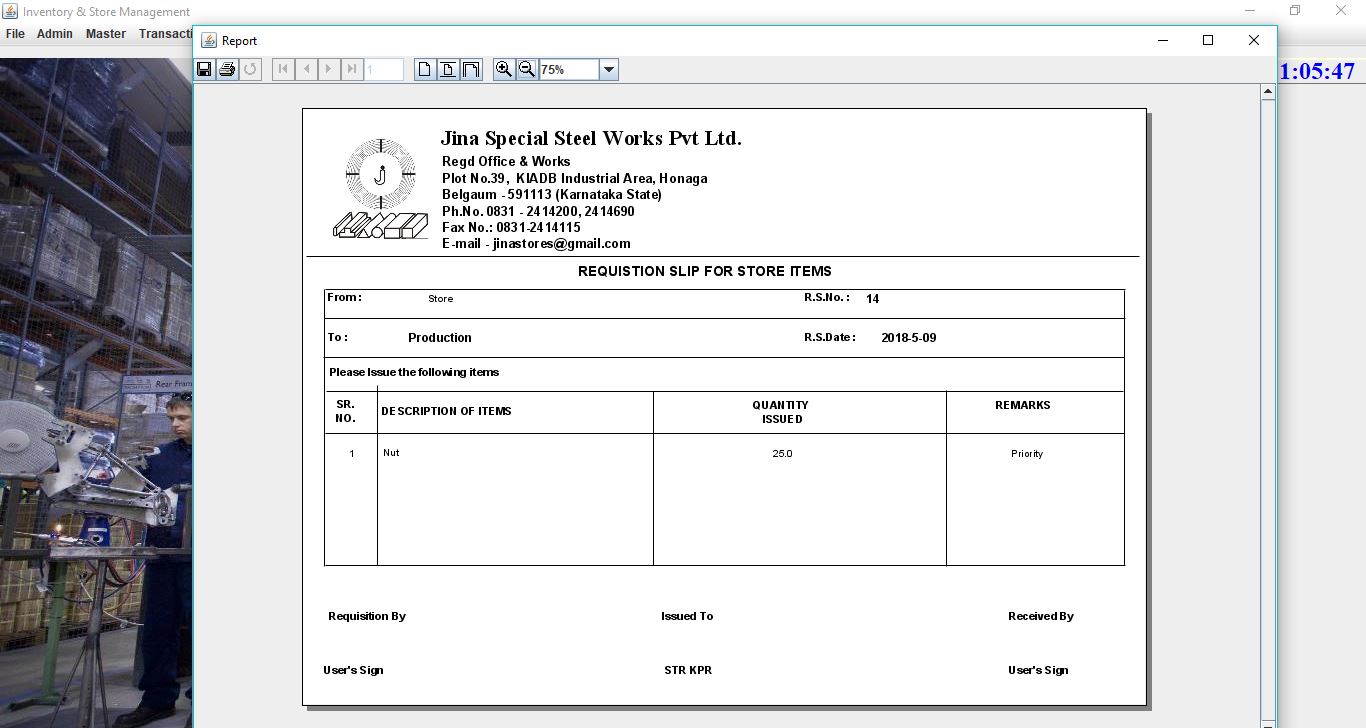
1. All the received material is added to stock using the Material Received option.



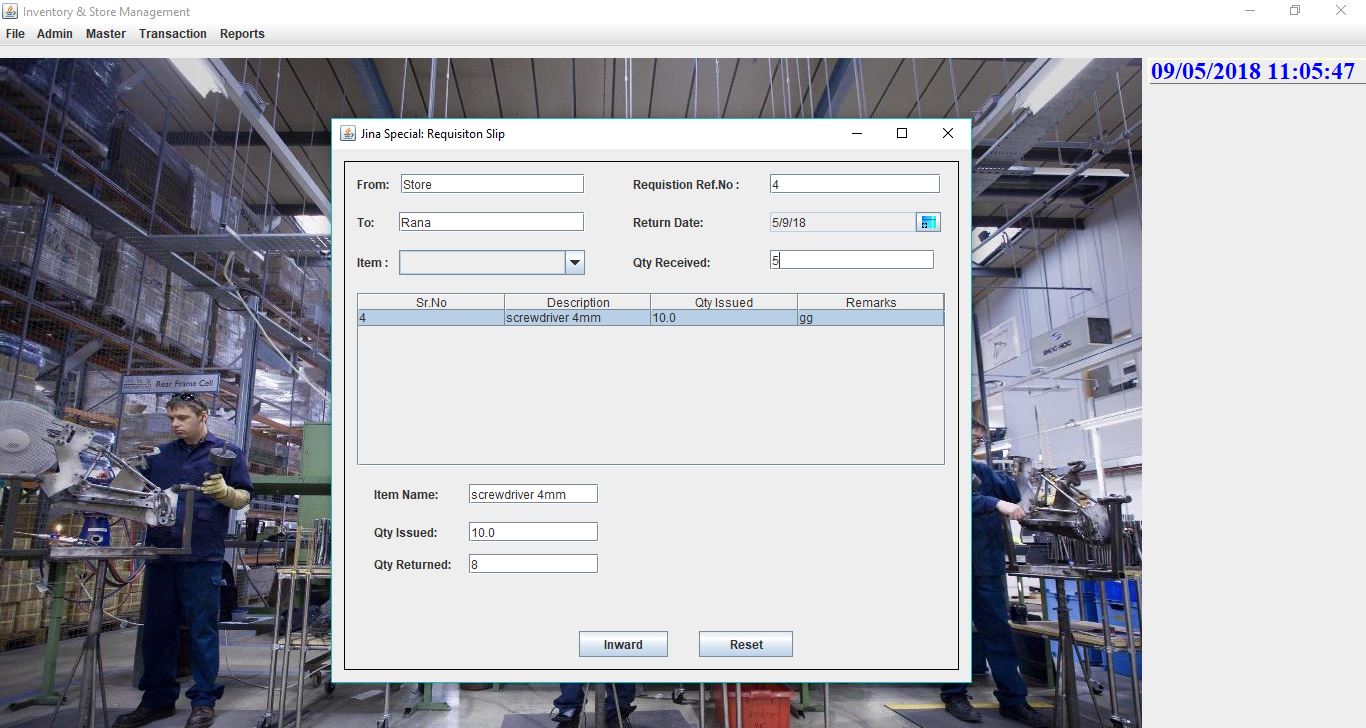
1. The Issue Material option is used to generate requisition slips.



