Introduction/Objectives

System Analysis

Fesibility Study

SRS Document

1. Software Engineering para diagrams Applications
2. Software /HardWare

System

a)DFD

b)ER/UML

Coding

Code Efficiency

Validation Checks

Testing Technics

Testing Scneries

Ourput Screens

Reports

Future Scope

Bibliography

Introduction/Objectives

This project can deal with the automation of the goods transfer of an organization . Every organization can require the raw material to manufacture the goods.

The cargo management system mainly contains a different Persons to control the entire organization. The manufacturer can take the goods from the vendor and they can convert into the finished goods or products. The cargo manager controls the entire organization. The vendor supplies raw materials to our organization. The storage manager can maintained the all information related to the stock maintained in the good downs. The inspection manager can identify the damaged goods and send return to the cargo manager.

The bill payable can give amount of all the goods which are supplied by the vendor .

System Analysis

EXISTING SYSTEM

In the existing system every thing is manual. Occurrence of errors is more while accessing the data. Data maintenance creates a problem. Editing or modifying a record required way is not possible. Creation of entries and reports is problematic. The system is irregular and inefficient due to lack of uniformity.

PROPOSED SYSTEM

To overcome the difficulties of an offline system, which requires lot of human intervention and lot of time and money, cargo maintainces is looking for web application over Internet. The computerized “Online System” has many benefits over the manual system. The time consumption in achievement of tasks in case of computerized system is much less than the manual system. Maintenance of number of files is generally reduced. Cost of retrieving the data is reduced.

Retrieving the data in a desired manner is possible. Manual work is reduced. Retrieval and access of data is easy. Transactions are processed quickly and easily. Information sharing becomes quite easy.

Fesibility Study

**REQURIEMENTS ANALYSIS**

The requirement phase basically consists of three activities:

* Requirement Analysis
* Requirement Specification
* Requirement Validation

**Requirement Analysis:**

Requirement Analysis is a software engineering task that bridges the gap between system level software allocation and software design. It provides the system engineer to specify software function and performance, indicate software’s interface with the other system elements and establish constraints that software must meet.

The basic aim of this stage is to obtain a clear picture of the needs and requirements of the end-user and also the organization. Analysis involves interaction between the clients and the analysis. Usually analysts research a problem by asking questions and reading existing documents. The analysts have to uncover the real needs of the user even if they don’t know them clearly. During analysis it is essential that a complete and consistent set of specifications emerge for the system. Here it is essential to resolve the contradictions that could emerge from information got from various parties. This is essential to ensure that the final specifications are consistent.

It may be divided into 5 areas of effort.

* Problem recognition
* Evaluation and synthesis
* Modeling
* Specification
* Review

Each Requirement analysis method has a unique point of view. However all analysis methods are related by a set of operational principles.

They are

* The information domain of the problem must be represented and understood.
* The functions that the software is to perform must be defined.
* The behavior of the software as a consequence of external events must be defined.
* The models that depict information, function and behavior must be partitioned in a hierarchical or layered fashion.
* The analysis process must move from essential information to Implementation detail

**Requirement Analysis in this Project**

The main aim in this stage is to assess what kind of a system would be suitable for a problem and how to build it. The requirements of this system can be defined by going through the existing system and its problems. They discussing (speak) about the new system to be built and their expectations from it. The steps involved would be

**Evaluation and Synthesis:**

In the proposed system this application saves the lot of time, and it is time saving process when we use this application. Using this application we can easy to manage daily treatments and easy to maintain the historical data. No specific training is required for the employees to use this application. They can easily use the tool that decreases manual hours spending for normal things and hence increases the performance.

**REQUIREMENTS SPECIFICATION**

**Specification Principles**:

Software Requirements Specification plays an important role in creating quality software solutions. Specification is basically a representation process. Requirements are represented in a manner that ultimately leads to successful software implementation.

Requirements may be specified in a variety of ways. However there are some guidelines worth following: -

Representation format and content should be relevant to the problem

Information contained within the specification should be nested

Diagrams and other notational forms should be restricted in number and consistent in use.

Representations should be revisable.

**Software Requirements Specifications**:

The software requirements specification is produced at the culmination of the analysis task. The function and performance allocated to the software as a part of system engineering are refined by establishing a complete information description, a detailed functional and behavioral description, and indication of performance requirements and design constraints, appropriate validation criteria and other data pertinent to requirements.

**FEASIBILITY STUDY**

All projects are feasible given unlimited resources and infinite time. But the development of software is plagued by the scarcity of resources and difficult delivery rates. It is both necessary and prudent to evaluate the feasibility of a project at the earliest possible time.

Three key considerations are involved in the feasibility analysis.

**Economic Feasibility**:

This procedure is to determine the benefits and savings that are expected from a candidate system and compare them with costs. If benefits outweigh costs, then the decision is made to design and implement the system. Otherwise, further justification or alterations in proposed system will have to be made if it is to have a chance of being approved. This is an ongoing effort that improves in accuracy at each phase of the system life cycle.

**Technical Feasibility:**

Technical feasibility centers on the existing computer system (hardware, software, etc.,) and to what extent it can support the proposed addition. If the budget is a serious constraint, then the project is judged not feasible.

**Operational Feasibility**:

People are inherently resistant to change, and computers have been known to facilitate change. It is understandable that the introduction of a candidate system requires special effort to educate, sell, and train the staff on new ways of conducting business.

**FEASIBILITY STUDY IN THIS PROJECT**

**1. Technical feasibility:**

The system is self-explanatory and does not need any extra sophisticated training. As the system has been built by concentrating on the Graphical User Interface Concepts, the application can also be handled very easily with a novice User. The overall time that is required to train the users upon the system is less than half an hour. The System has been added with features of menu-driven and button interaction methods, which makes the user the master as he starts working through the environment. The net time the customer should concentrate is on the installation time.

