The Relational database Design/Modeling/Implementation in MySQL & Oracle Project

Raw Materials Inventory Management System (RMITS)

Christian McArthur
Fall 2011
CS 5332

Professor: Dr. C.J. Hwang
Date: December 2, 2011

Texas State University

Table of Contents

Chapter 1 Introduction....4
1.1 Motivation....4
1.2 Project Description and Scope....4
1.3 Software Used to Prepare This Document....5
Chapter 2 Study of System Functional Requirements....5
2.1 Introduction....5
2.2 Summary of Interview and Study....5
2.2.1 Business Owner....5
2.2.2 Accountant....6
2.2.3 Machinist....6
2.3 Functional Requirements: Input Business Forms and Screens....6
2.3.1 Purchase Order Form....6
2.3.2 Purchase Receipt....6
2.3.3 Request for Quote....7
2.4 Functional Requirements: Output Business Forms and Screens....7
2.4.1 Inventory Report....7
2.4.2 Work Order....7
2.4.3 Project Report....8
Chapter 3 External Scheme (User's Views)....8
3.1 Diagram of System Model of the Project....8
3.2 Design of the User Interface for User's View....8
3.3 Design of User Reports and Screens of the User's Views....9
Chapter 4 Conceptual Schema and Logical Model of the System....10
4.1 Descriptions of All Relations and Attributes....10
4.1.1 Customer Relation....10
4.1.2 Projects Relation....10
4.1.3 Materials Relation....11
4.1.4 Vendor Relation....11
4.1.5 PurchaseOrders Relation....12
4.1.6 MaterialsPurchased Relation....12
4.1.7 MaterialsUsed Relation....12
4.2 Functional Dependencies for Normalized Relations....13
4.3 Example Tuples for All Normalized Relations....13
4.3.1 Customer Table....13
4.3.2 Projects Table....13
4.3.3 MaterialsUsed Table....14
4.3.4 Materials Table....14
4.3.5 Vendor Table....14
4.3.6 PurchaseOrder Table....15
4.3.7 MaterialsPurchased Table....15
4.4 Descriptions of all Required Relationships....15
4.5 ER Diagram of the System....16
Chapter 5 Data Dictionary....16
5.1 Relations, Attributes, Domains, Comments....16

5.1.1 Customer 16
5.1.2 Projects....17
5.1.3 Material 17
5.1.4 MaterialsUsed....17
5.1.5 Vendor....18
5.1.6 PurchaseOrder....18
5.1.7 MaterialsPurchased....18
5.2 Attributes, Types, Descriptions, Restrictions....19
5.3 Cross References....20
Chapter 6 Queries and RDBMS Implementation I....20
6.1 The Software, RDBMS and Platform Used....20
6.2 Database Creation and Required Commands....20
6.3 User Interface Implementation 23
6.3.1 New Customer Form....23
6.3.2 New Material Vendor Form....23
6.3.3 New Material Form....24
6.3.4 Purchase Order Form....24
6.3.5 Purchase Order List....25
6.3.6 Purchase Receipt Form....25
6.3.7 Inventory Report....26
6.3.8 Request for Quote Form 26
6.3.9 Work Order Report....27
6.3.10 RFQ Bid Form 27
6.3.11 Work Order – Assign Machinist Form....28
6.3.12 Work Order Details....29
Chapter 7 Queries and RDBMS Implementation II....29
7.1 The Software, RDBMS and Platform Used....29
7.2 Database Creation and Required Commands....29
7.3 User Interface Implementation....32
7.3.1 New Customer Form....32
7.3.2 New Materials Vendor Form 32
7.3.3 New Raw Materials Form 33
7.3.4 Purchase Order Entry Form 33
7.3.5 Purchase Receipt Form 34
7.3.6 Inventory Report....34
7.3.7 Request For Quote Form 35
7.3.8 Work Order Details....35
Chapter 8 Conclusion....36

Chapter 1 Introduction

1.1 Motivation

There are a lot of enterprise management software that allows a business to track almost every imaginable aspect of the business. For example, in the manufacturing industry, there are software packages that can track inventory levels, help manage requests for quote and work orders, levels of productivity even down to the level of tracking the percentage of time a particular machine tool is in use. These software packages can be very expensive to the point that they are out of reach of small and medium sized businesses. In recent years, with the advent of the Internet, Service Oriented Architectures (SOA), and web-based applications, the features of these expensive software applications have started being available to smaller business in a cost-effective manner. For this project, the goal is to create a small part of one of these possible web applications that would help a small manufacturing business track its raw materials inventory. While I do not know if there are existing web applications that serve the same purpose, assuming this project is properly designed and implemented it could easily be used by many businesses.

1.2 Project Description and Scope

I propose to create a database and user interface that would allow a manufacturing supplier to be able to track its raw materials inventory. This raw materials inventory tracking system (RMITS) will allow a supplier to track amounts of the raw materials it has on hand and allow those amounts to be dynamically adjusted when work orders are accepted and when materials are purchased. To complete a work order to produce a certain product, some amount of raw materials is required. When work orders are accepted, the inventory for the required material should be reduced from the available inventory. When a manager purchases additional raw materials the inventory for the newly purchased materials should be increased. The system should be able to show the current amount of raw materials currently on hand, show a historical record of when materials are added or removed from inventory and for what purpose.

The RMITS database will need to store a variety of information about materials, work orders, and raw material purchases. The information about raw materials includes, but may not be limited to:

- material type (various metals, plastics, ceramics, etc)
- material shape (bars, rods, etc)
- material size
- amount of material.

The information about inventory purchases can include, but may not be limited to:

- the material purchased
- the vendor
- the price paid for the material
- the date of the purchase.

The information about work orders can include, but may not be limited to:

- the material(s) required for completing the work order
- the date the work order was accepted
- the completion date

- the customer
- the price charged to the customer for completing the work order.

A web-based user interface will be implemented to allow a user to provide the initial raw materials inventory amounts, to enter in information about work orders that have been accepted, and to enter in information about inventory purchases. The user interface should also provide reporting functionality for the amounts of raw materials on hand and a transaction history of materials. For this prototype project, the user interface will designed for a "single user." By this I mean there will not be use accounts or tracking of activity based upon an individual person using the website.

NOTE: This project is not being designed with any particular manufacturer or client in mind. It is being designed as a generic system. In this document, references to manufacturing employees and business owners are fictional.

1.3 Software Used to Prepare This Document

This document was written primarily with LibreOffice Writer (http://www.libreoffice.org), a free, open-source office suite. The diagrams, including the ER Diagram, presented in this document were created using Microsoft Visio 2010. Screenshots from the implementations were edited using MS Paint.

Chapter 2 Study of System Functional Requirements

2.1 Introduction

To help evaluate the needs and requirements of the manufacturing company and its employees who will be using the RMITS system, a study of the system's functional requirements is performed. Since this project is not being designed with any one particular company or a particular set of users in mind, the interviews and business forms provided in this chapters are fictional. Section 2.2 includes fictional interviews with the manufacturing business' owner, accountant and head machinist. Section 2.3 shows example forms for what the manufacturing company's employees and customers would fill out to provide to update information about the current raw materials inventory and projects to be performed. Section 2.4 shows example forms that produces information about the current raw materials inventory for the company and projects performed by the company.

2.2 Summary of Interview and Study

2.2.1 Business Owner

As a small machining shop, we've never had a complete computer system to track all aspects of the business. The biggest thing we are lacking now is a way to track our raw materials inventory. If I want to know how much of a particular material we have on hand I'd have to go back to the store room and physically count it. So when we get a work order in from a customer we don't necessarily know if we have the materials on hand to complete the work order or if we need to place an order with one of our vendors. This also affects how we bid on RFQs since we don't have good accounting for the cost of what we have in stock versus what we need to order.

2.2.2 Accountant

I am responsible for determining the amount of profit the company earns. One difficulty in doing this is knowing, when we complete a work order, how much it cost in raw materials. If we purchase materials specifically for the work order then figuring the cost out is trivial. However, if the machinist pulls the material from the store room it is much harder to determine how much the material cost is since we don't have a good method for tracking our inventory. I would like to calculate the cost of the materials used for a work order using a FIFO method.

2.2.3 Machinist

One of the big issues I face is making sure I am using the correct sized material for the project I'm working on. We don't want to create too much waste. So if I need 10 pieces of a 5 by 5 steel block I want to use that instead of a 7 by 8 steel block. The problem is I have to hunt around in the storage room to see what the different sizes and shapes of materials we have are. If I could see a list that shows for sure what is available that would be very useful.

2.3 Functional Requirements: Input Business Forms and Screens

2.3.1 Purchase Order Form

Vendor Name	**Contact Person**	**Phone Number**
Belisle Mfg Supplies	Dan Johnson	512-555-1234

Quantity	**Stock Number**	**Material Type**	**Shape**	**Size**
25	A101	Aluminum	Box	5x3x1
10	P523	Plastic	Sphere	3.5

2.3.2 Purchase Receipt

Vendor Name	**Contact Person**	**Phone Number**
Belisle Mfg Supplies	Dan Johnson	512-555-1234
Purchaser Name	**Contact Person**	**Phone Number**
Tx State Mfg Group	John White	512-245-5486

Stock Number	**Quantity**	**Price (per item)**	**Comments**
A101	25	10.25	
P523	10	N/A	Out of Stock

2.3.3 Request for Quote

Customer Company Name	Contact Person	Phone Number	Order Number
Able Tool	James Douglas	512-555-8745	154785

Part Name	Material(s) required	Quantity	Process(es) Required
Firing Pin	Alloy Steel	10	Milling, Turning, Heat Treating

Details: This is a RFQ for a prototype run of firing pins for high caliber machine guns. CAD files and additional specifications are attached to this RFQ. The specifications are not for distribution to third parties.

2.4 Functional Requirements: Output Business Forms and Screens

2.4.1 Inventory Report

Quantity	Material Type	Shape	Size
10	Aluminum	Box	5x3x0.5
22	Alloy Steel	Cylinder	1.5x10
1	Alloy Steel	Cylinder	1.5x5

2.4.2 Work Order

Customer Company Name	Contact Person	Phone Number	Order Number
Able Tool	James Douglas	512-555-8745	154785

Part Name	Material(s) required	Quantity	Billing amount
Firing Pin	Alloy Steel	10	$100

Process(es) Required	Machinist Assigned To	Specifications
Milling, Turning, Heat Treating	Dave Henderson	See attached CAD diagram.

2.4.3 Project Report

Customer Company Name	Contact Person	Phone Number	Order Number
Able Tool	James Douglas	512-555-8745	154785

Part Name	Material(s) required	Quantity	Machinist Assigned To
Firing Pin	Alloy Steel	10	Dave Henderson

Amount Material Used	Material Type	Shape	Size
12	Alloy Steel	Cylinder	0.5x4
1	Alloy Steel	Cylinder	0.25x4

Machinist Comments: I attempted to first use a smaller diameter material so there would be less material removal involved. It turns out the amount of work involved in fabricating the part put too much stress on the small workpiece and it broke. I reverted to a larger diameter workpiece which required more machining. After a couple of trial runs, I was able to complete the parts without breaking the workpiece.

Chapter 3 External Scheme (User's Views)

3.1 Diagram of System Model of the Project

Illustration 1 shows an example page flow diagram as a model of the system structure. The web-based system will originate with a home page. From there the user will be able to view a list of all reports and all forms available in the system.