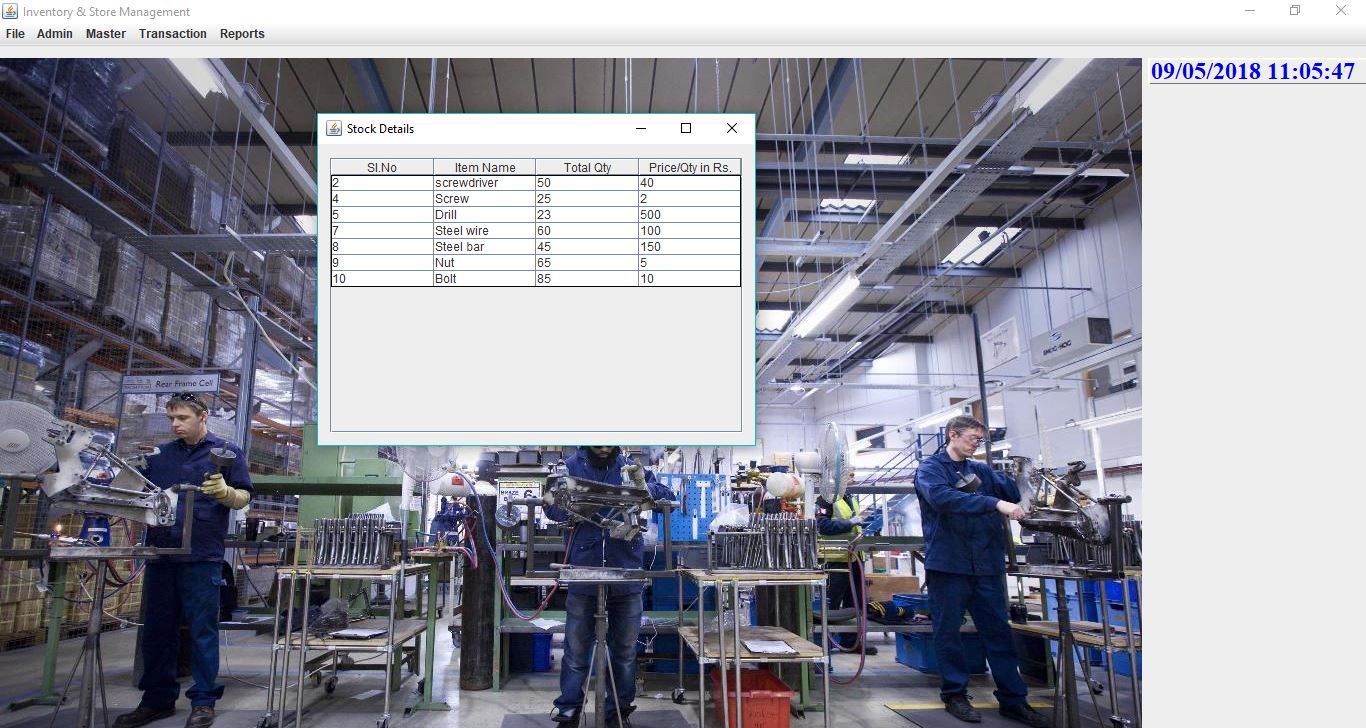
1. A sample requisition slip generated for the store items.



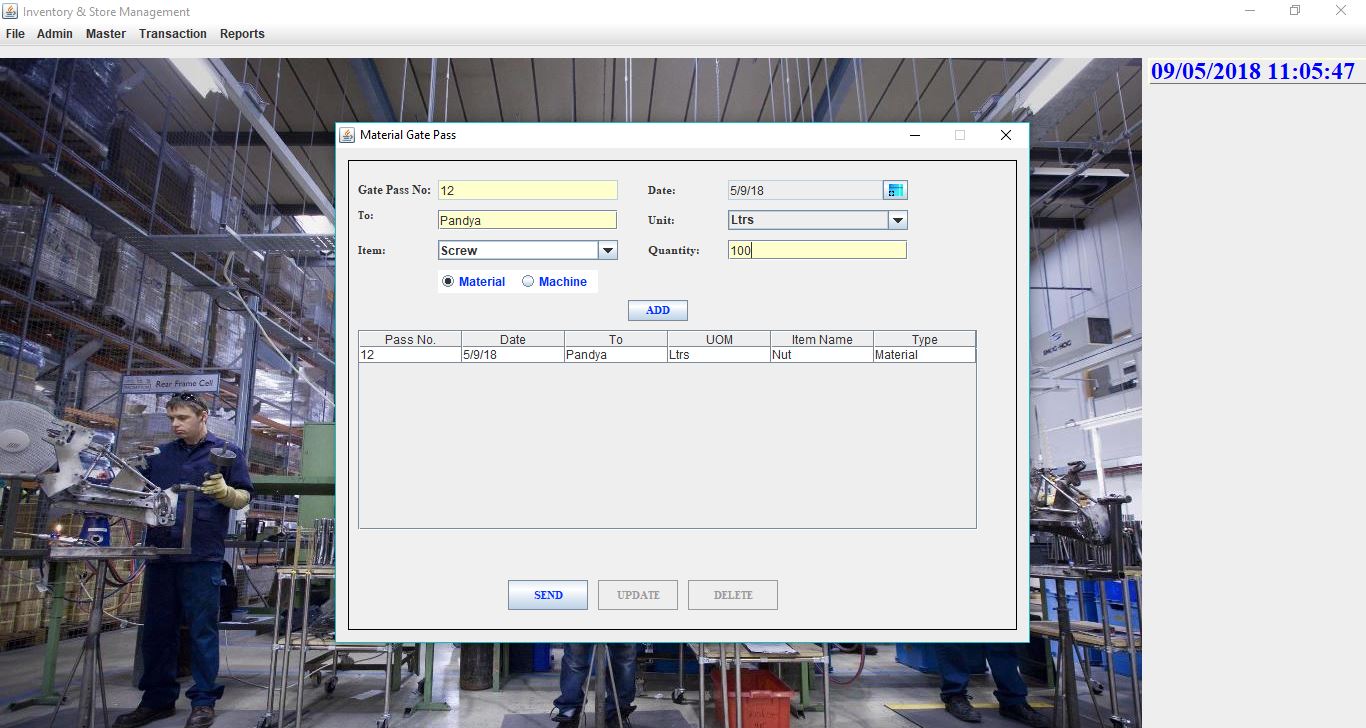
1. The Returned Material option is used to return the materials which were unused by the respective department the material was issued to. The returned materials are added back to the stock.