**2. Financial Feasibility:**

**i) Time Based:** Contrast to the manual system management can generate any report just by single click. In manual system it is too difficult to maintain historical data which become easier in this system. Time consumed to add new records or to view the reports is very less compared to manual system. So this project is feasible in this point of view

**ii)Cost Based:** No special investment need to manage the tool. No specific training is required for employees to use the tool. Investment requires only once at the time of installation. The software used in this project is freeware so the cost of developing the tool is minimal and hence the overall cost.

SRS Document

1. Software Engineering para diagrams Applications

Cargo Management System is cargo to market the goods, to find the requires raw materials buy them. Once a raw materials is purchased. Its calculated in the bills payables settle online payments.

At the time of arrival of raw material, Eletronic weighting scale is used to capture the weights of raw materials in to the computers.

The Cargo Management system consists of 6 modules.

1. Vender
2. Cargo Manager
3. Inspection Manager
4. Storage Manager
5. Bill payable
6. Manufacturer

ROLES AND RESPONSIBILITIES

VENDOR

ROLE: A person who provides the goods to that company.

Responsibilities:

1) he/she have to register

2) he/she have to take ID,Password

3) Receives request from cargo manager

4) send response to cargo manager

CARGO MANAGER

ROLE: Cargo manger is nothing but the person who maintains the company in order to increase their profits.

RESPONSIBILITIES:

1. He/she have to login
2. He/she take the information from the inspection manager about the raw material.
3. He/she have to select the vender
4. Sends the request to the vender about the required rawmaterial.

INSPECTION MANAGER

ROLE: These are middle man between the cargo manager and storage manager.

RESPONSIBILITIES:

1. login in the website
2. receives the raw material from the vender
3. Collect the defect goods and sends back to cargo manager.
4. verify with the store manager about the required raw material.

STORAGE MANAGER

ROLE: He/she take care of storage details.

RESPONSIBILITIES:

1. Login in the website
2. Get the raw material from Inspection manager.
3. Sends request to the inspection manager about the goods
4. Payments bills will have to bill payments.

BILL PAYBLE

ROLE: The person who checks the bills.

RESPONSIBILITIES:

1. Login in the website
2. Clear the bills from storage manager
3. send request to the storage manager about payments.

MANUFACTURER

ROLE: Manufacturer is nothing but the source who manufacture the goods.

RESPONSIBILITIES:

1. Login in the website
2. Request the raw material need for the required stock to the storage manager
3. Send the response to the inspection manager bout the receives raw materials.

Hierarchical representation of data gives the over all view of the project. Each role has the hierarchical representation.

|  |  |  |  |
| --- | --- | --- | --- |
| NO | REQUEST | ELABORATION | FUTHER ELABORATION |
| 1 | Request | Vendor has to enter all his details | ID,PSW proof is compulsory. |
| 2. | Login | Vender has to enter ID and PSW | Server side validations |
| 3. | Gets the request | Gets the request from the cargo manager about raw material |  |
| 4. | Send the response | Send the raw material to the inspection manager |  |

|  |  |  |  |
| --- | --- | --- | --- |
| NO | REQUEST | ELABORATION | FUTHER ELABORATION |
| 1 | Login | Cargo manager has to enter ID and PSW | Server side validations |
| 2. | View | View the request from inspection manager about raw material |  |
| 3. | Send request | Send request from Inspection Manager about the raw material |  |
| 4. | Receive | Receive the confirmation mail from the vendor about raw material |  |

|  |  |  |  |
| --- | --- | --- | --- |
| NO | REQUEST | ELABORATION | FUTHER ELABORATION |
| 1 | Login | Inspection manager has to enter ID and PSW | Server side validations |
| 2. | Gets raw material | Gets the raw material from the vendor |  |
| 3. | Defect raw material | Send information to the cargo manager about broken raw material |  |
| 4. | Get request | Get request from the storage manager about goods |  |
| 5 | Send response | Send response to the storage manager about goods |  |

|  |  |  |  |
| --- | --- | --- | --- |
| NO | REQUEST | ELABORATION | FUTHER ELABORATION |
| 1 | Login | Storage Manager has to enter ID and PSW | Server side validations |
| 2. | Gets goods | Gets the goods from the inspection manager |  |
| 3. | Send request | Send request to the inspection manager about the goods |  |
| 4. | Bill payable | Payment bills to the bill payable |  |

|  |  |  |  |
| --- | --- | --- | --- |
| NO | REQUEST | ELABORATION | FUTHER ELABORATION |
| 1 | Login | Bill payable has to enter ID and PSW | Server side validations |
| 2. | Clear | Clear the bills from the storage manager about goods |  |
| 3. | Send request | Send request to the Storage manager about bills |  |

|  |  |  |  |
| --- | --- | --- | --- |
| NO | REQUEST | ELABORATION | FUTHER ELABORATION |
| 1 | Login | Manufacturer manager has to enter ID and PSW | Server side validations |
| 2. | Send request | Send request to the storage manager about the raw material |  |
| 3. | Receive | Receive goods from the inspection manager |  |
|  |  |  |  |

5W’S1H:

Who, when, what, where, why, how identify for whom the requirements are drawn against “who”. Apply 5W’s1H to capture all the aspects of requirements. This may be repeated for different classes of user identified in the context diagram.