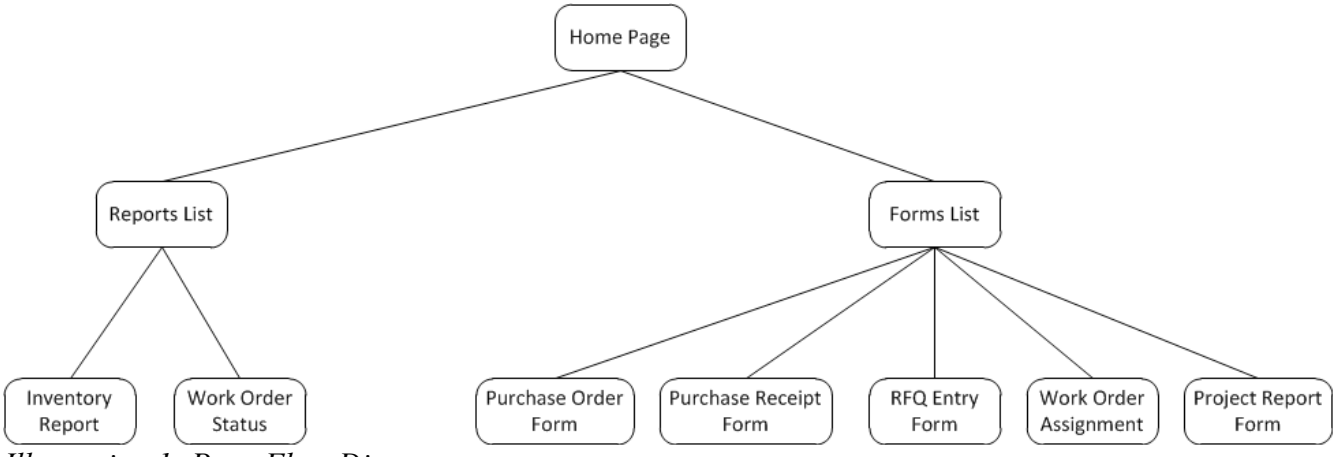


Illustration 1: Page Flow Diagram

3.2 Design of the User Interface for User's View

The user interface which will provide all of the user reports and screens will be developed with PHP on

an Apache web server. The layout of each page on the site will include a set of navigational menus and links on the left-hand side of the web page. There will be two sets of navigational menus. The first will be for data-entry forms which will include options to enter a purchase order, RFQ, and work order information. The second menu will be for reports which will include an inventory report, work order status reports, and possibly others.

3.3 Design of User Reports and Screens of the User's Views

The website which contains the user interface to RMITS will provide a number of user reports and screens for filling in forms for data entry. This section provides a list of those reports and form screens with brief descriptions of their purpose.

- Inventory Report: The inventory report will provide a list of all raw materials that are currently being stored with the manufacturer. The report will essentially be a table showing, for each material, the material's type, shape, dimensions, and how many of this raw material is on hand. The column headers for this table will be clickable to allow for the information in the table to be sorted based upon the header that was clicked.
- Work Order Status: This is a report to show the information available about a particular work order. The report will include information collected about the work order by that period of time which will include at least the RFQ information. Depending on the status of the work order it may include RFQ bid amount, machinist assignment and specifications, and final completed project information.
- Purchase Order Form: This is a form that will allow the user to enter information about a purchase order. This form should be filled out just prior to sending the order to the vendor. The information provided will include that materials that are expected to be purchased.
- Purchase Receipt Form: When a vendor fulfills a purchase order and delivers the materials to the manufacturer, the purchase receipt form is filled out. This form will include the final price of the purchased materials and quantities that were delivered. Any comments from the vendor about the materials purchased can also be provided.
- RFQ Entry: When an RFQ is sent to the manufacturer from a customer, the RFQ's information can be provided with this form. It will include information about the part to be produced, the material it is made out of, the number of parts needed, processes required to produce it and any additional comments about the RFQ.
- Work Order Assignment Form: When a bid for an RFQ has been accepted by the customer, the RFQ becomes a work order that needs to be completed. The first step to complete the work order is to assign it to a machinist. There will also be a comment section for any additional information that needs to be supplied at this phase.
- Project Report Form: When a work order has been completed this form is used to finalize the information about the status of the project. It can include a list of materials actually used in completing the project and any comments by the machinist in charge of project.

Chapter 4 Conceptual Schema and Logical Model of the System

4.1 Descriptions of All Relations and Attributes

There are six relations, or tables, required to create the database for this project. These relations are: Customer, Projects, Materials, Vendor, PurchaseOrders, MaterialsPurchased, and MaterialsUsed. These should be all the relations needed to store and manage the data for a single-user RMITS. Details about each of these relations and their attributes are found in the subsections below.

4.1.1 Customer Relation

The customer relation contains information about the customers of the manufacturer. Customers will submit 'Requests for Quotes' (RFQ) to the manufacturer. If the quote by the manufacturer is approved, a work order is created, the part(s) produced and shipped to the customer. The attributes for the Customer relation are:

- CustomerID: This is the primary key of the relation. It is a unique identifier for the customer in the RMITS.
- CompanyName: This is the name of the company represented by the customer. This is an optional field since some customers may be individuals without a formal company name.
- ContactPerson: This is the name of the person who is the primary contact with the customer. For companies, this may be an individual who is in charge of outsourcing work to manufacturers or some other contact person. For individuals, this will be the individual's name.
- PhoneNumber: This is for the phone number for the contact person for the customer.

4.1.2 Projects Relation

The projects relation contains information about projects proposed or worked on by the manufacturer. A project will be created when a customer submits a RFQ. As a bid is made on the RFQ, machinist(s) are assigned and the project completed additional information about the project is updated. The attributes for the Projects relation are:

- ProjectID: This is the primary key of the relation. It is a unique identifier for the project in the RMITS.
- CustomerID: This is a foreign key linked to the *Customer* relation. It specifies the particular customer this project is for.
- PartName: This is the name of the part that is to be produced as part of the project. This can be used to help describe the part.
- Processes: Every part will require one or more manufacturing processes to be performed to produce the part. This attribute has a string value and may contain multiple processes in that string. Currently, there are no formatting requirements for this field.
- Quantity: This is an integer value that specifies how many units of the part need to be produced for this project.

- RFQDetails: This is an optional attributed that can contain a textual description, special details or instructions about the project in the RFQ phase.
- BidAmount: This is a numerical value specifying how much the manufacturer has bid to complete the project. This is usually how much the customer will be charged for producing the part, should the customer accept the bid.
- Machinist: This is a text value specifying the name of the machinist assigned to this project. This attribute is normally assigned after a customer has accepted the manufacturer's bid.
- Specifications: This attribute can contain text providing any specifications or notes about the project. It may provide the specifications itself or may contain a link or location to CAD file(s) with the project specifications.
- Comments: This attribute provided a general area for providing comments about the project or notes not provided for elsewhere in the relation.

4.1.3 Materials Relation

The Materials relation contains information about the raw materials the manufacturer has in its inventory. Raw materials are used when completing work orders (projects) and are restocked after placing an order to purchase more. The attributes of the Materials relation are:

- MaterialID: This is the primary key of the *Materials* relation. It uniquely identifies the raw material information being stored.
- Type: This is the type of material. It is a text field that will contain values similar to: Bronze, Alloy Steel, Ceramic, Plastic, etc.
- Shape: This is the shape of the raw material. It is a text field that can contain values such as: bar, sphere, cylinder, ring, etc.
- Dimensions: This is the dimensions of the raw material. It is currently a text field to allow for the dimensions of any shape of material to be described.
- Quantity: This is the number of this raw material that is currently in the manufacturer's inventory. It will be decreased as projects are worked on and increased following material purchases.

4.1.4 Vendor Relation

The Vendor relation contains information about raw materials vendors the manufacturer may be customers themselves of. This relation contains information about the vendor itself including:

- VendorID: This is the primary key for the *Vendor* relation. It uniquely identifies the record for this vendor and its information.
- VendorName: This is the name of the vendor, usually the name of the company that raw materials are purchased from. It is an optional field.
- ContactPerson: This is an optional field containing the name of the contact person for the vendor.

- PhoneNumber: This is a required field that contains a text version of the vendor's phone number.

4.1.5 PurchaseOrders Relation

The PurchaseOrders relation contains information about purchase orders made with a particular vendor. It, currently, contains two attributes:

- PONumber: This is the purchase order number for the purchase. It is the primary key for the *PurchaseOrders* relation.
- VendorID: This is a foreign key related to the *Vendor* relation. It allows the purchase order to be specifically linked to be with a particular vendor.

4.1.6 MaterialsPurchased Relation

The MaterialsPurchased relation contains information about a particular raw material that is purchased as part of a purchase order. The attributes of this relation are:

- PONumber: This is part of a composite primary key and is also a foreign key linked to the *PurchaseOrders* relation. It specifies that the material being purchased is part of a particular purchase order, and therefore, was purchased from a particular vendor.
- MaterialID: This is the other part of a composite primary key and is a foreign key linked to the *Material* relation. It helps identify the particular material that was purchased.
- Quantity: This is an integer specifying the amount of the material that was ordered from and/or delivered by the vendor.
- Price: This is the cost that the vendor charged for the quantity of material that was purchased. NOTE: Currently, RMITS doesn't have a well defined way of handling shipping costs and taxes on purchases. It is left to the user how to include these additional costs in the price of materials purchased.
- Comments: This is an optional text field that can be used to specify additional comments about the purchase of material. The comments may be from vendor such as a message that the vendor was out of stock of this material when the purchase order was placed.

4.1.7 MaterialsUsed Relation

The MaterialsUsed relation contains information about raw materials that are used or consumed while completeing a work order (project). The attributes of the MaterialsUsed relation are:

- MaterialID: This is part of a composite primary key and is also a foreign key linked to the *Material* relation. It is used to show specifically which materials are being used while working on the project.
- ProjectID: This is the other part of a composite primary key and is also a foreign key linked to the *Projects* relation. It is used to show what materials were used while working on the specified project.
- Amount: This is an integer value specifying the amount of the raw material that was used while

working on the project.

4.2 Functional Dependencies for Normalized Relations

To assist in providing simplicity to the dependencies among attributes and creating relationships most relations include a single attribute as an identification number. This ID number is a unique number that is an auto-incremented value created by the DBMS. There are two relations that do not include a single, unique identification number: *MaterialsUsed* and *MaterialsPurchased*. A detailed list of the Primary and Foreign keys for each relation are:

- Customer Relation: The primary key is an auto-incrementing *CustomerID* attribute.
- Projects Relation: The primary key is an auto-incrementing *ProjectID* attribute. The relation also contains a foreign key, *CustomerID*, to link the project to a tuple in the Customer relation.
- Material Relation: The primary key is an auto-incrementing *MaterialID* attribute.
- Vendor Relation: The primary key is an auto-incrementing *VendorID* attribute.
- PurchaseOrder Relation: The primary key is an auto-incrementing *PONumber* attribute.
- MaterialsUsed Relation: The primary key for this relation is a composite key composed of two foreign keys: *MaterialID* from the *Material* relation and *ProjectID* from the *Projects* relation.
- MaterialsPurchased Relation: The primary key for this relation is a composite key composed of two foreign keys: *MaterialID* from the *Material* relation and *PONumber* from the *PurchaseOrder* relation.

4.3 Example Tuples for All Normalized Relations

The following subsections show each of the relations in the RMITS with sample data.

4.3.1 Customer Table

CustomerID	CompanyName	ContactPerson	PhoneNumber
1	Able Tool	James Douglas	512-555-8745
2	Lighting Automatic	Chris Stevenson	210-555-4874
3	DSN Innovations	Steve Filch	512-832-4587
4		Mitch Franke	512-658-8750
5	AccuTrex	Cody Ray	210-547-8874

4.3.2 Projects Table

ProjectID	CustomerID	PartName	Processes	Quantity
1	1	Firing Pin	Milling, Turning, Heat Treating	10
2	2	Hinge Joist	Milling, Drilling	100
3	3	Gearbox	Milling, Drilling, Heat Treating	5
4	3	Piston	Turning, Milling	35
5	4	Pulley	Milling	50
6	5	Curtain Rings	Milling	1000

RFQDetails	BidAmount	Machinist	Specifications
This RFQ is a prototype run... (blah, blah, blah)	$100.00	Dave Henderson	See CAD files.
	$1,550.00		
Private RFQ, no other bidders.	$500.00	John Harvey	No detailed specifications
Private RFQ, no other bidders.	$500.00	John Harvey	No detailed specifications
		Dave Henderson	See CAD files.
Bid via Infoneer's Mfg Marketplace	$500.00	Dave Henderson	See CAD files.