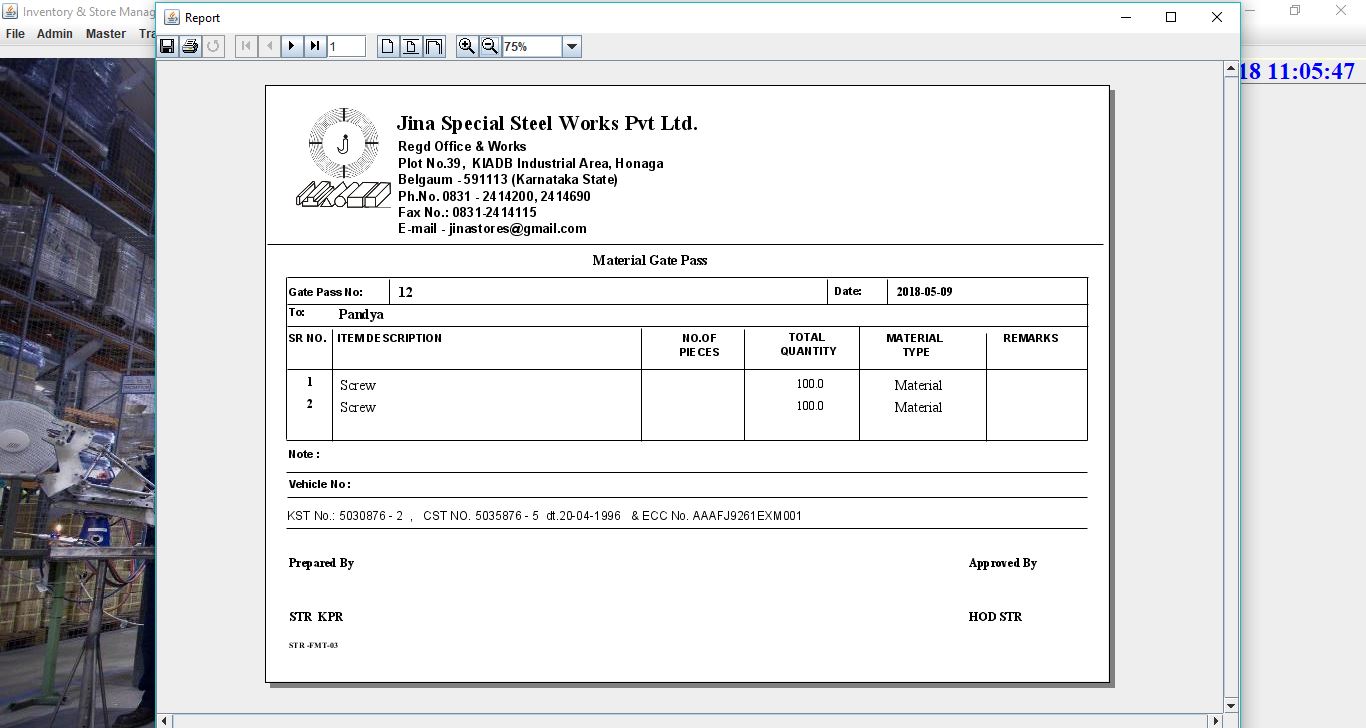
1. The Stock Details option shows the stock details of all the items received.



1. The Gate Pass is generated for raw materials which are to be sent out for various operations that cannot be performed in the industry.



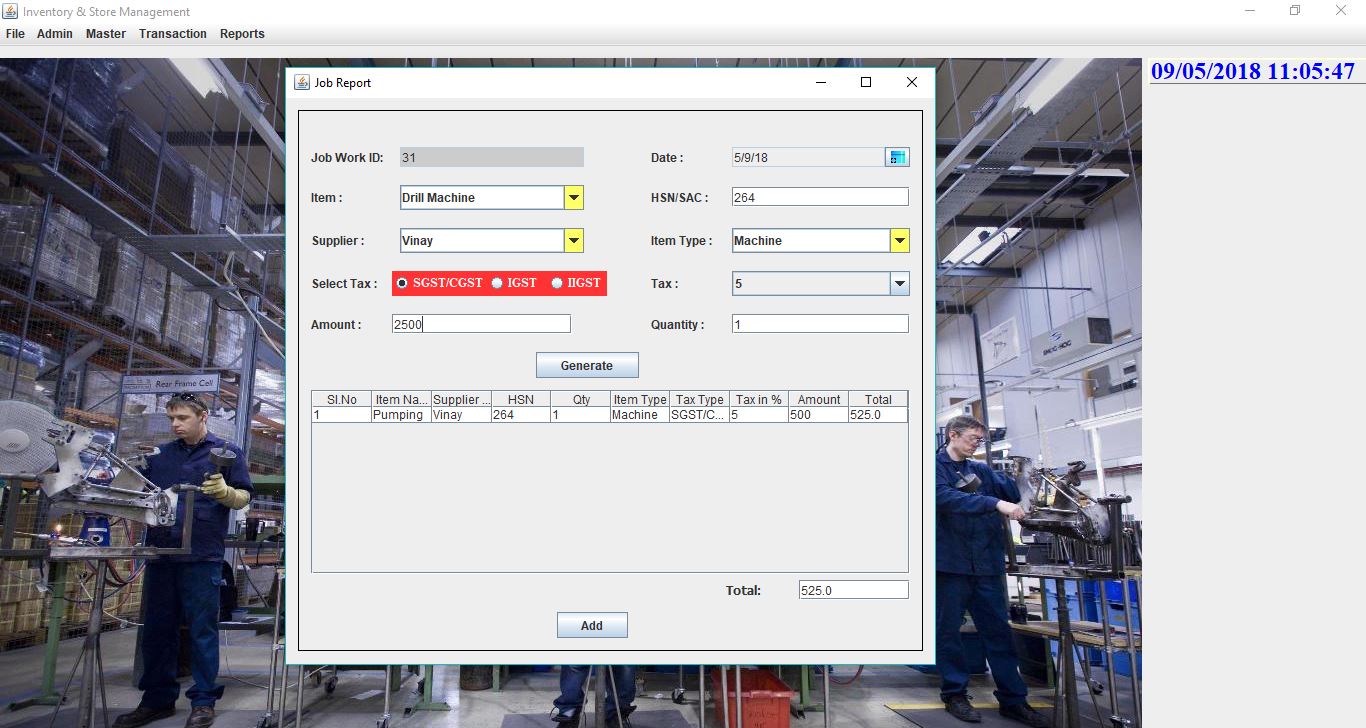
1. A sample gate pass generated is shown below.



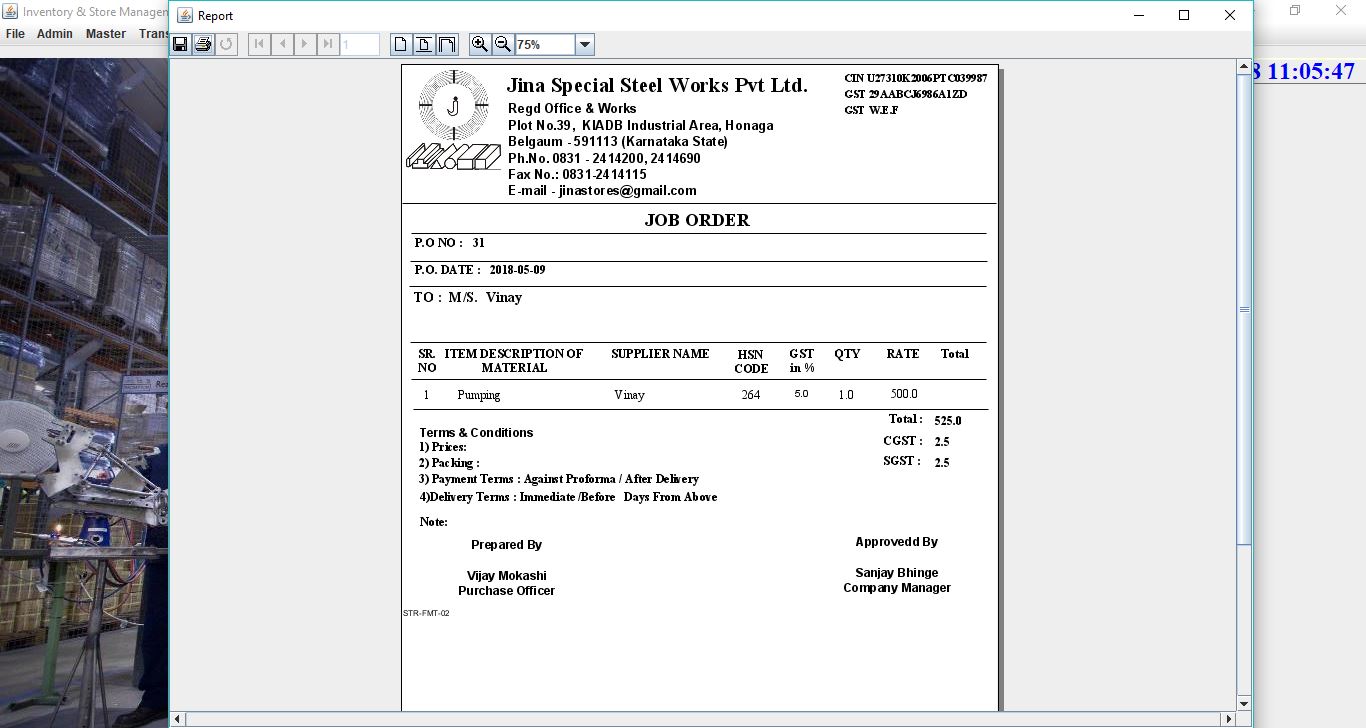
1. The Pending P.O. options helps in tracking the pending purchase orders as per item names.