Actor: VENDOR

|  |  |  |
| --- | --- | --- |
| 5W’s1H | User Requirements | Source(who gave the input who compiled them date and time) |
| Who | Supplies the raw material | Vender |
| What | Sends the raw material to the company depends on requirements | Vender |
| When | When the cargo manager sends the request | Vender |
| Where | In the web site | Vender |
| Why | For supply of raw material | Vender |
| How | By using cargo management system application | vender |

Actor: CARGO MANAGER

|  |  |  |
| --- | --- | --- |
| 5W’s1H | User Requirements | Source(who gave the input who compiled them date and time) |
| Who | Maintains the company | Cargo Manager |
| What | Request to vender, decide about the stock required, inspection manager about the goods | Cargo Manager |
| When | When need to raise more capital | Cargo Manager |
| Where | Through online | Cargo Manager |
| Why | For new expansions | Cargo Manager |
| How | By using cargo management system application | Cargo Manager |

Actor: INSPECTION MANAGER

|  |  |  |
| --- | --- | --- |
| 5W’s1H | User Requirements | Source(who gave the input who compiled them date and time) |
| Who | Checks the goods | Inspection Manager |
| What | Receive the stock, check the damage stock, transportation. | Inspection Manager |
| When | It needs to raise the company | Inspection Manager |
| Where | In the store | Inspection Manager |
| Why | For new expansions | Inspection Manager |
| How | By using cargo management system application | Inspection Manager |

Actor: STORAGE MANAGER

|  |  |  |
| --- | --- | --- |
| 5W’s1H | User Requirements | Source(who gave the input who compiled them date and time) |
| Who | Take care of storage details | Storage Manager |
| What | Receives stock from inspection manager request about the required goods. | Storage Manager |
| When | It needs to raise the company | Storage Manager |
| Where | In the store | Storage Manager |
| Why | For new expansions | Storage Manager |
| How | By using cargo management system application | Storage Manager |

Actor: BILL PAYABLE

|  |  |  |
| --- | --- | --- |
| 5W’s1H | User Requirements | Source(who gave the input who compiled them date and time) |
| Who | Who checks the bills | Bill payable |
| What | Clear the bills from storage manager. | Bill payable |
| When | When they should clear the bill | Bill payable |
| Where | In the Website | Bill payable |
| Why | For clearances | Bill payable |
| How | By using cargo management system application | Bill payable |

Actor: MANUFACTURER

|  |  |  |
| --- | --- | --- |
| 5W’s1H | User Requirements | Source(who gave the input who compiled them date and time) |
| Who | Who manufacture the goods | Manufacturer |
| What | Request the raw material | Manufacturer |
| When | When there is requirement | Manufacturer |
| Where | In the manufacturing unit | Manufacturer |
| Why | To provide the required goods | Manufacturer |
| How | By using cargo management system application | Manufacturer |

1. Software /HardWare

Hardware Specifications:

The Hardware on which the project developed has the following configuration:

Application Server Configuration

|  |  |
| --- | --- |
| Content Type | Content Used |
| Computer processor | Pentium III |
| Clock speed | 700MHz Processor |
| Hard Disk | 10GB |
| RAM | 64/128MB |
| Ethernet Card | 32 bit PCI |
| Modem | 56KBPS |

Database Server Configuration

|  |  |
| --- | --- |
| Content Type | Content Used |
| Computer processor | Pentium III |
| Clock speed | 700MHz Processor |
| Hard Disk | 10GB |
| RAM | 64/128MB |
| Ethernet Card | 32 bit PCI |
| Fire wall protected | |

Client System Configuration

|  |  |
| --- | --- |
| Content Type | Content Used |
| Computer processor | Pentium III |
| Clock speed | 650MHz Processor |
| Hard Disk | 10GB |
| RAM | 64/128 MB |
| Modem | 56KBPS |

Software Specifications:

The software’s on which the project developed has the following configuration:

Application Server Configuration

|  |  |
| --- | --- |
| Content Type | Content Used |
| Operating System | Windows 2000 Server |
| Application Server | Web Logic 7.0 |
| Driver | Thin driver |
| Front End | Java |
| Scripts | JavaScript |

Database Server Configuration

|  |  |
| --- | --- |
| Content Type | Content Used |
| Operating System | Win2000 server |
| Back End | Oracle 8I |

System

a)DFD

**CONTEXT DIAGRAM**

Context diagram gives the overall view of the applications. By using combination of the 3 symbols.

CONTEXT DIAGRAM:

ENTITY

PROJECT

DATAFLOW

**DATA FLOW DIAGRAM**

Data flow diagram is a structure analysis tool that is used for graphical representation of data processes through an organization. The data flow approach emphasis on the logic underlying the system, by using combinations of of four symbols.

Data flow diagrams follow top down approach. A full description of a system actually consists of set of DFD’S, which comprises of various levels.

**SQUARE:**

It defines a source or destination of system data.

**ARROW:**

It indicates data flow data in motion. It is a pipelines through which information flows.

**CIRCLE or BUBBLE:**

It represents a process that translates incoming data flows in to outgoing data flows.

**OPEN RECTANGLE:**

**It is a data source at rest a temporary repository of data.**

**DUPLICATE ENTITY:**

**DATA BASE:**

**DUPLICATE DATABASE:**

**PROCESS:**

**FLOW OF DATA:**

b)ER/UML

ER Diagrams

An entity relationship model(ER-Model) is a detailed logical representation of data for an organization or for a business area the ER-Model expresses in terms of entitles and attributes of both the entities and their relationships.

An ER-Model is normally expresses as an entity-relationship diagram which a graphical representation of an ER-Model.

The basis constructs of entity relationship and attributes.

There are different symbols in ER-diagram.

ENTITY:

An entity is a person, place, object event or concept in the user environment about which the organization which to maintain data.

RELATIONSHIP:

A relationship is an association among the instance of one or more entity types.

ATTRIBUTE:

Each entity type has a set of attributes associated with it. An attribute is a property or charactertic of an entity.