Comments
I attempted to first used... (blah, blah, blah)
Did not win bid
Created CAD files and specifications in creation of prototypes. Materials provided by customer
Created CAD files and specifications in creation of prototypes. Materials provided by customer

4.3.3 MaterialsUsed Table

MaterialID	ProjectID	Amount
3	1	1
2	1	12
8	6	1010
4	2	103
2	5	6

4.3.4 Materials Table

MaterialID	Type	Shape	Dimensions	Quantity
1	Aluminum	Box	5x3x0.5	10
2	Alloy Steel	Cylinder	0.5x4	22
3	Alloy Steel	Cylinder	0.25x4	33
4	Aluminum	Box	5x3x1	1
5	Plastic	Sphere	3.5	0
6	Stainless Steel	Ring	2x0.5	10
7	Stainless Steel	Ring	2.5x0.5	8
8	Plastic	Ring	2x0.5	0
9	Stainless Steel	Cylinder	3x10	3

4.3.5 Vendor Table

VendorID	VendorName	ContactPerson	PhoneNumber
1	Belisle Mfg Supplies	Dan Johnson	512-555-1234
2	Steel Mat Products	Jan Davidson	210-555-8745
3	Alcoa		888-555-7458
4		Jason Steel	512-999-7364
5	High Capacity Plastics	Donald Trumpet	801-234-5934

4.3.6 PurchaseOrder Table

PONumber	VendorID
12548	1
12549	5
12550	3
15551	2

4.3.7 MaterialsPurchased Table

PONumber	MaterialID	Quantity	Price	Comments
12548	4	25	$10.25	
12548	5	10	$0.00	Out of stock
12549	5	10	$18.50	
12550	1	20	$20.00	
12551	4	9	$12.75	
12551	7	7	$14.50	

4.4 Descriptions of all Required Relationships

There are a number of relationships between the relations (tables) in the RMITS database. The relationships in the system are:

- Customer-Projects: This is a one-to-many relationship that allows projects to be linked to a particular customer. This means that a project belongs to a single customer. A single customer can have multiple projects with the manufacturer.
- Projects-MaterialsUsed: This relationship is part of a many-to-many relationship between Projects and Materials. A project may use multiple types of raw materials and a single raw material may be used in multiple projects. This particular relationship describes a single project and the many materials it may use.
- Material-MaterialsUsed: This relationship is part of a many-to-many relationship between Projects and Materials. A project may use multiple types of raw materials and a single raw material may be used in multiple projects. This particular relationship describes a single material and the many projects it may be used in.
- Vendor-PurchaseOrder: This is a one-to-many relationship that allows various purchase orders for raw materials to be linked to the vendor the materials were purchased from.
- PurchaseOrder-MaterialsPurchased: This relationship is part of a many-to-many relationship between *PurchaseOrders* and *Materials*. A purchase order may be for the purchase of multiple types of raw materials and a raw material may be purchased multiple times through different purchase orders. This particular relationship describes a single purchase order and the material(s) that are purchased with it.
- MaterialsPurchased-Materials: This relationship is part of a many-to-many relationship between *PurchaseOrders* and *Materials*. A purchase order may be for the purchase of multiple

types of raw materials and a raw material may be purchased multiple times through different purchase orders. This particular relationship describes a single material and the different purchase orders it was obtained through.

4.5 ER Diagram of the System

Illustration 2 shows the entity-relationship diagram of the relationships for RMITS. It contains the relations described in section 4.1 and the relationships between them.

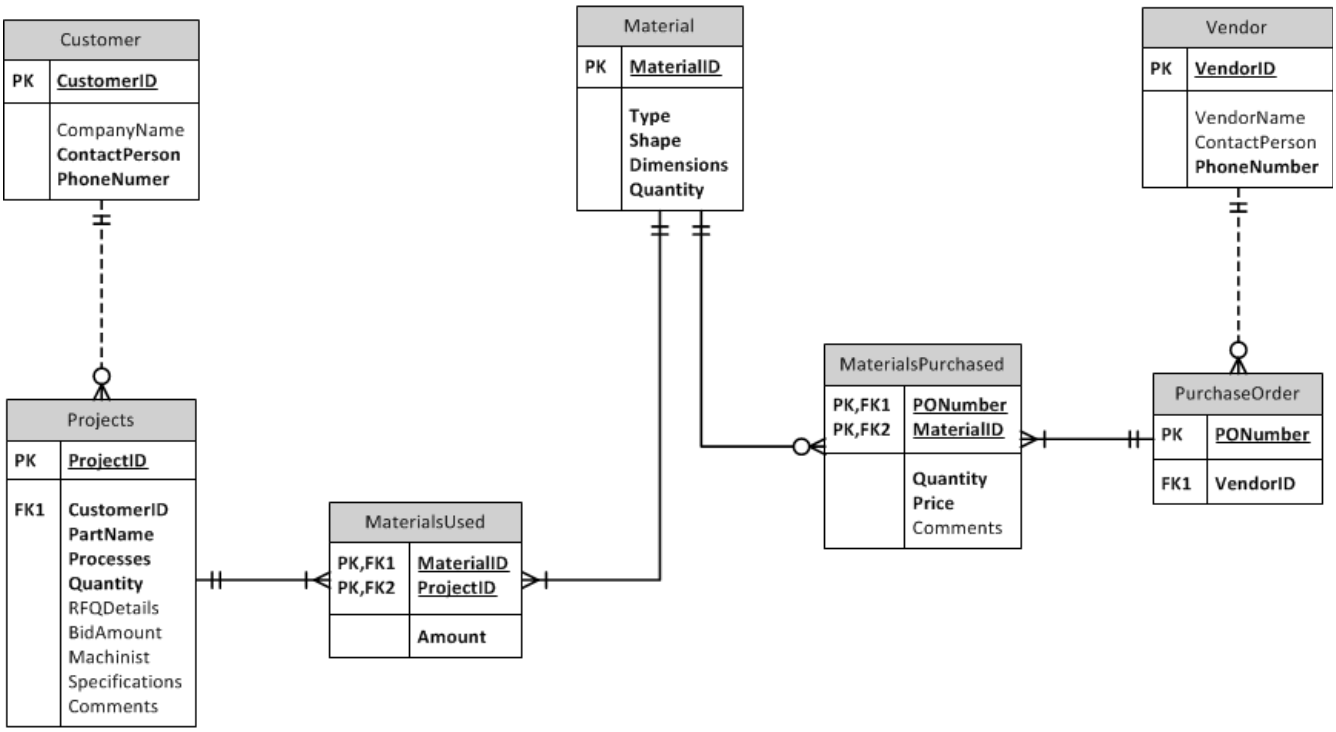


Illustration 2: ER Diagram of Database System

Chapter 5 Data Dictionary

5.1 Relations, Attributes, Domains, Comments

This section provides a list of attributes for each of the relations in the RMITS system. Detailed descriptions of these relations and attributes can found in Section 4.1.

5.1.1 Customer

Attribute	Domain	Comments
CustomerID	Integers greater than 0	Can be auto-incremented by DBMS.
CompanyName	Any company name	Can be blank.
ContactPerson	Any person's name	Can be blank.
PhoneNumber	A phone number	Required field.

5.1.2 Projects

Attribute	Domain	Comments
ProjectID	Integers greater than 0	Can be auto-incremented by DBMS.
CustomerID	Integers greater than 0	Linked to the Customer table.
PartName	The name of a part	Required field.
Processes	Manufacturing processes	Requried field.
Quantity	Integers greater than 0	Requried field.
RFQDetails	General text.	Can be blank.
BidAmount	Dollar values	Can be blank.
Machinist	A person's name	Can be blank.
Specifications	General text	Can be blank.
Comments	General text	Can be blank.

5.1.3 Material

Attribute	Domain	Comments
MaterialID	Integers greater than 0	Can be auto-incremented by the DBMS.
Type	The material composition	Required field
Shape	Geometric shape of the raw material	Required field
Dimensions	The dimensions of the shape of the material	Required field
Quantity	Integers greater than 0	Required field

5.1.4 MaterialsUsed

Attribute	Domain	Comments
MaterialID	Integers greater than 0	Linked to Material table
ProjectID	Integers greater than 0	Linked to Project table
Amount	Integers greater than 0	Required field

5.1.5 Vendor

Attribute	Domain	Comments
VendorID	Integers greater than 0	Can be auto-incremented by the DBMS.
CompanyName	The name of a company	Can be blank
ContactPerson	The name of a person	Can be blank
PhoneNumber	A phone number.	Required field.

5.1.6 PurchaseOrder

Attribute	Domain	Comments
PONumber	Integers greater than 0	Can be auto-incremented by the DBMS
VendorID	Integers greater than 0	Linked to Vendor table

5.1.7 MaterialsPurchased

Attribute	Domain	Comments
PONumber	Integers greater than 0	Linked to PurchaseOrder table
MaterialID	Integers greater than 0	Linked to Material table
Quantity	Integers greater than 0	Required field
Price	Dollar values	Required field
Comments	General text	Can be blank

5.2 Attributes, Types, Descriptions, Restrictions

Attribute	Type	Description	Restrictions
CustomerID	Integer	Unique identifier for customers.	Greater than 0.
CompanyName (Customer)	String	Name of customer's company name.	
ContactPerson (Customer)	String	Contact person with customer.	
PhoneNumber (Customer)	String	Phone number for contact person or company.	Not null.
ProjectID	Integer	Unique identifier for projects.	Greater than 0.
PartName	String	Name of the project's part.	Not null.
Processes	String	Processes required to complete the project.	Not null.
Quantity (Project)	Integer	The number of parts to produce.	Greater than 0.
RFQDetails	String	Information about the RFQ.	
BidAmount	Double	Amount bid to earn the project.	Greater than 0.
Machinist	String	Person assigned to complete the project.	
Specifications	String	Project specifications.	
Comments (Project)	String	Additional comments about project.	
MaterialID	Integer	Unique identifier for materials.	Greater than 0.
Type	String	The material composition.	Not null.
Shape	String	The shape of the material.	Not null.
Dimensions	String	The size of the material.	Not null.
Quantity (Material)	Integer	The amount of material on hand.	Greater than 0.
Amount	Integer	Amount of materials used in a project.	Greater than 0.
VendorID	Integer	Unique identifier for vendors.	Greater than 0.
CompanyName (Vendor)	String	Company name of the vendor.	
ContactPerson (Vendor)	String	Contact person at the vendor.	
PhoneNumber (Vendor)	String	Phone number for the vendor.	Not null.
PONumber	Integer	Unique identifier for a puchase order.	Greater than 0.
Quantity (MaterialsPurchased)	Integer	Amount of materials purchased.	Greater than 0.
Price	Double	Cost of materials purchased.	Greater than 0.
Comments (MaterialsPurchased)	String	Additional information about purchases.	

5.3 Cross References

Attribute	Primary Table	Foreign Table
CustomerID	Customer	Project
ProjectID	Project	MaterialsUsed
MaterialID	Material	MaterialsUsed, MaterialsPurchased
VendorID	Vendor	PurchaseOrder
PONumber	PurchaseOrder	MaterialsPurchased

Chapter 6 Queries and RDBMS Implementation I

6.1 The Software, RDBMS and Platform Used

This first implementation of the RMITS was completed on a MySQL server with a user interface implemented in PHP.