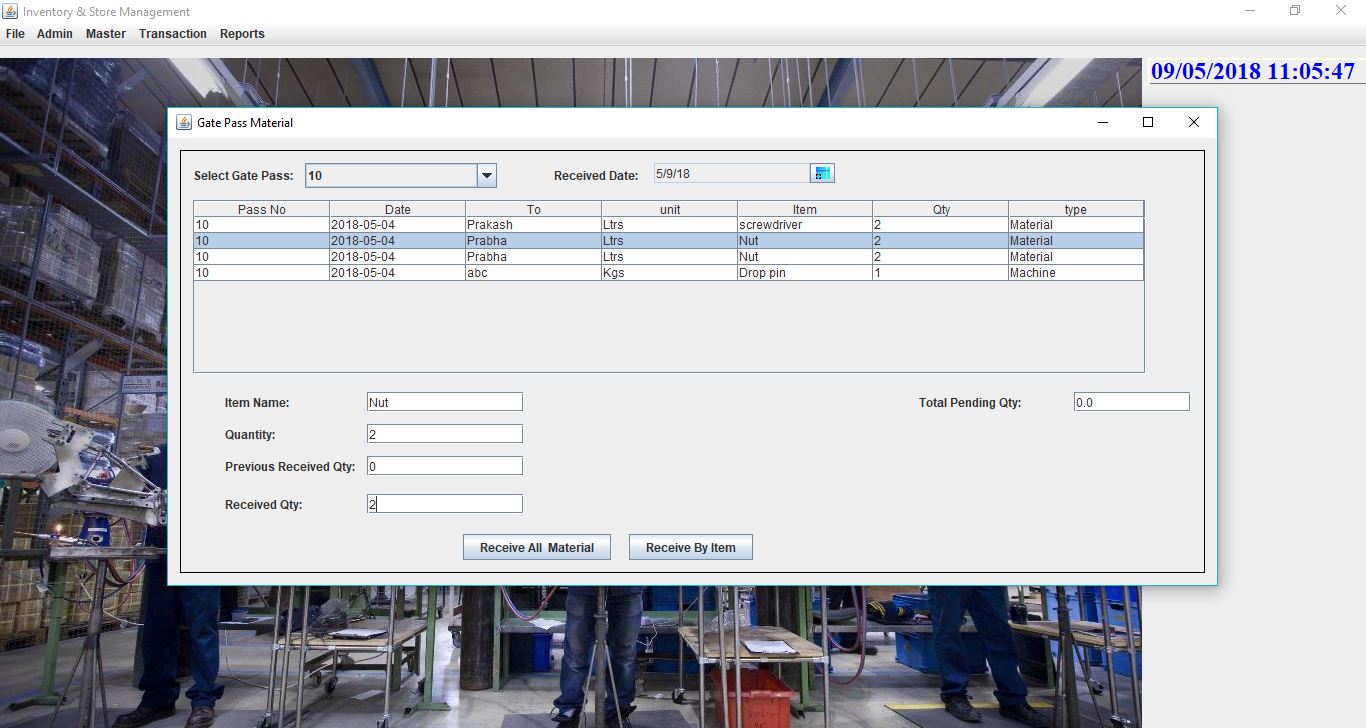
1. The Job Report option generates Job Orders to carried out by the workers.



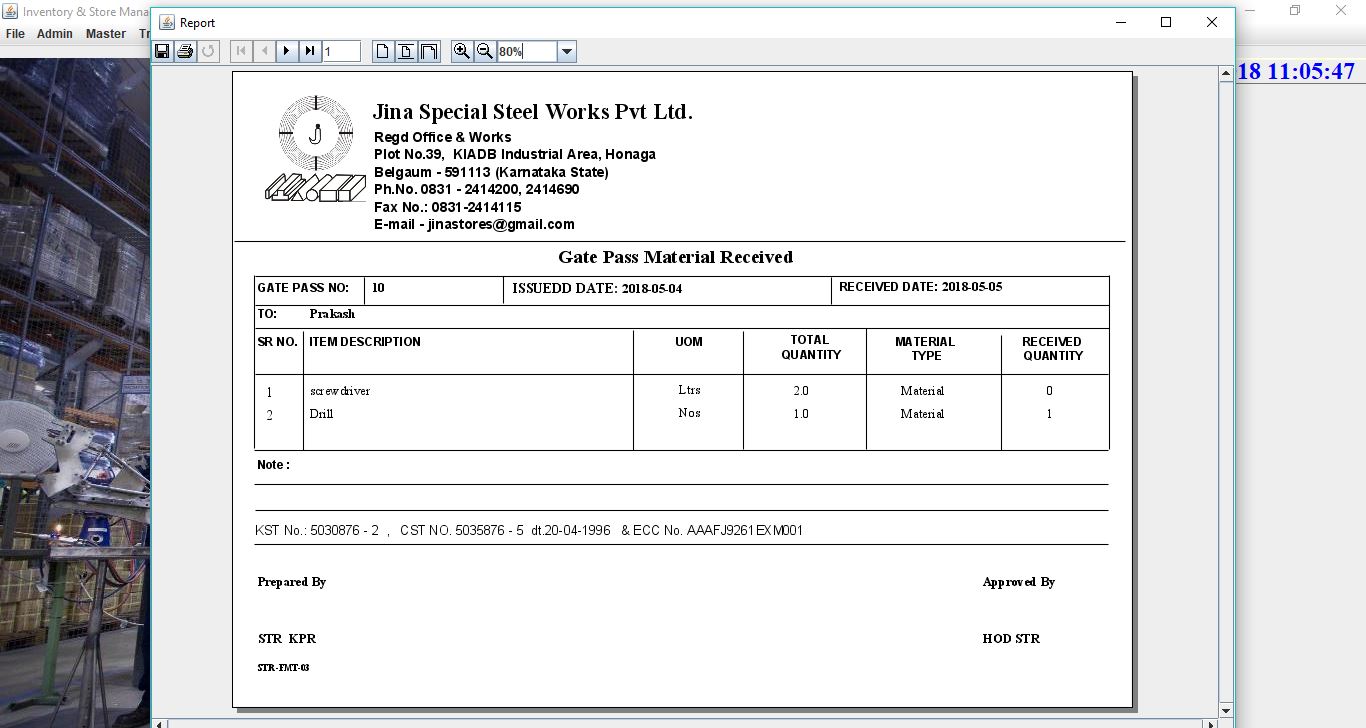
1. A sample Job Order report is shown below.