COMPOSITE ATTRIBUTE

DATAFLOW:

MANY TO ONE:

ONE TO MANY:

MANY TO MANY:

MANDETARY:

OPTIONAL:

DUPLICATE ENTITY:

VENDOR

order

CARGOM

INSPECM

request

request

STORAM

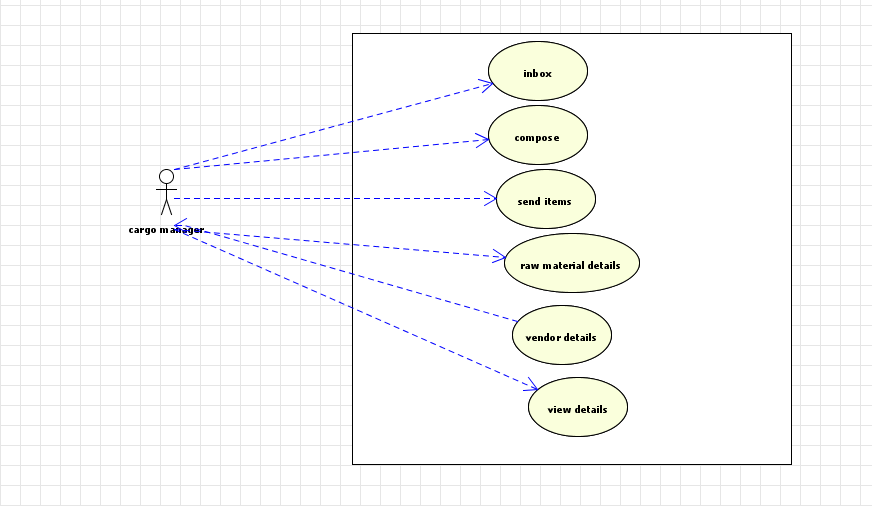
payment

BILLPAY

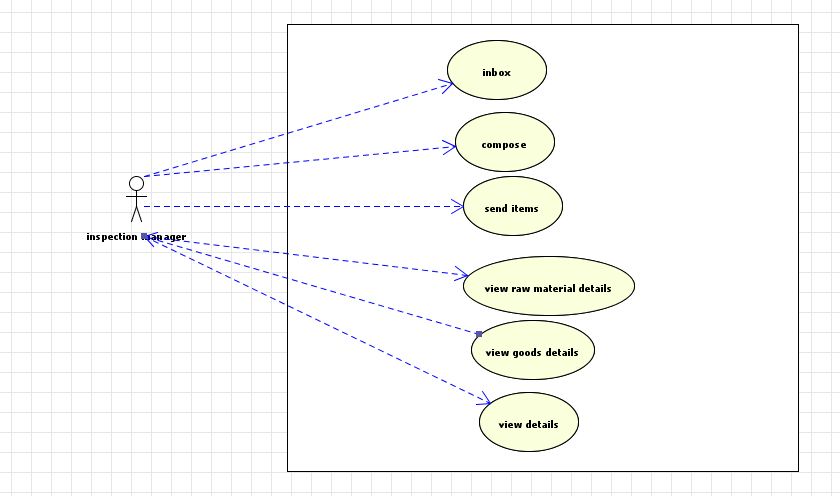
INSPECM

requs

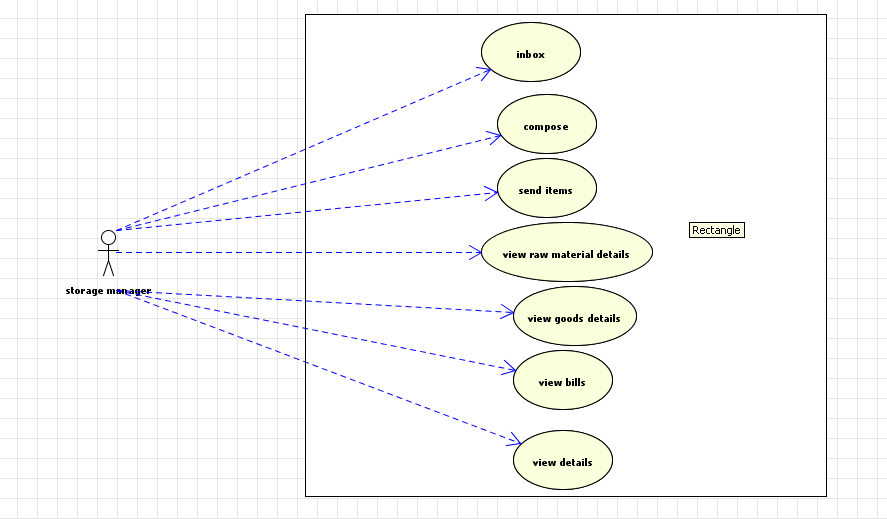
Cargo manager



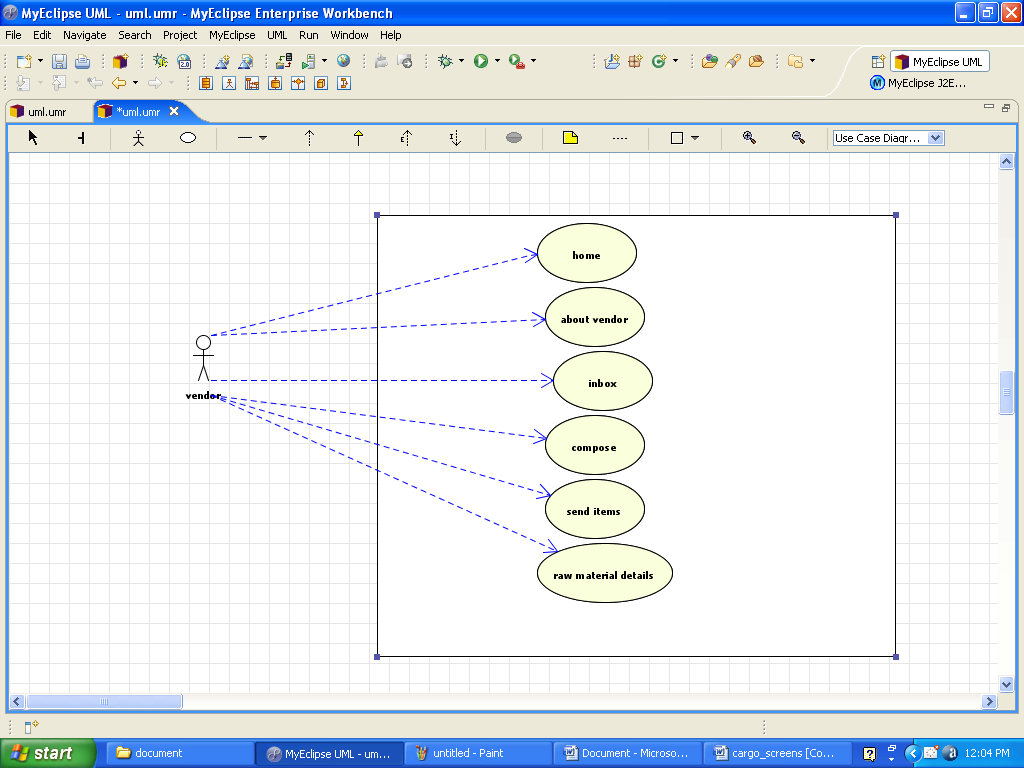
Inspection manager



Inspection manager



Vendor



Coding

DBConnection

package beans;

import java.sql.Connection;

import java.sql.DriverManager;

public class DBConnection {

DBConnection(){}

public static Connection getConnection()throws Exception

{

System.out.println("the connection is obtained from the noraml database connection");

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

Connection conn=DriverManager.getConnection("jdbc:odbc:mydsn","scott","tiger");

return conn;

}

/\*public static Connection getConnection()throws Exception

{

System.out.println("the connection from the datasource");

Context ctxt=new InitialContext();

Context initContext=(Context)ctxt.lookup("java:/comp/env");

DataSource ds=(DataSource)initContext.lookup("jdbc/cargo");

Connection conn=ds.getConnection();

return conn;

}\*/

}

Logged IN Users

package beans;

import java.util.Enumeration;

import javax.servlet.ServletContext;

public class LoggedInUsers {

ServletContext context;

String name=new String();

String username=new String();

boolean found;

//constructor for taking context and username from LoginServlet class

public LoggedInUsers(ServletContext ctxt,String username){

context=ctxt;

this.username=username;

}

//Finds whether user has already logged in

public boolean getExistingUser(){

Enumeration enumeration=context.getAttributeNames();

while(enumeration.hasMoreElements()){

name=enumeration.nextElement().toString();

String name2=context.getAttribute(name).toString();

if(name2.equals(username)){

found=true;

}//if end

}//while end

//context.setAttribute(username,username);

return found;

}

}

Code Efficiency

Code efficiency is a measurement which indicates how the algorithms used in an application perform the desired job in terms of memory and resources usage. The code which consumes less RAM and less processing power can be termed as efficient code. For code efficiency we have to use ideal approaches in solving problems.

In this project Servlets technology is used. By declaring a servlet as a singleton class we can increase the efficiency of code. A class declared as singleton does not allow a new object to be created for each request. In this way it takes less memory in server although number of clients are accessing. So for each client a new process is not created. The jsps developed are also converted to servlets by the web container after deployment.They too can be declared as singleton class. By declaring like this way we make our code efficient.