6.2 Database Creation and Required Commands

The SQL commands to create the database and its associated tables are:

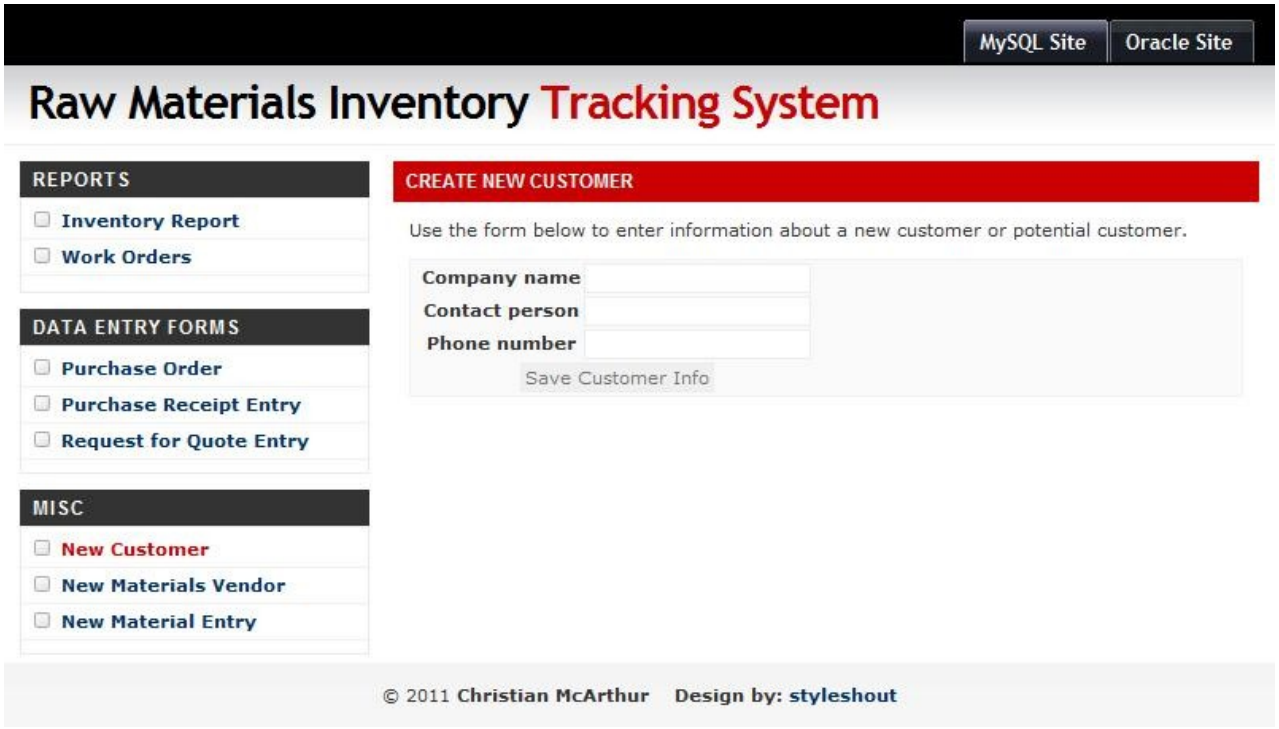
```
-- --------------------------------------------------------
--
-- Table structure for table `Customer`
--
CREATE TABLE `Customer` (
  `CustomerId` int(11) NOT NULL auto_increment,
  `CompanyName` text,
  `ContactPerson` text NOT NULL,
  `PhoneNumber` text NOT NULL,
  PRIMARY KEY  (`CustomerId`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
-- --------------------------------------------------------
--
-- Table structure for table `Material`
--
CREATE TABLE `Material` (
  `MaterialID` int(11) NOT NULL auto_increment,
```

```
  `Type` text NOT NULL,
  `Shape` text NOT NULL,
  `Dimensions` text NOT NULL,
  `Quantity` int(11) NOT NULL,
  PRIMARY KEY  (`MaterialID`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
-- --------------------------------------------------------
--
-- Table structure for table `MaterialsPurchased`
--
CREATE TABLE `MaterialsPurchased` (
  `PONumber` int(11) NOT NULL,
  `MaterialID` int(11) NOT NULL,
  `Quantity` int(11) NOT NULL,
  `Price` double NOT NULL,
  `Comments` text,
  PRIMARY KEY  (`PONumber`,`MaterialID`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
-- --------------------------------------------------------
--
-- Table structure for table `MaterialsUsed`
--
CREATE TABLE `MaterialsUsed` (
  `MaterialID` int(11) NOT NULL,
  `ProjectID` int(11) NOT NULL,
  `Amount` int(11) NOT NULL,
  PRIMARY KEY  (`MaterialID`,`ProjectID`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
-- --------------------------------------------------------
--
-- Table structure for table `Projects`
--
CREATE TABLE `Projects` (
```