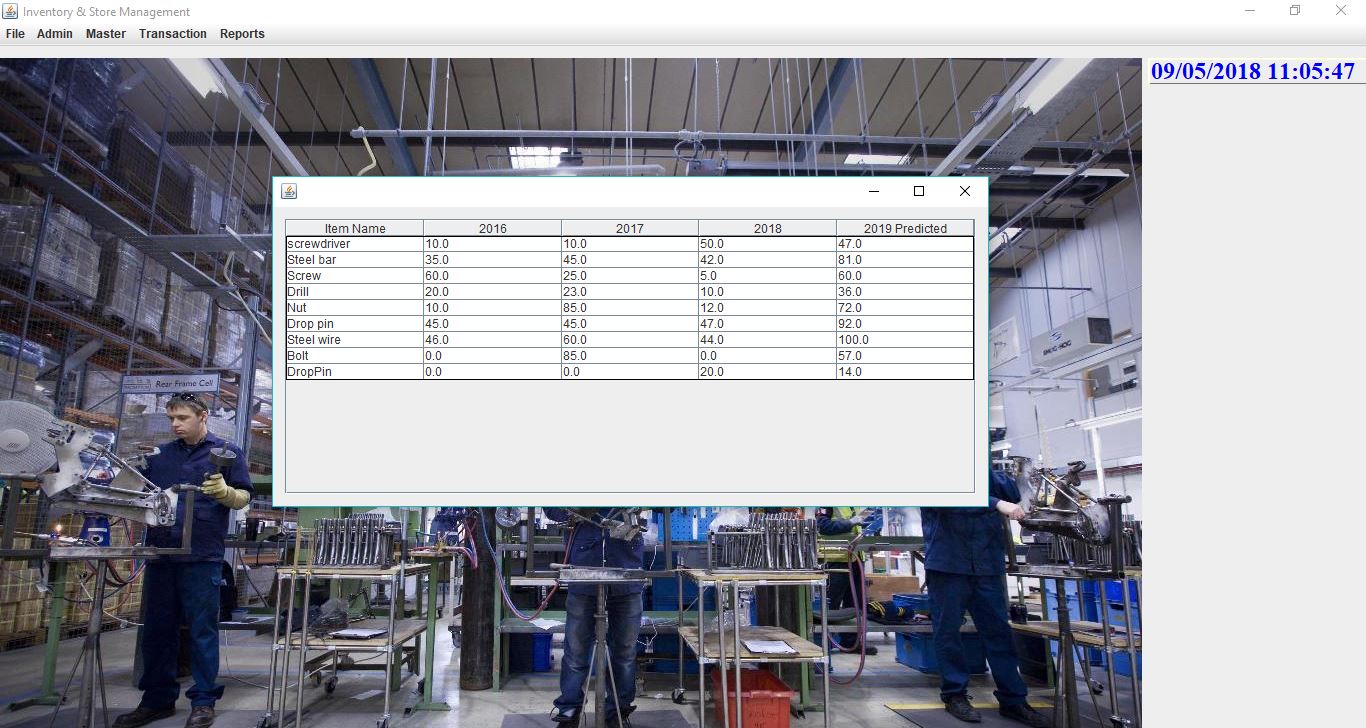
1. The Gate Pass Material Received option allows generating acknowledges for the received gate passes.



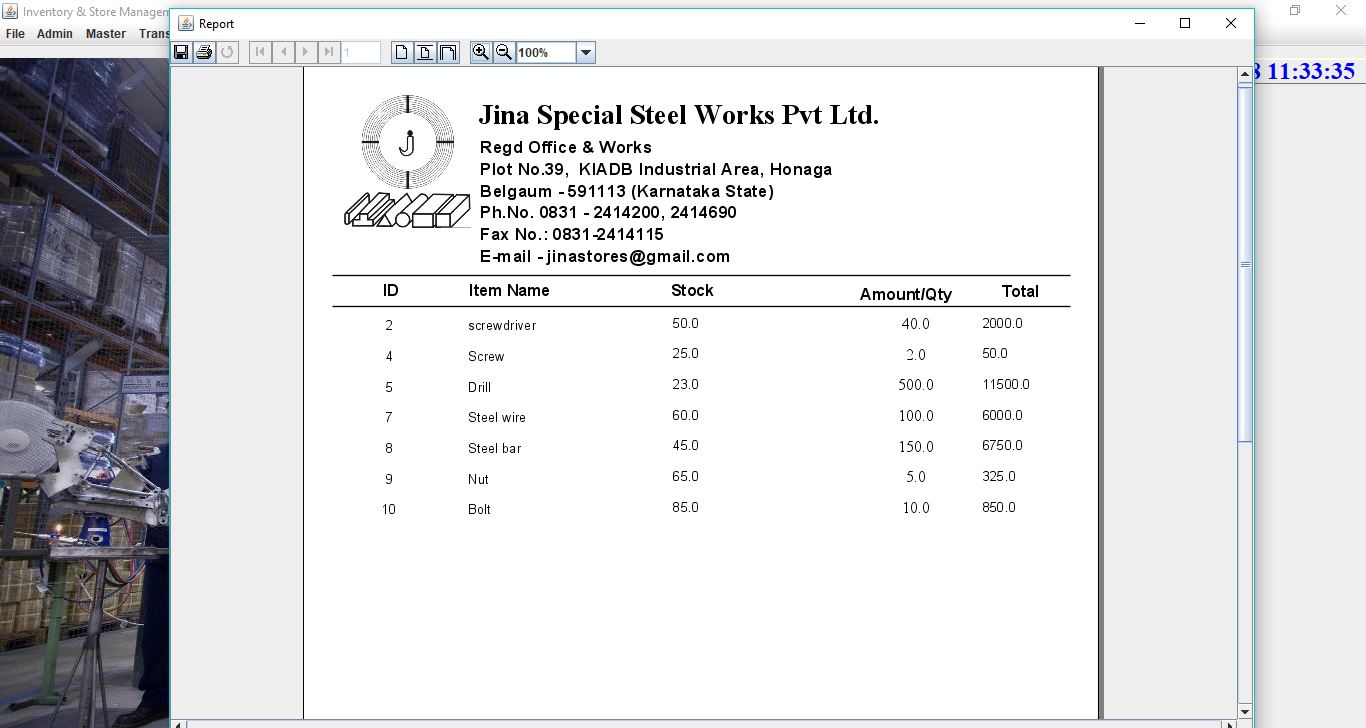
1. A sample of the Gate Pass Received generated report.