Validation Checks

The system has been tested and implemented successfully and thus ensured that all the requirements as listed in the software requirements specification are completely fulfilled. In case of erroneous input corresponding error messages are displayed.

Testing Technics

Testing is a process, which reveals errors in the program. It is the major quality measure employed during software development. During testing, the program is executed with a set of conditions known as test cases and the output is evaluated to determine whether the program is performing as expected.

In order to make sure that the system does not have errors, the different levels of testing strategies that are applied at differing phases of software development are:

**1.Unit Testing**

Unit Testing is done on individual modules as they are completed and become executable. It is confined only to the designer's requirements.

**Each module can be tested using the following two strategies:**

**i) Black Box Testing:**

In this strategy some test cases are generated as input conditions that fully execute all functional requirements for the program. This testing has been uses to find errors in the following categories:

a) Incorrect or missing functions

b) Interface errors

c) Errors in data structure or external database access

d) Performance errors

e) Initialization and termination errors.

In this testing only the output is checked for correctness. The logical flow of the data is not checked.

**ii) White Box testing**

In this the test cases are generated on the logic of each module by drawing flow graphs of that module and logical decisions are tested on all the cases.

It has been uses to generate the test cases in the following cases:

1. Guarantee that all independent paths have been executed.
2. Execute all logical decisions on their true and false sides.
3. Execute all loops at their boundaries and within their operational bounds.
4. Execute internal data structures to ensure their validity.

**2. Integrating Testing**

Integration testing ensures that software and subsystems work together as a whole. It tests the interface ofall the modules to make sure that the modules behave properly when integrated together.

**3. System Testing**

Involves in-house testing of the entire system before delivery to the user. Its aim is to satisfy the user the system meets all requirements of the client's specifications.

**4. Acceptance Testing**

It is a pre-delivery testing in which entire system is tested at client's site on real world data to find errors.

##### COMPILING TEST

It was a good idea to do our stress testing early on, because it gave us time to fix some of the unexpected deadlocks and stability problems that only occurred when components were exposed to very high transaction volumes.