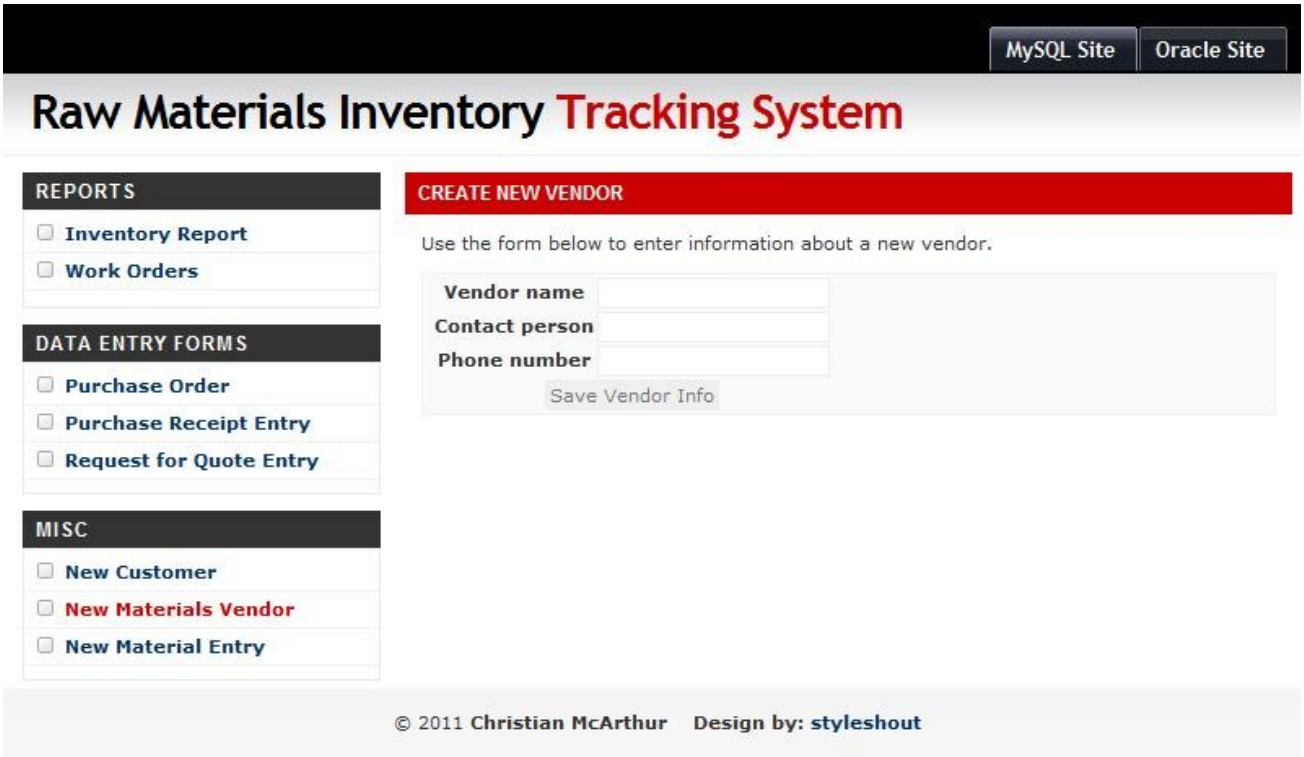
```
  `ProjectID` int(11) NOT NULL auto_increment,
  `CustomerID` int(11) NOT NULL,
  `PartName` text NOT NULL,
  `Processes` text NOT NULL,
  `Quantity` int(11) NOT NULL,
  `RFQDetails` text,
  `BidAmount` double default NULL,
  `Machinist` text,
  `Specifications` text,
  `Comments` text,
  PRIMARY KEY  (`ProjectID`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
-- --------------------------------------------------------
--
-- Table structure for table `PurchaseOrder`
--
CREATE TABLE `PurchaseOrder` (
  `PONumber` int(11) NOT NULL auto_increment,
  `VendorID` int(11) NOT NULL,
  PRIMARY KEY  (`PONumber`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
-- --------------------------------------------------------
--
-- Table structure for table `Vendor`