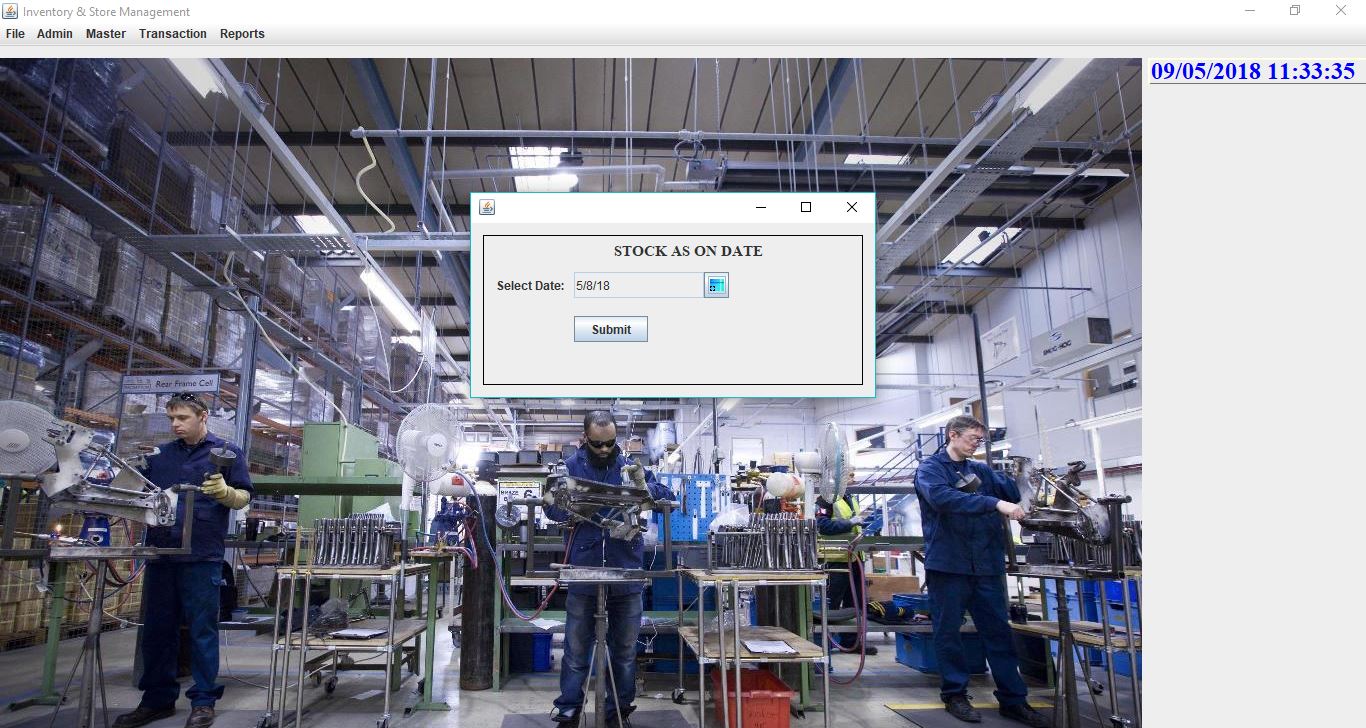
1. The Predictive Average option shows the required quantity of items predicted for the next year depending on the previous year stock details.



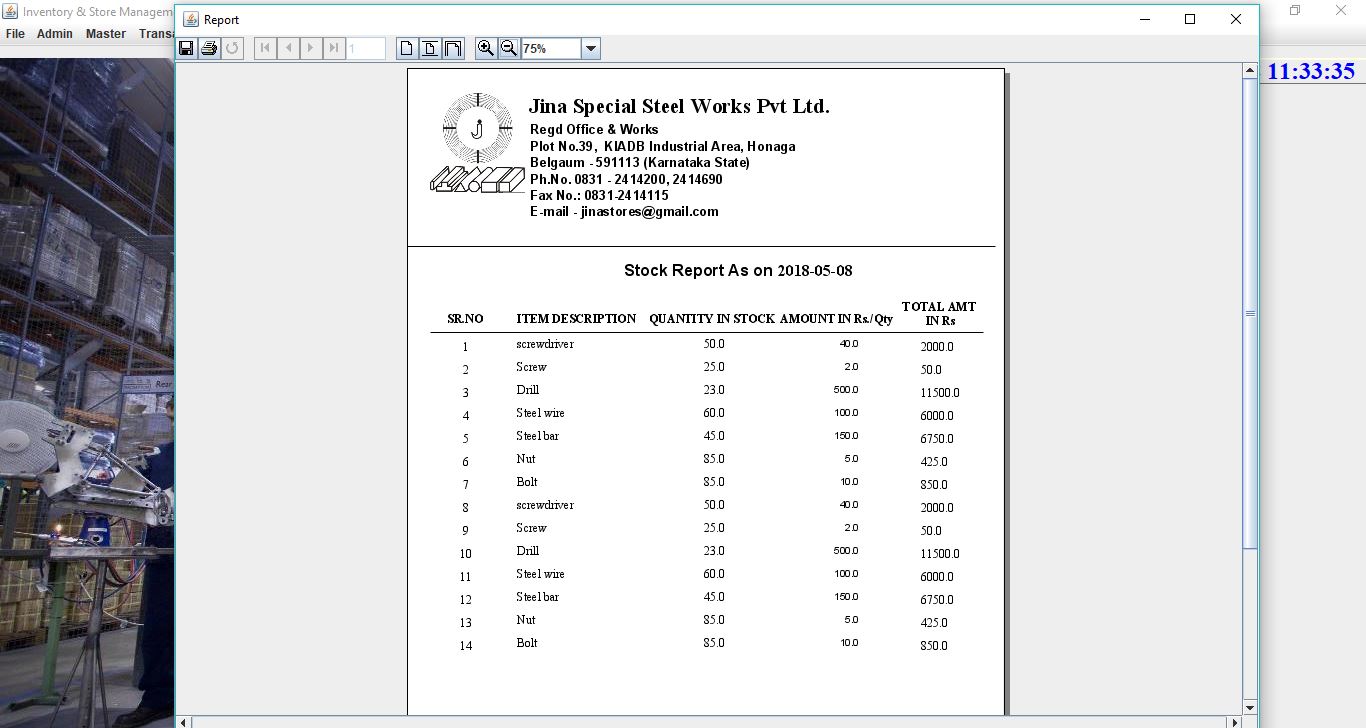
1. The Stock Details option generates a report to display the stock details.