##### EXECUTION TEST

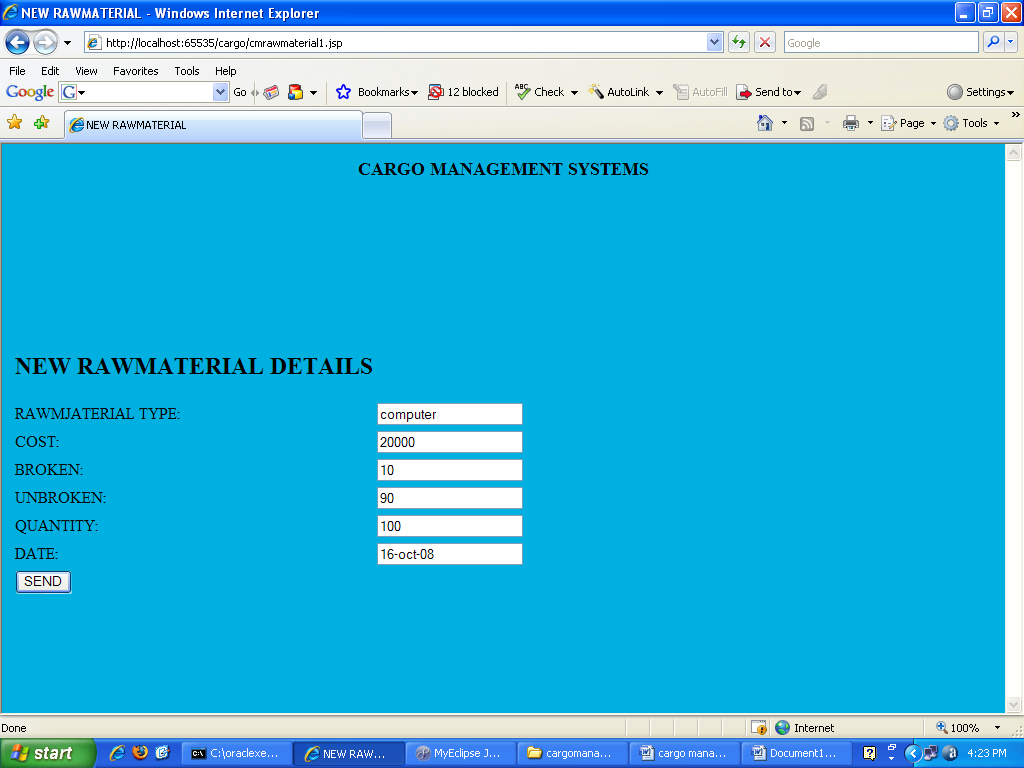
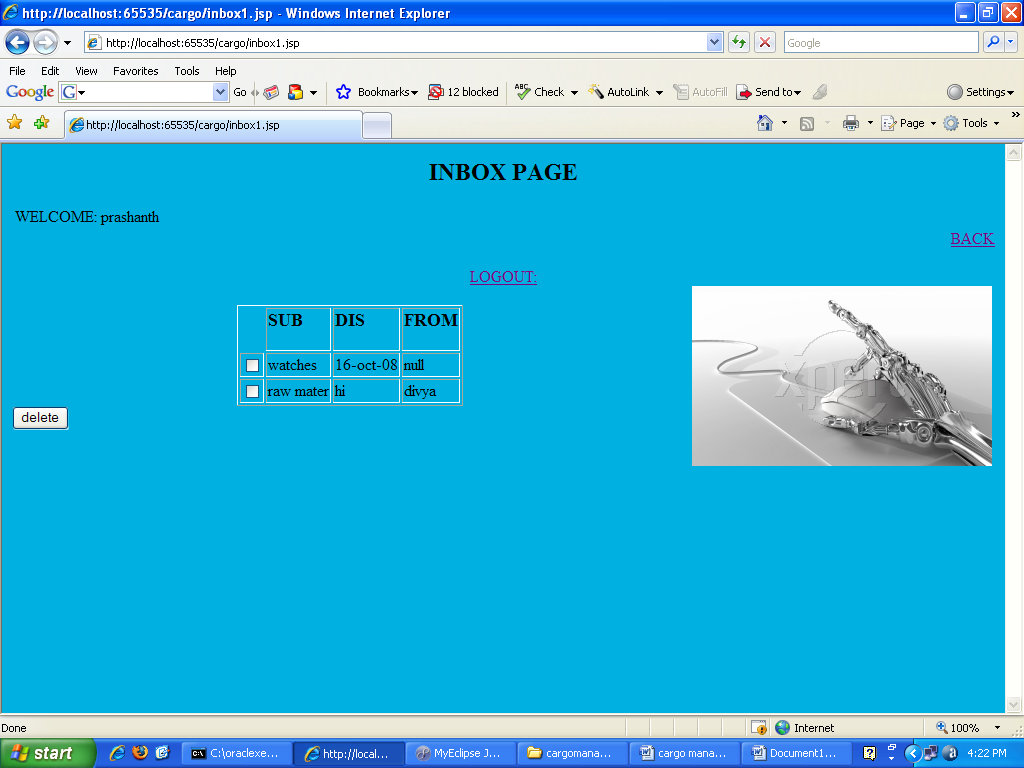
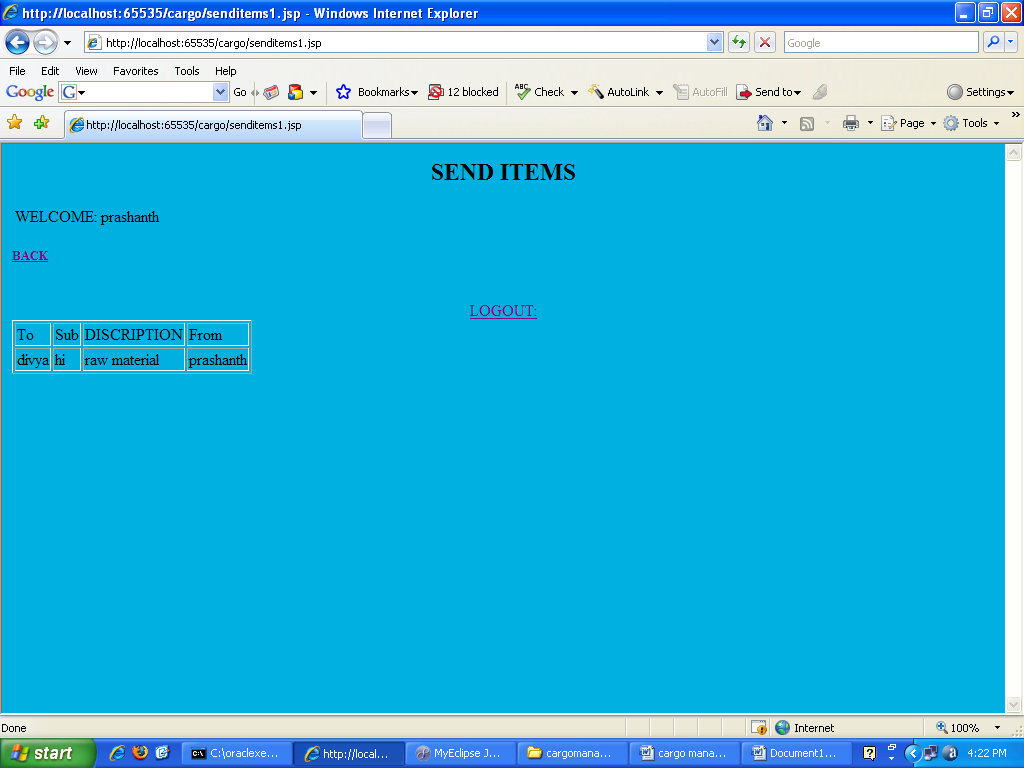
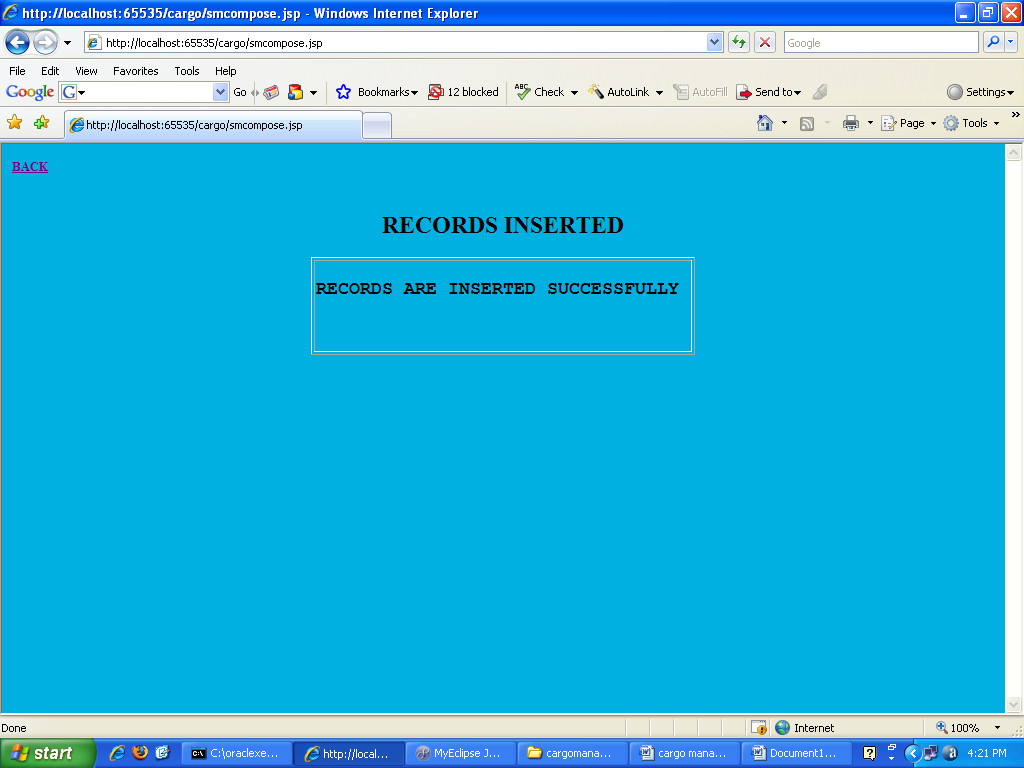