--
CREATE TABLE `Vendor` (
  `VendorID` int(11) NOT NULL auto_increment,
  `VendorName` text,
  `ContactPerson` text,
  `PhoneNumber` text NOT NULL,
  PRIMARY KEY  (`VendorID`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

6.3 User Interface Implementation

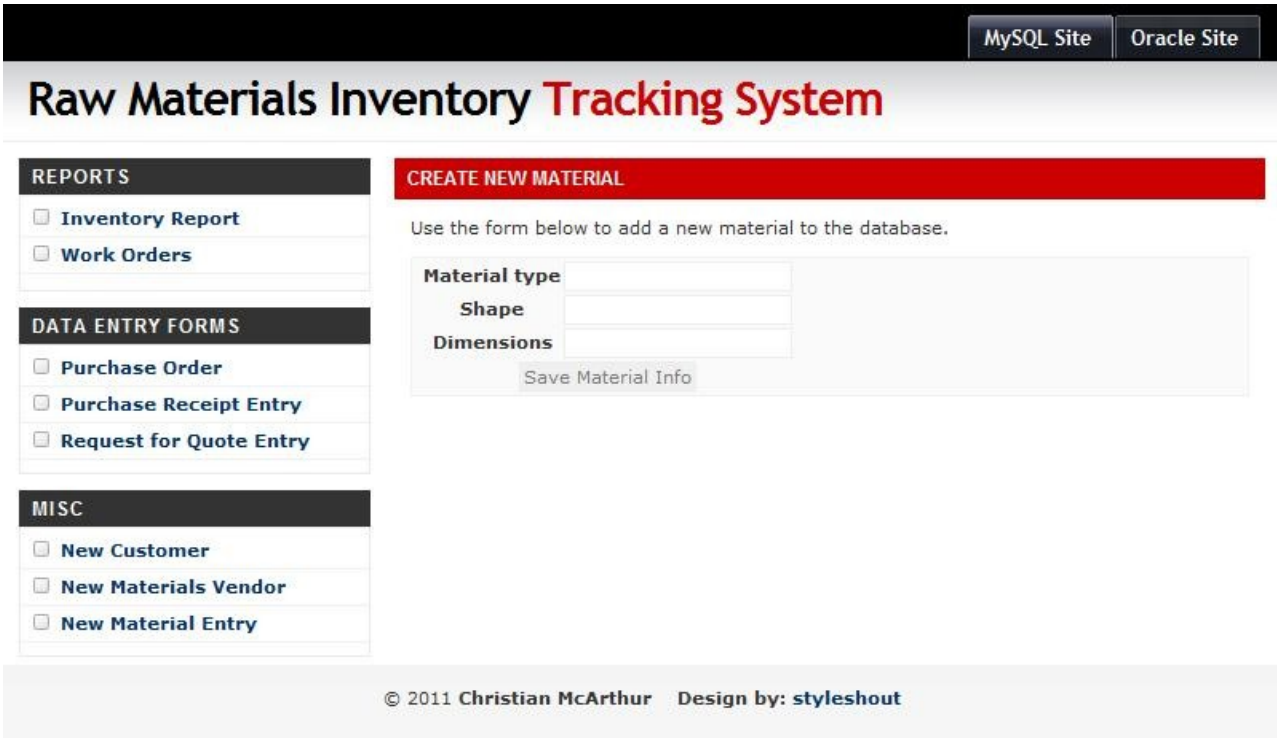
6.3.1 New Customer Form



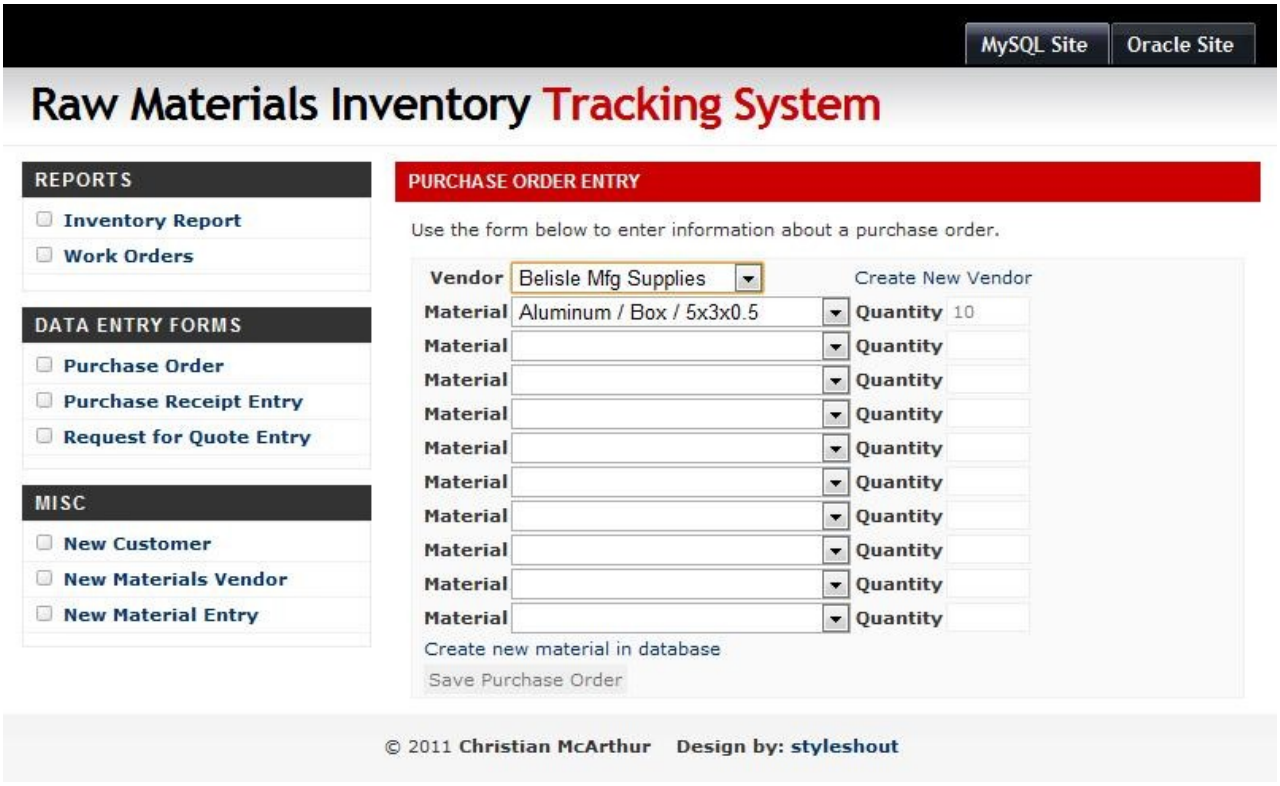
6.3.2 New Material Vendor Form



6.3.3 New Material Form

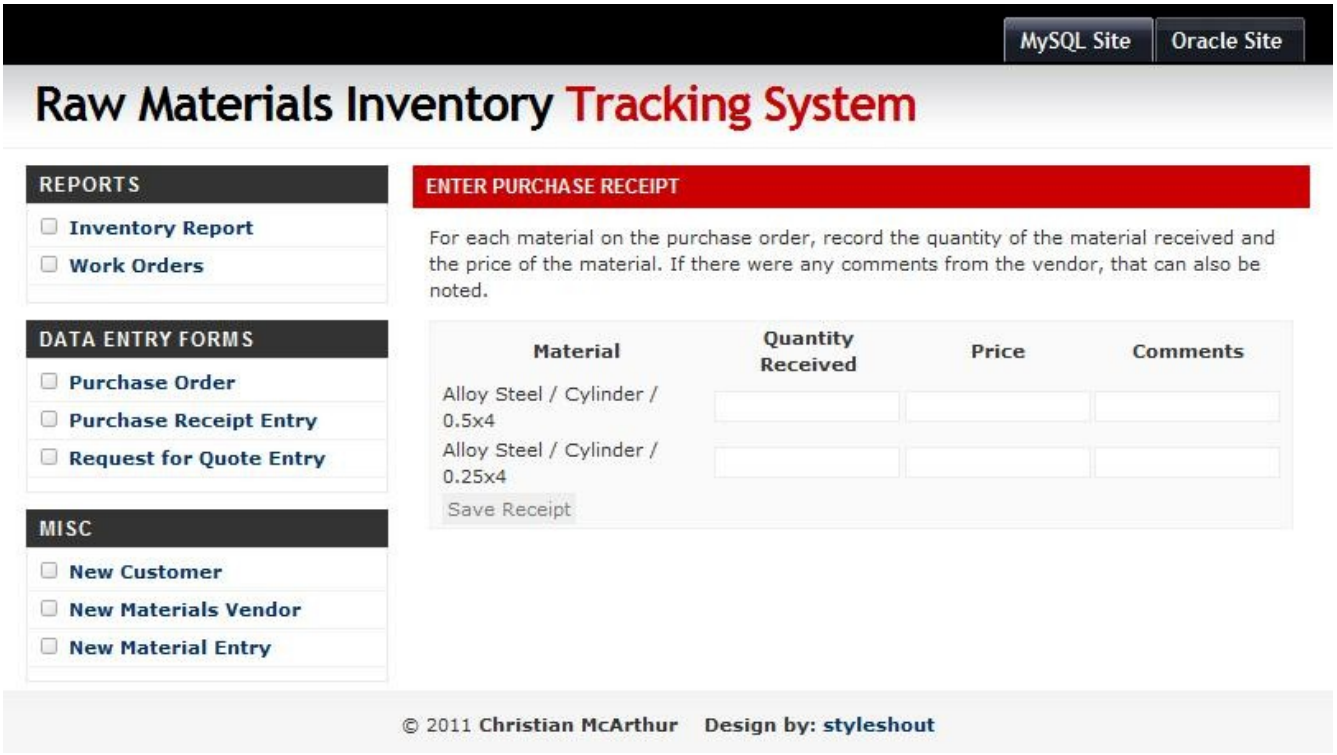


6.3.4 Purchase Order Form

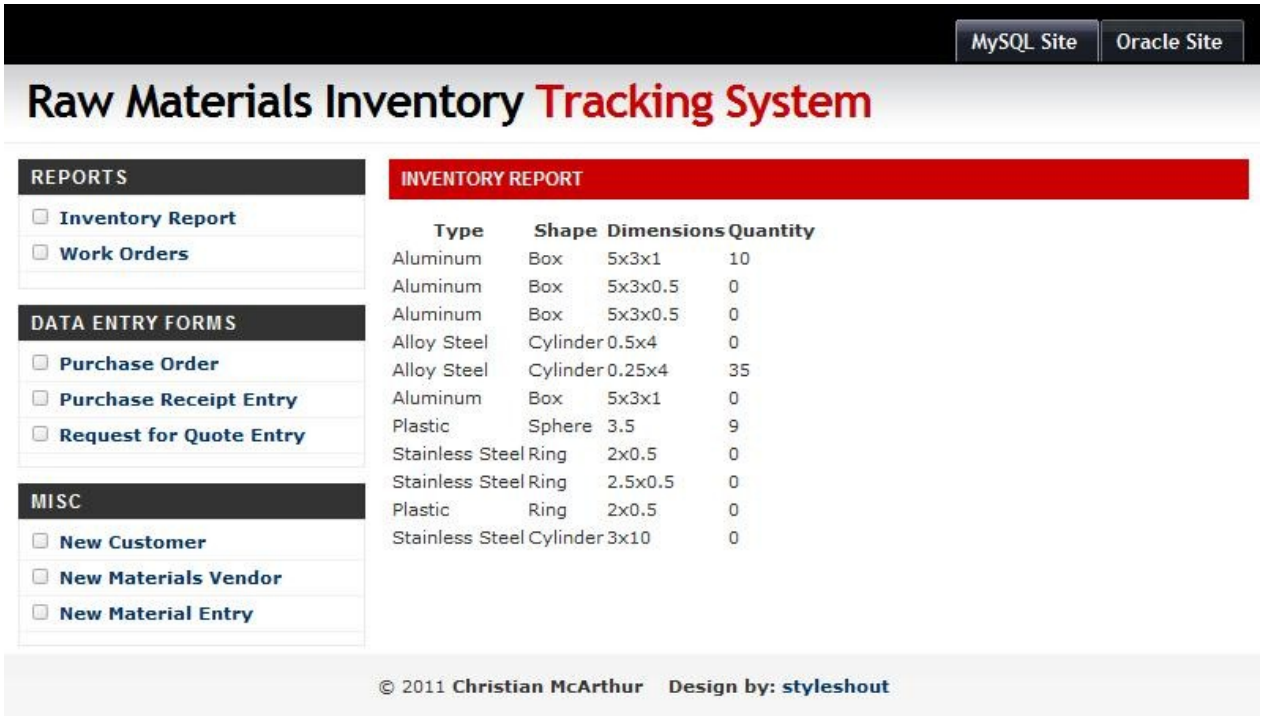


6.3.5 Purchase Order List

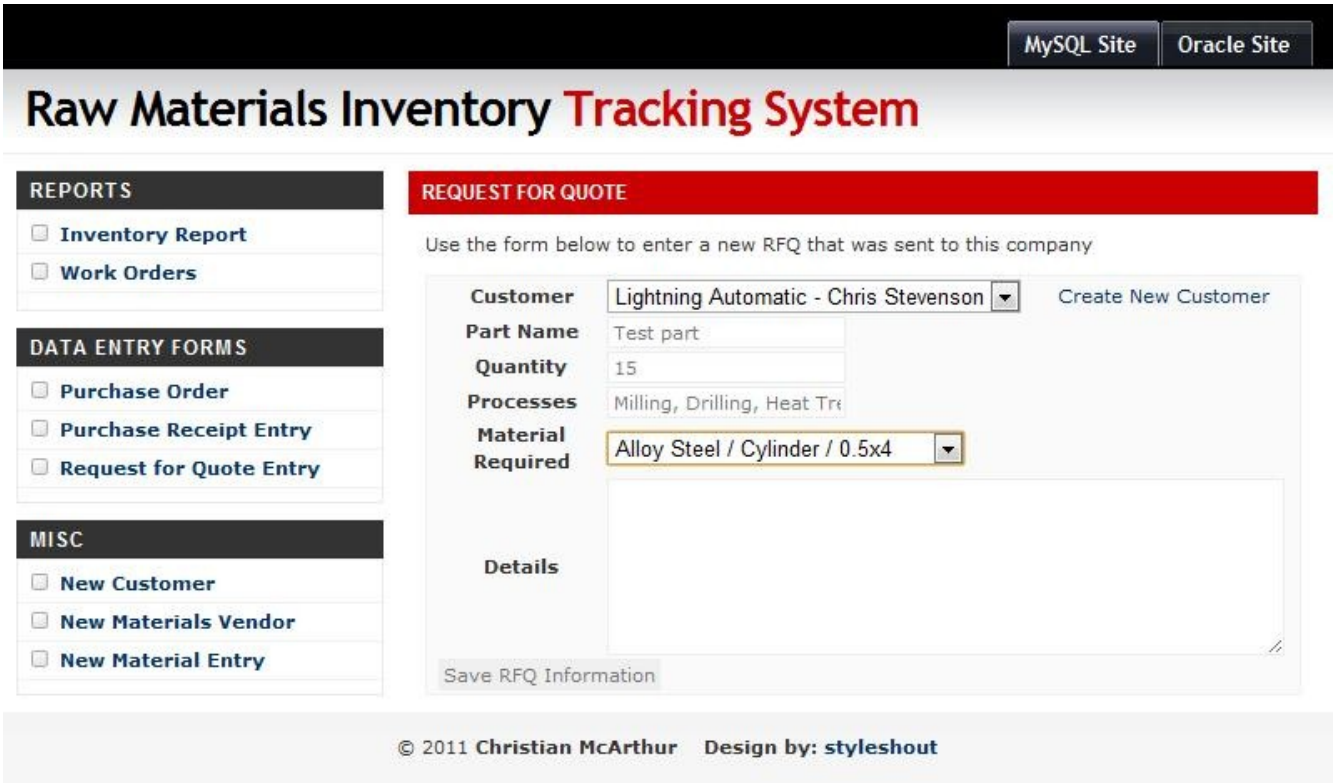
6.3.6 Purchase Receipt Form



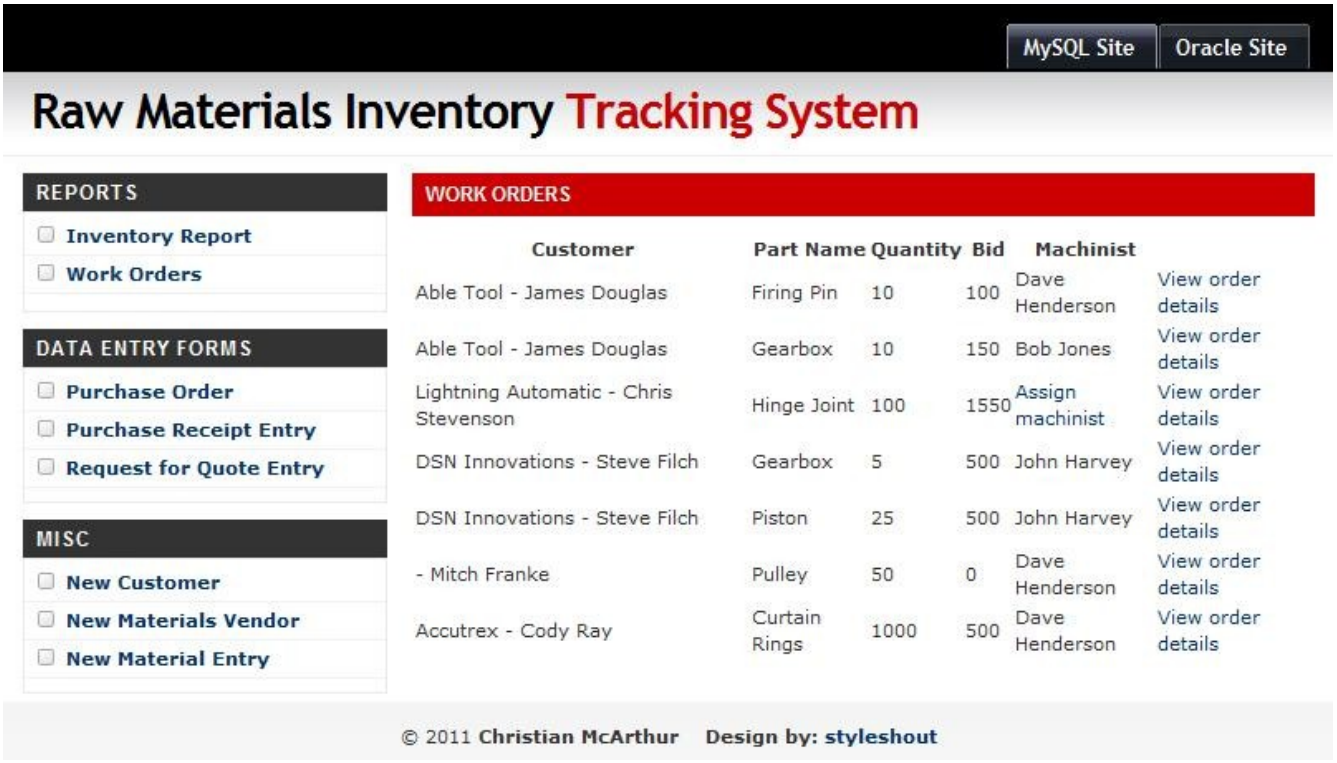
6.3.7 Inventory Report



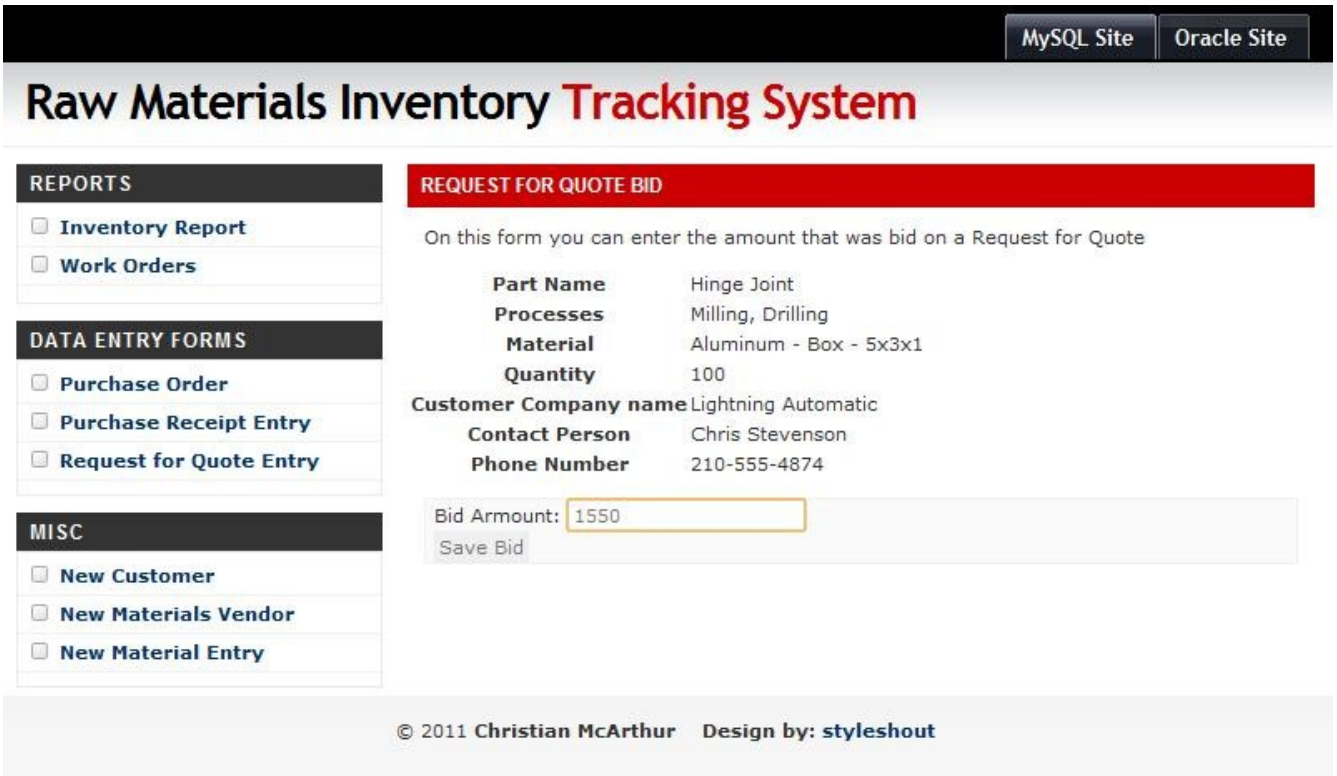
6.3.8 Request for Quote Form