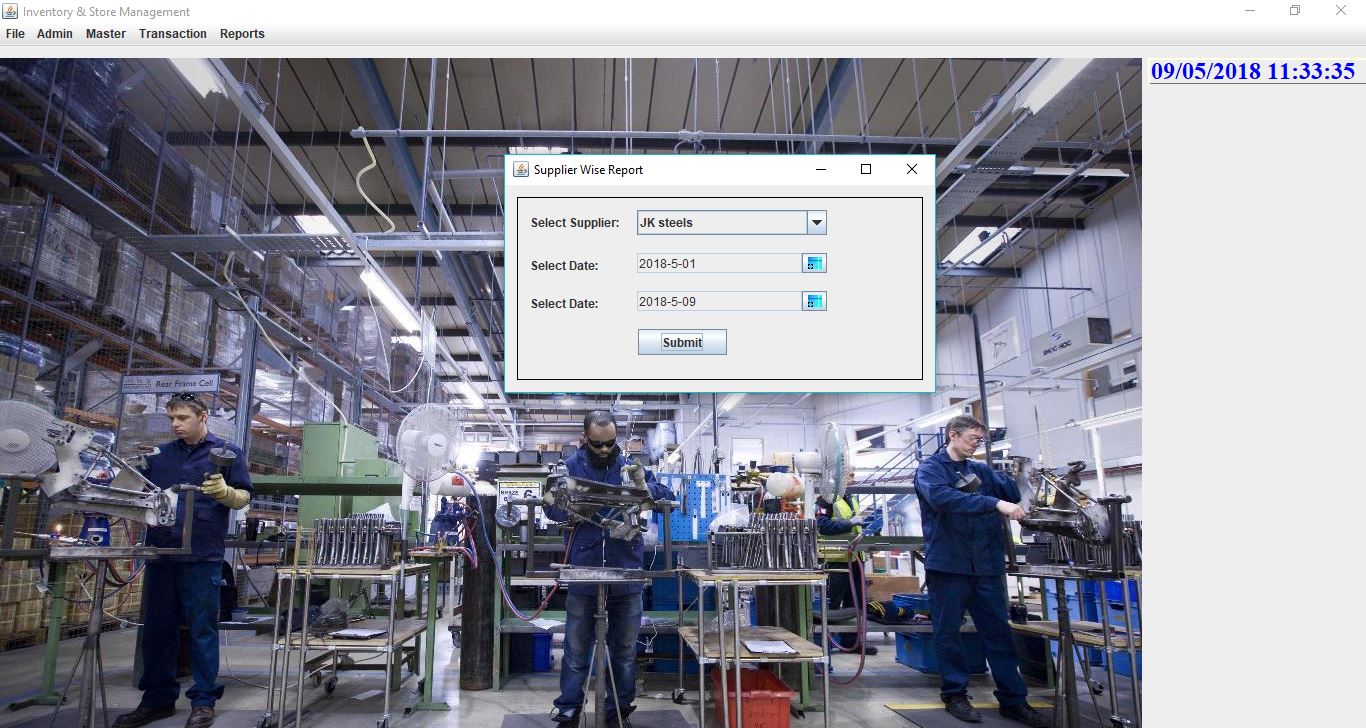
1. The Date Wise Details show the stock as on the date selected by the user.



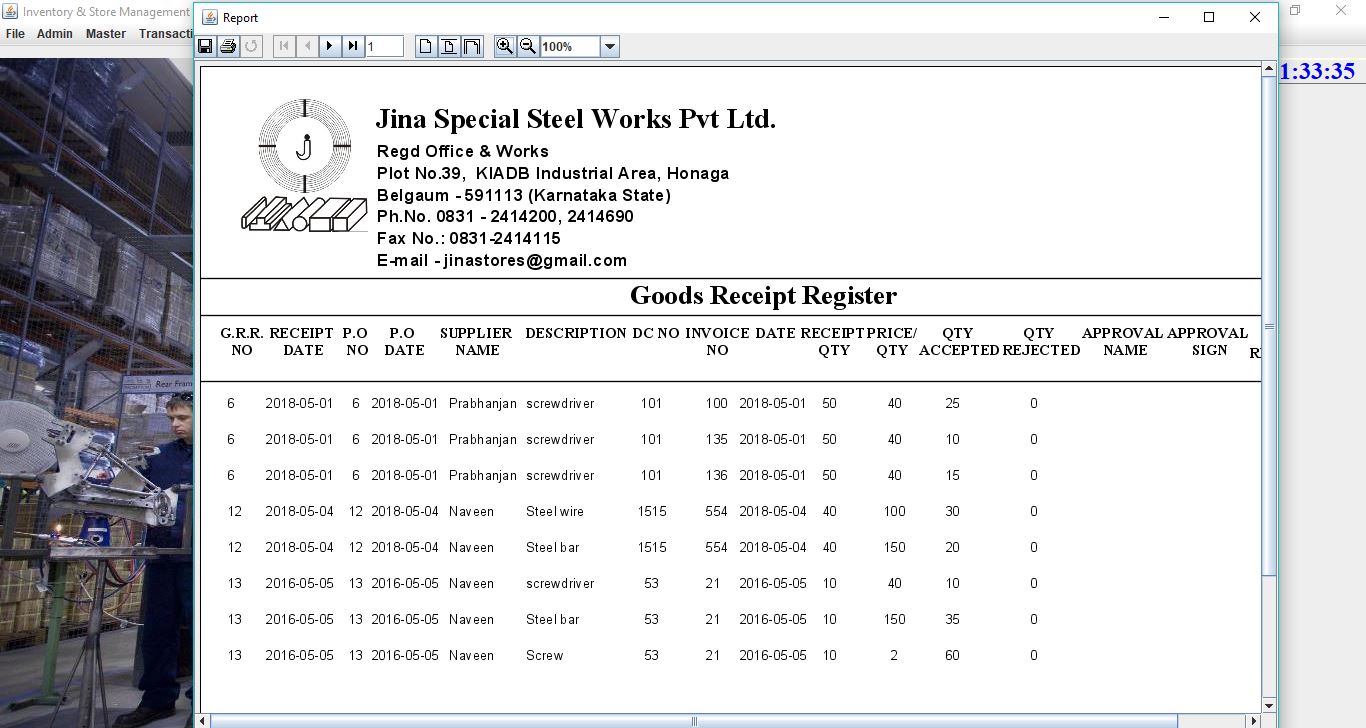
1. A sample report of stock as on date.