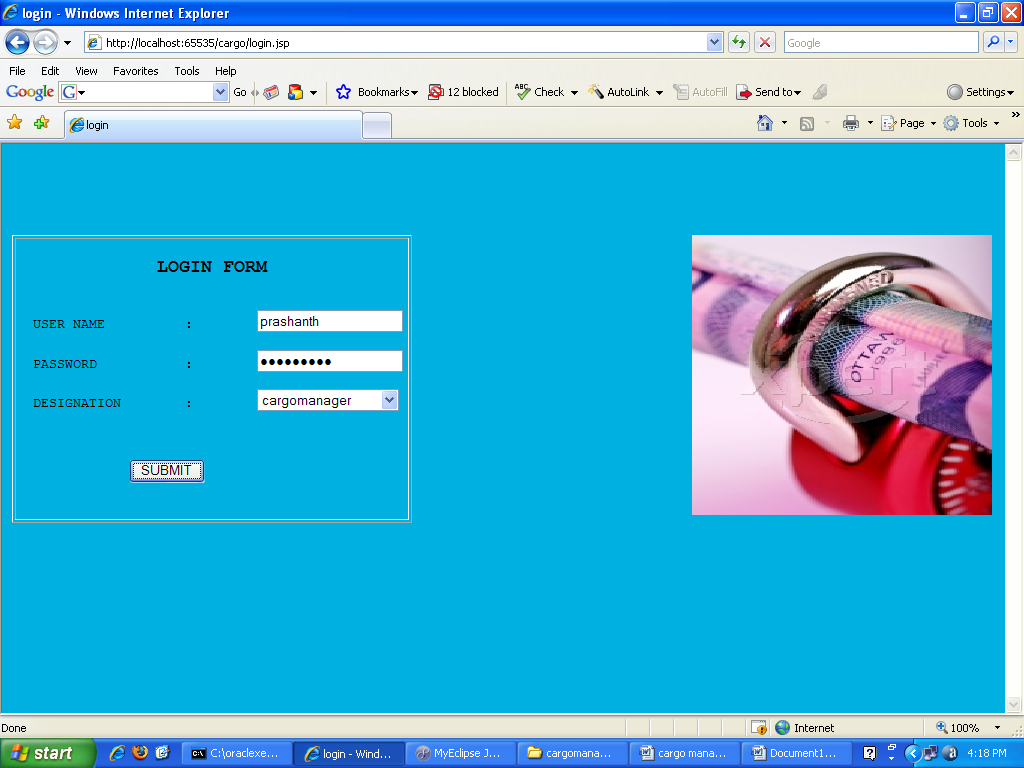
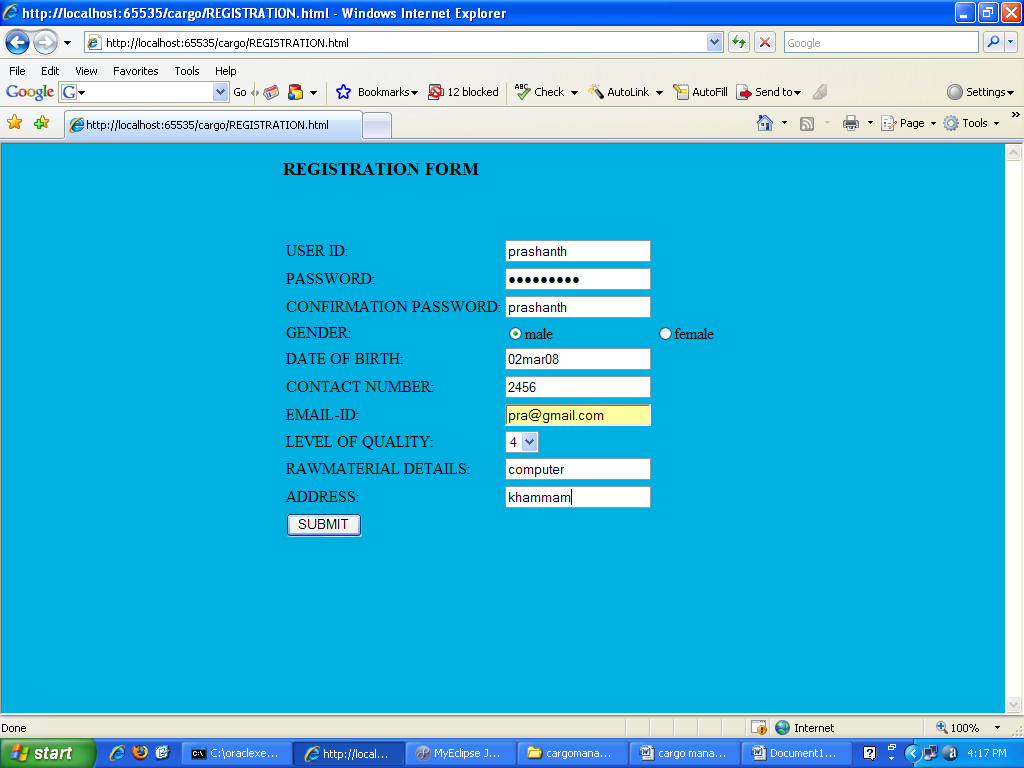
This program was successfully loaded and executed. Because of good programming there was no execution error.