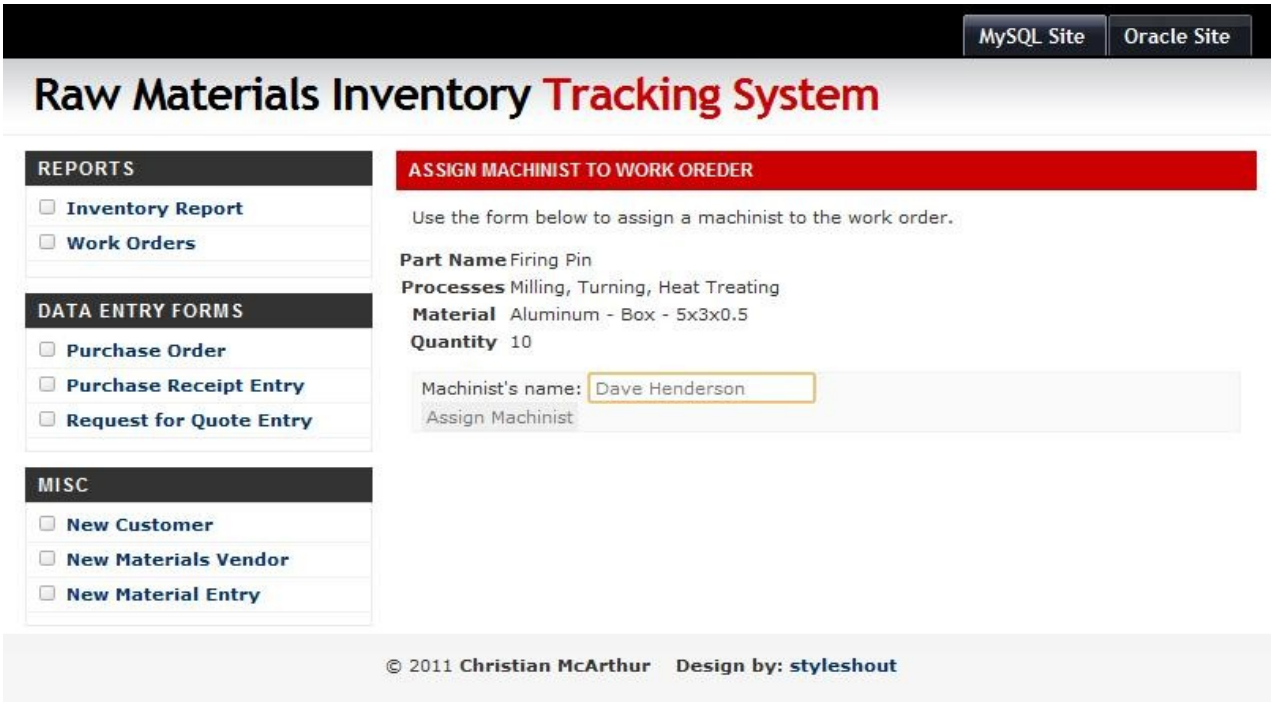
6.3.9 Work Order Report



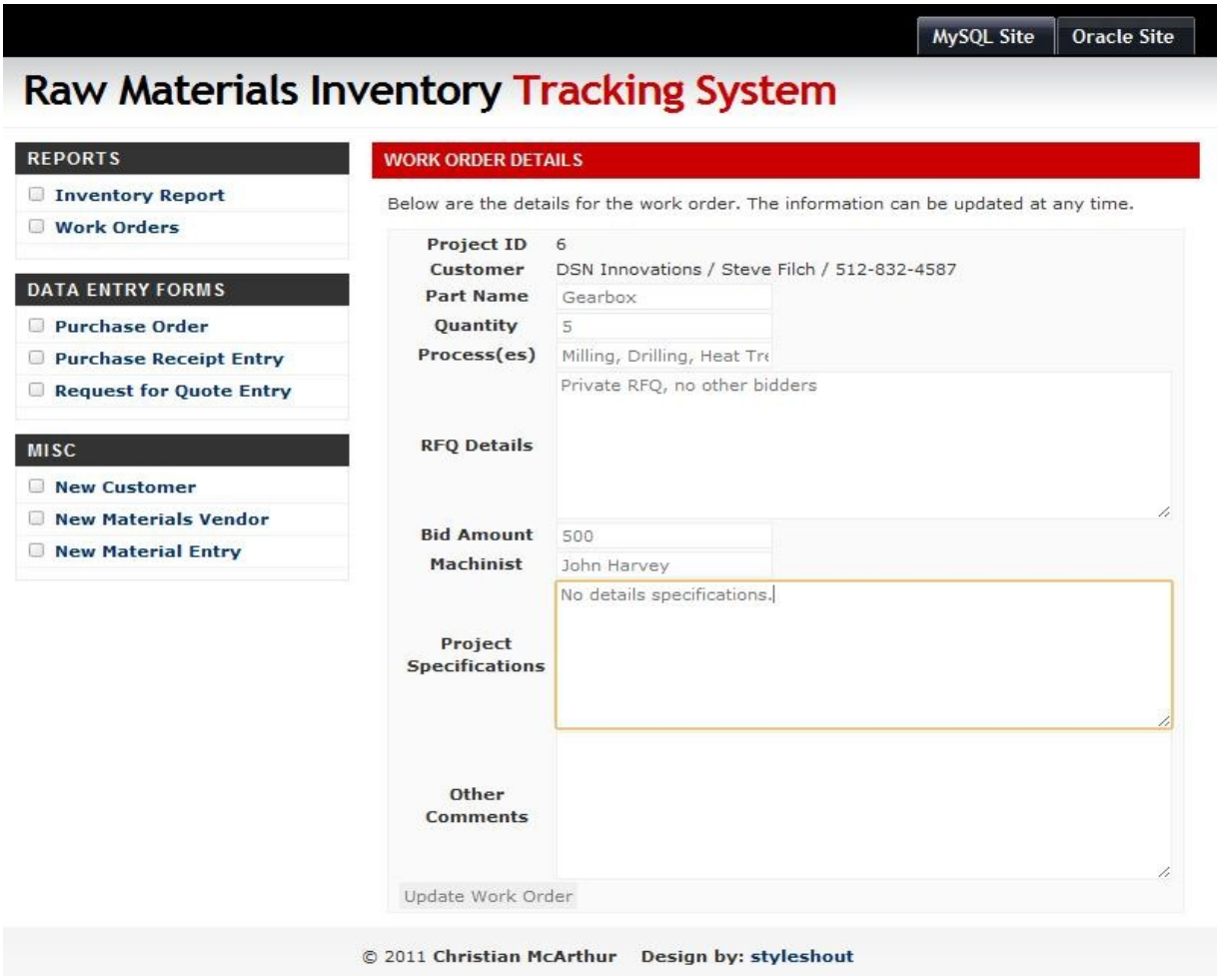
6.3.10 RFQ Bid Form



6.3.11 Work Order – Assign Machinist Form



6.3.12 Work Order Details



Chapter 7 Queries and RDBMS Implementation II

7.1 The Software, RDBMS and Platform Used

This implementation of the RMITS was completed using an Oracle database. The creation of tables was performed using the Linux SQL*Plus command line interface. The user interface for the end user was developed using Java with a Swing graphical user interface.

7.2 Database Creation and Required Commands

The SQL commands to create the database and its associated tables are:

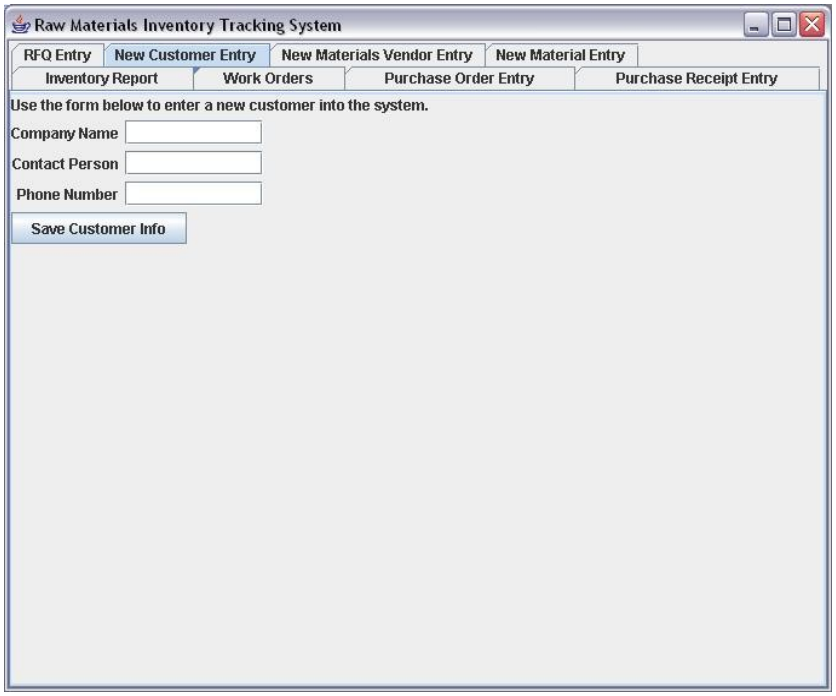
```
CREATE TABLE Customer (
   CustomerID NUMBER(4,0) NOT NULL,
   CompanyName VARCHAR2(200),
```