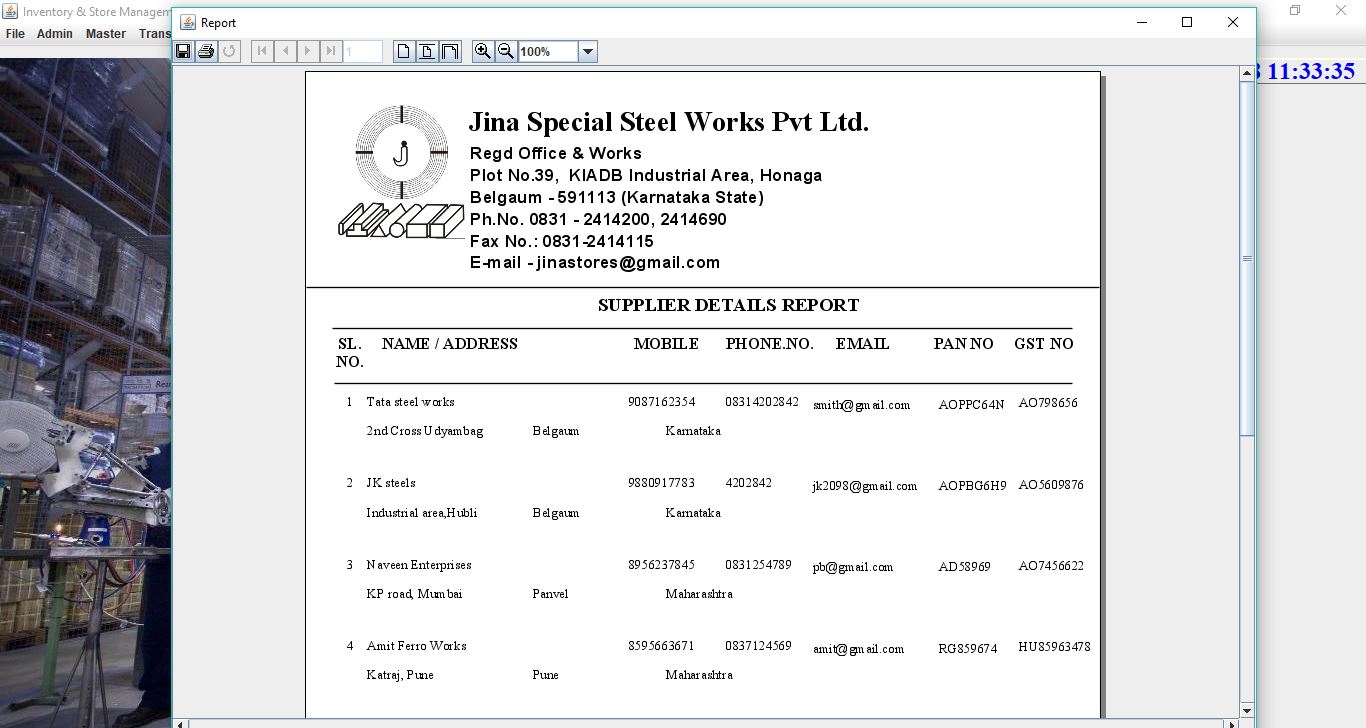
1. The Supplier Wise Details shows the stock details corresponding to the selected suppliers and the range of date specified.



1. A sample report of the Good Receipt Register (GRR).



1. A sample generated report which shows the supplier details.



**CONCLUSION**

An Enterprise Resource Management System (ERMS) has been designed with modules for purchase order, supplier, finance, stock and production. This system reflects all the requirements provided by the client and is developed according to company’s administration. The model is developed considering the future customization needed to be implemented according to the company’s necessary. The new model is a revolution over the existing model which is operated manually. The digitalised model therefore enhances all the management work to provide an optimum solution over the manual system. The proposed ERMS model can be used by any enterprise for developing their own ERMS that works efficiently on all platforms and devices. ERMS plays a key role in every enterprise and can be used to manage its entities and modules easily.

**FUTURE ENHANCEMENTS**

1. ‘Password Recovery’ option can be added to make the application more user-friendly.
2. The application can be made more graphical by displaying the stock levels and other details in a graphical manner.
3. A database of all the employees of the organization and their details can be made.
4. Employee attendance tracking can be made to integrate with the application.
5. Bar code scanning or RFID tracking can be used for better management of raw materials.
6. The existing predictive analysis can be improved to predict various other things relevant to the organization.

**REFERENCES**

[1] F. Mulazzani, B. Russo and G. Succi, "ERP Systems Development: Enhancing Organization's Strategic Control through Monitoring Agents," Computer and Information Science, 2009. ICIS 2009. Eighth IEEE/ACIS International Conference on, Shanghai, 2009, pp. 535-542.

[2] T. Yang, J. Choi, Z. Xi and Y. Sun, C. Ouyang and Y. Huang, "Research of Enterprise Resource Planning in a Specific Enterprise," 2006 IEEE International Conference on Systems, Man and Cybernetics, Taipei, 2006, pp. 418-422.

[3] Y. Xu, N. Rahmati and V. C. S. Lee, "A review of literature on Enterprise Resource Planning systems," 2008 International Conference on Service Systems and Service Management, Melbourne, VIC, 2008, pp. 1-6.

[4] N. Gibson, C. P. Holland and B. Light, "Enterprise resource planning: a business approach to systems development," Systems Sciences 1999. HICSS-32. Proceedings of the 32nd Annual Hawaii International Conference on, Maui, HI, USA, 1999, pp. 9.

[5] C. Han, J. Yi and Y. Li, "Research of ERP System Application in Project Management," Computing, Measurement, Control and Sensor Network (CMCSN), 2012 International Conference on, Taiyuan, 2012, pp. 368-370.

[6] S. Nagpal, S. K. Khatri and A. Kumar, "Comparative study of ERP implementation strategies," Systems, Applications and Technology Conference (LISAT), 2015 IEEE Long Island, Farmingdale, NY, 2015, pp. 1-9.

[7] Y. Y. Shih, "A Study of ERP Systems Selection via Fuzzy AHP Method," Information Engineering and Electronic Commerce (IEEC), 2010 2nd International Symposium on, Ternopil, 2010, pp. 1-4.

[8] M. Parhizkar and M. Comuzzi, "A framework for impact analysis of post-implementation enterprise resource planning modifications," 2016 SAI Computing Conference (SAI), London, 2016, pp. 706-714.

[9] K. Y. Chen, H. Y. Lee and S. Tu, "Study on User's Satisfaction of Enterprise Resource Planning System - An Example of Manufacturing," 2016 International Symposium on Computer, Consumer and Control (IS3C), Xi'an, 2016, pp. 1010-1013.

[10] B. R. Bhardwaj, "Sustainable supply chain management through enterprise resource planning (ERP): A model of sustainable computing," Computing for Sustainable Global Development (INDIACom), 2014 International Conference on, New Delhi, 2014, pp. 166-171.

[11] W. David, Y. Ruldeviyani and P. Sandhyaduhita, "Analysis and design of Enterprise Resource Planning (ERP) system for Small and Medium Enterprises (SMEs) in the Sales Business Function Area," Advanced Computer Science and Information Systems (ICACSIS), 2013 International Conference on, Bali, 2013, pp. 255-260.

[12] W. Shaojun, W. Gang, l. Min and G. Guoan, "Enterprise resource planning implementation decision & optimization models," in Journal of Systems Engineering and Electronics, vol. 19, no. 3, pp. 513-521, June 2008.

[13] W. H. Tsai, S. J. Lin, K. C. Lee, W. R. Lin, J. Y. Liu and J. L. Hsu, "Examining the implementation risks affecting different aspects of Enterprise Resource Planning project success," Computers and Industrial Engineering (CIE), 2010 40th International Conference on, Awaji, 2010, pp. 1-6.

[14] M. Al-Ghamdi, A. P. Chester and S. A. Jarvis, "Predictive and Dynamic Resource Allocation for Enterprise Applications," Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on, Bradford, 2010, pp. 2776-2783.