```
   ContactPerson VARCHAR2(200) NOT NULL,
   PhoneNumber VARCHAR2(80) NOT NULL,
   PRIMARY KEY (CustomerID)
   );
CREATE TABLE Material (
   MaterialID Number(4,0) NOT NULL,
   Type VARCHAR2(200) NOT NULL,
   Shape VARCHAR2(200) NOT NULL,
   Dimensions VARCHAR2(200) NOT NULL,
   Quantity NUMBER(4,0) NOT NULL,
   PRIMARY KEY (MaterialID)
   );
CREATE TABLE Vendor (
   VendorID NUMBER(4,0) NOT NULL,
   VendorName VARCHAR2(200),
   ContactPerson VARCHAR2(200),
   PhoneNumber VARCHAR2(80) NOT NULL,
   PRIMARY KEY (VendorID)
   );
CREATE TABLE PurchaseOrder (
   PONumber NUMBER(4,0) NOT NULL,
   VendorID NUMBER(4,0) NOT NULL,
   PRIMARY KEY (PONumber),
   FOREIGN KEY (VendorID) REFERENCES Vendor(VendorID)
   );
CREATE TABLE MaterialsPurchased (
   PONumber NUMBER(4,0) NOT NULL,
   MaterialID NUMBER(4,0) NOT NULL,
   Quantity NUMBER(4,0) NOT NULL,
   Price NUMBER(6,2),
   Comments VARCHAR2(800),
   PRIMARY KEY (PONumber, MaterialID),
   FOREIGN KEY (PONumber) REFERENCES PurchaseOrder(PONumber),
```

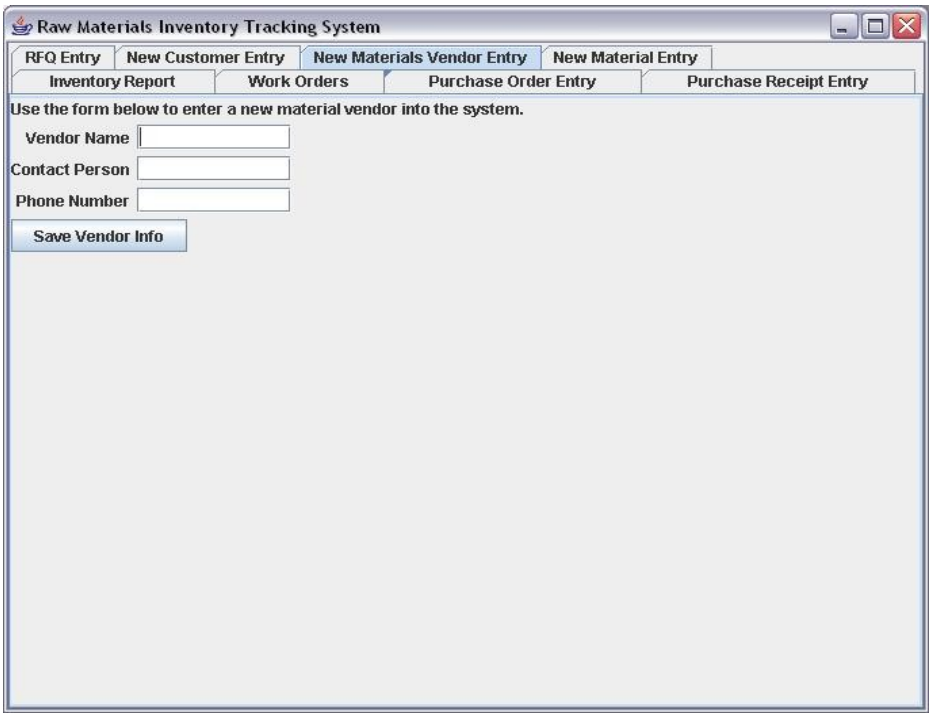
```
   FOREIGN KEY (MaterialID) REFERENCES Material(MaterialID)
  );
CREATE TABLE Projects (
   ProjectID NUMBER(4,0) NOT NULL,
   CustomerID NUMBER(4,0) NOT NULL,
   PartName VARCHAR2(200) NOT NULL,
   Processes VARCHAR2(200) NOT NULL,
   Quantity NUMBER(6,0) NOT NULL,
   RFQDetails VARCHAR2(800),
   BidAmount NUMBER(6,2),
   Machinist VARCHAR2(200),
  Specifications VARCHAR2(800),
  Comments VARCHAR2(800),
  PRIMARY KEY (ProjectID),
  FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID)
  );
CREATE TABLE MaterialsUsed (
   MaterialID NUMBER(4,0) NOT NULL,
   ProjectID NUMBER(4,0) NOT NULL,
   Amount NUMBER(6,0) NOT NULL,
   PRIMARY KEY (MaterialID, ProjectID),
   FOREIGN KEY (MaterialID) REFERENCES Material(MaterialID),
   FOREIGN KEY (ProjectID) REFERENCES Projects(ProjectID)
   );
```

7.3 User Interface Implementation

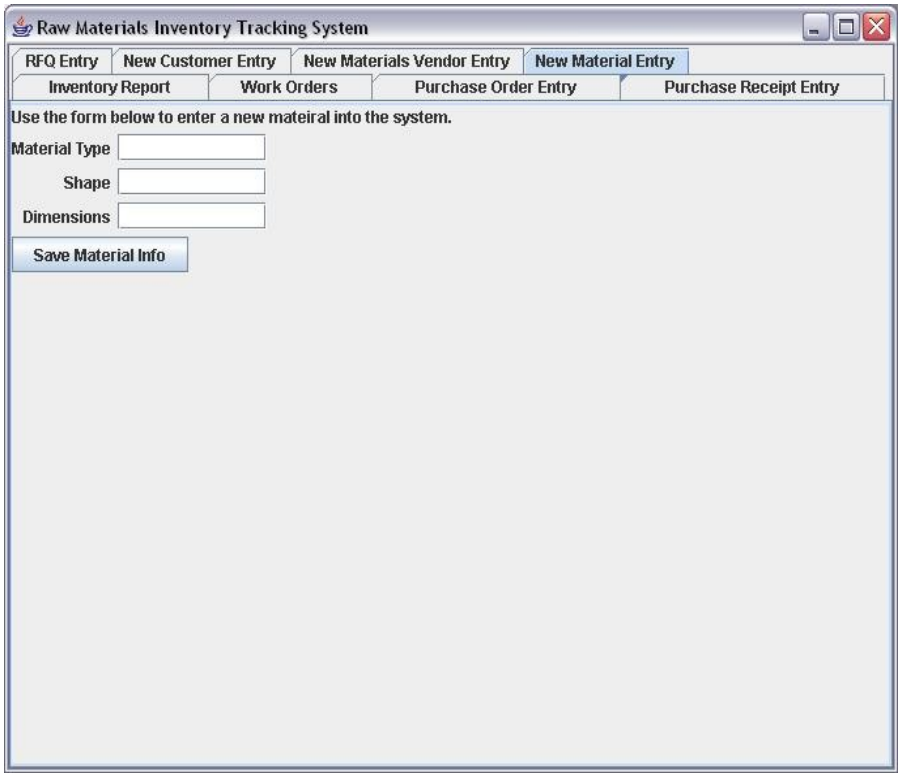
7.3.1 New Customer Form



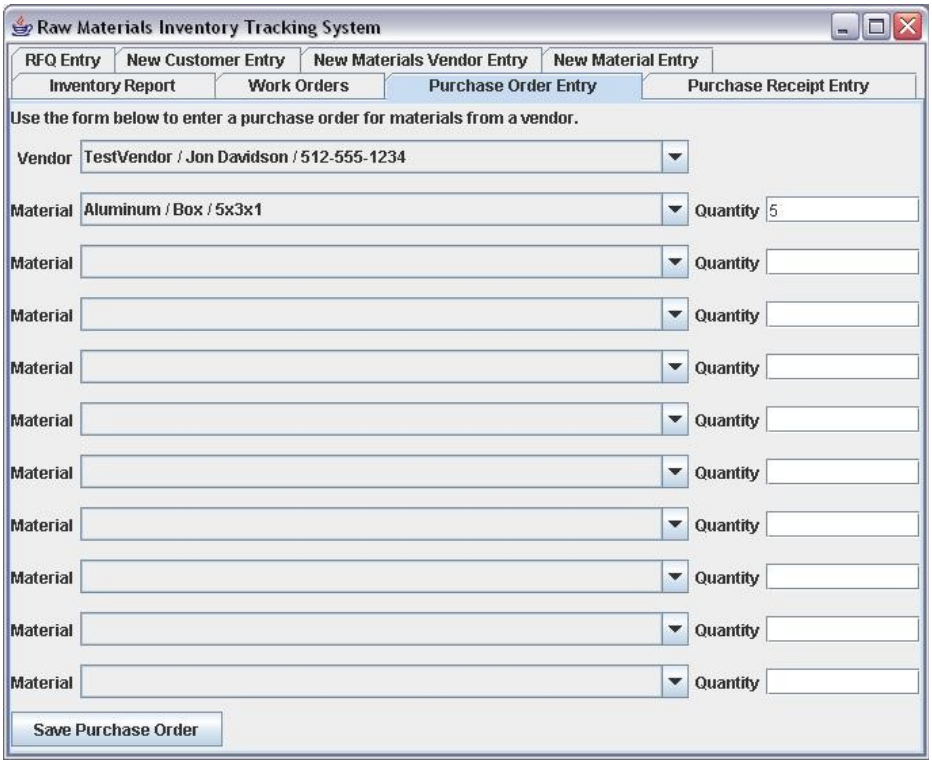
7.3.2 New Materials Vendor Form



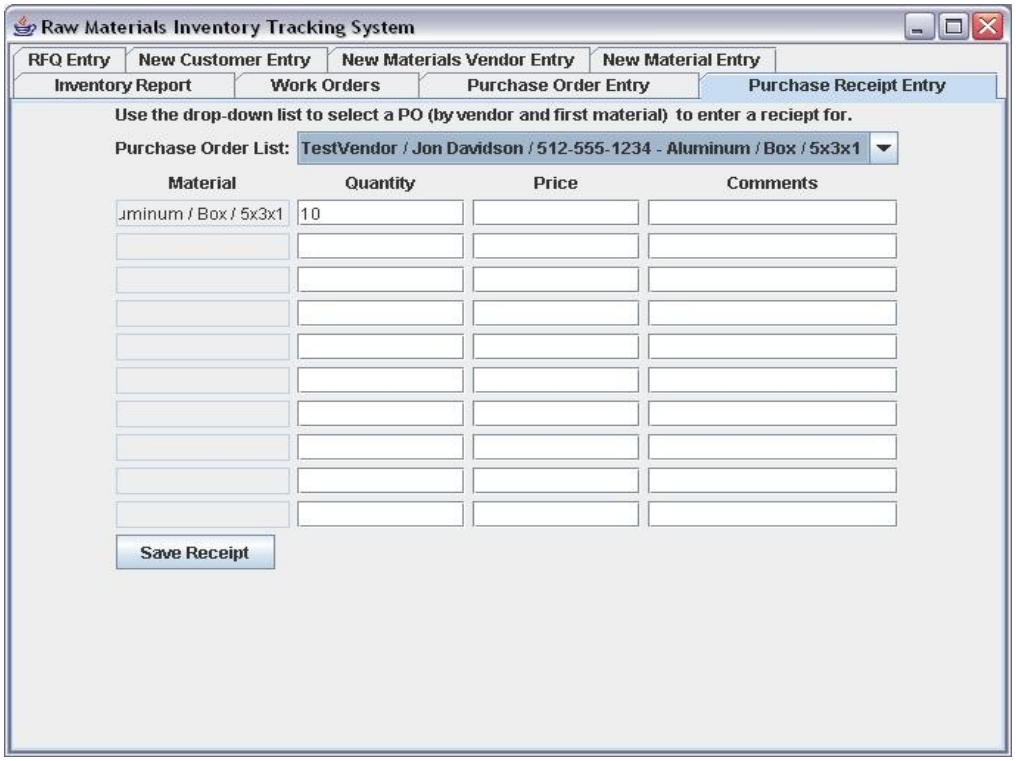
7.3.3 New Raw Materials Form



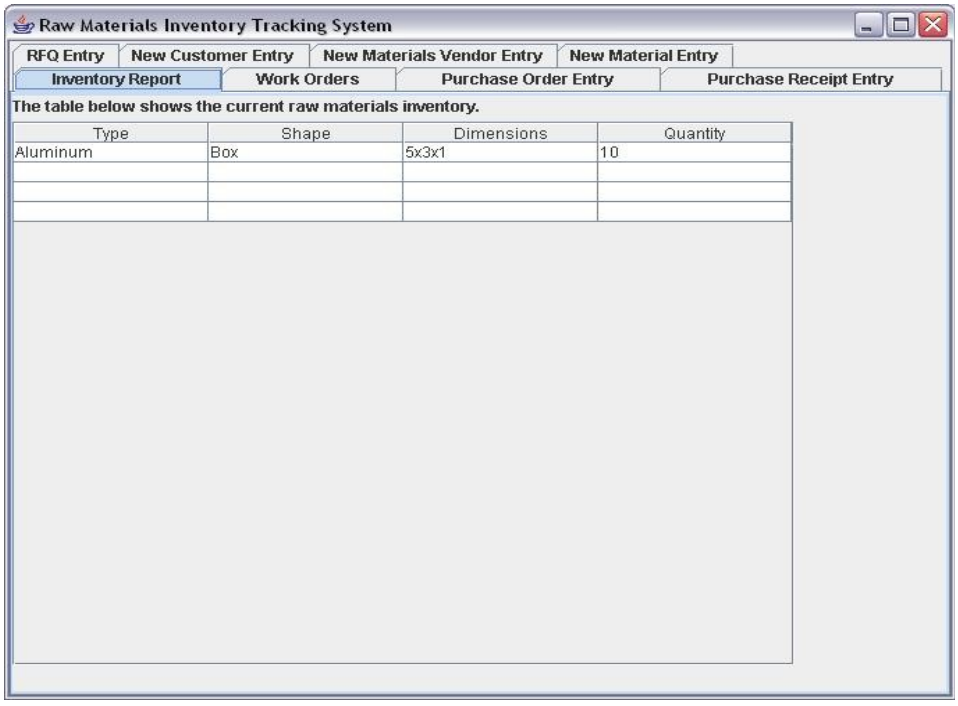
7.3.4 Purchase Order Entry Form



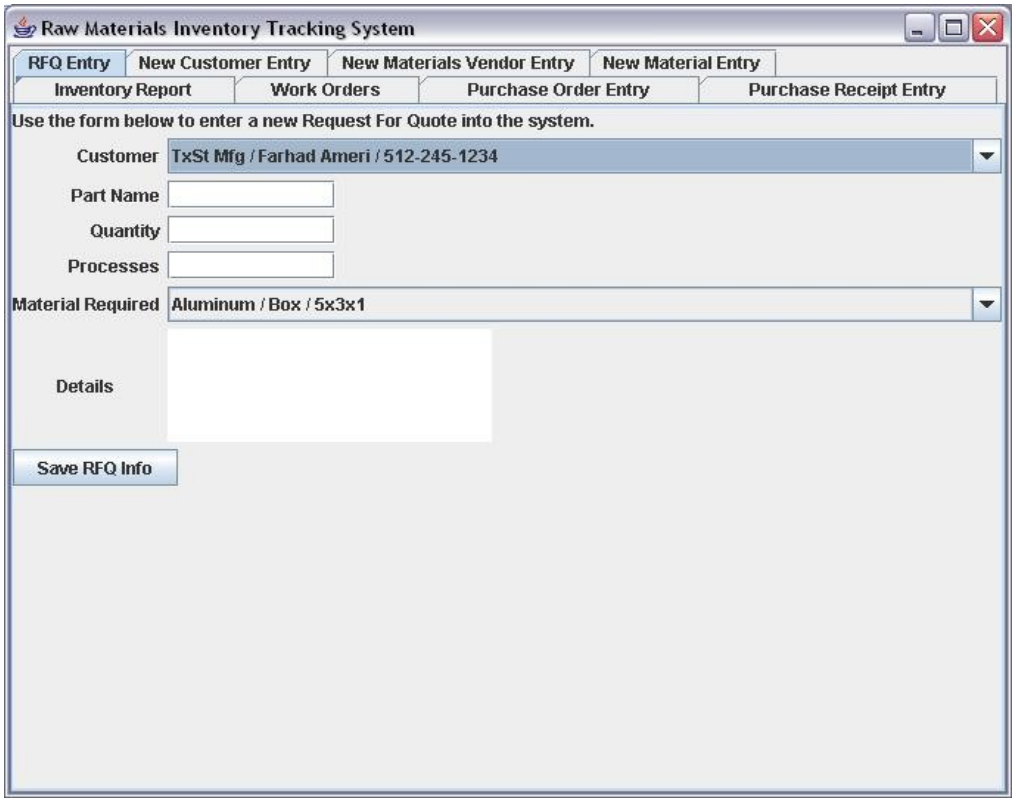
7.3.5 Purchase Receipt Form



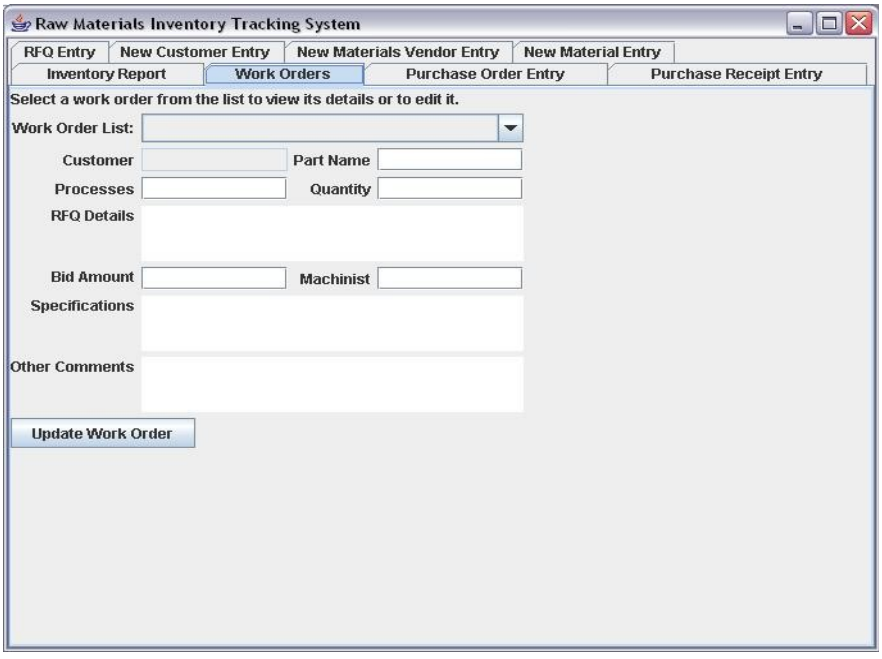
7.3.6 Inventory Report



7.3.7 Request For Quote Form



7.3.8 Work Order Details



Chapter 8 Conclusion

This project was unfortunately more of an academic exercise than a challenging one for me. The requirement of the end result including two different implementations caused me to be unable to really challenge myself as opposed to if I could focus on a single implementation. Two implementations were completed, the first being the MySQL and PHP implementation. Early on there was the plan to use the same PHP web-based interface for both DBMS implementations. However, late in the development cycle it was discovered that the Computer Science Department's web server and PHP installation did not support accessing an Oracle database server. Therefore, a rush job was performed on the second implementation consisting of an Oracle database accessed by a Java user interface.

It would have been interesting to spend more time with the Oracle implementation. I have no previous Oracle experience. The desire was to use the MySQL implementation to help guide the development process of the Oracle system. Very few of the “more advanced” features of Oracle were used including domains and views. Time constraints prevented this exploration from occurring. It will still be interesting to look into Oracle more as it is the preferred DBMS (aside from DB2 for the IBM shops) of larger companies.