##### OUTPUT TEST

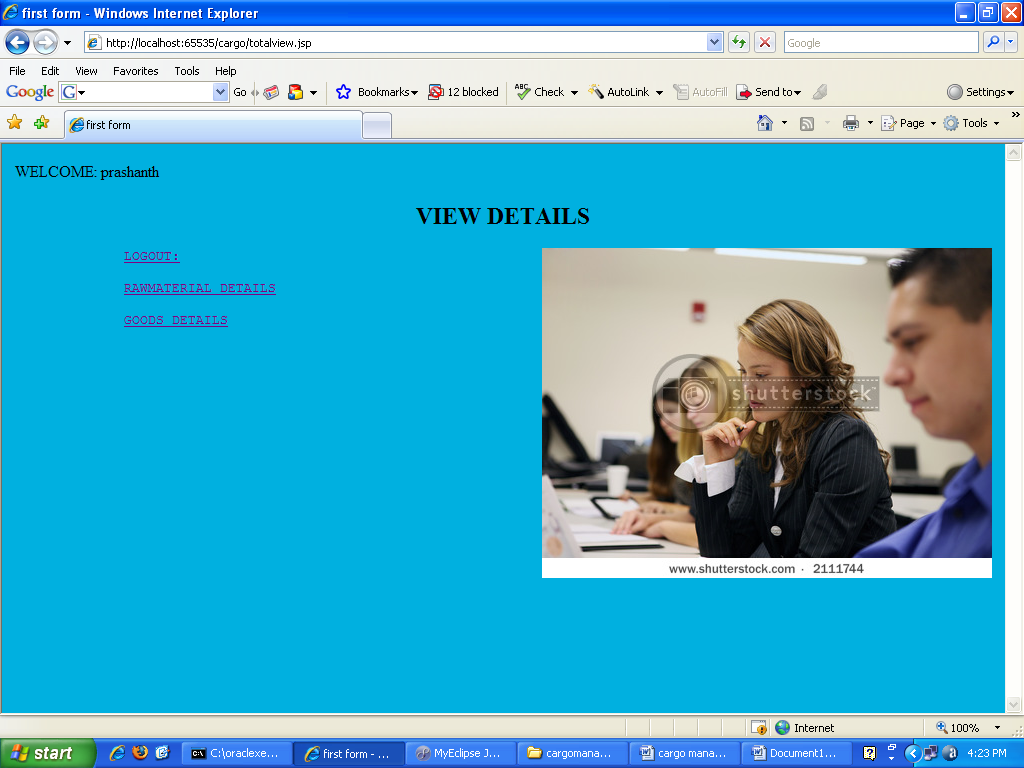
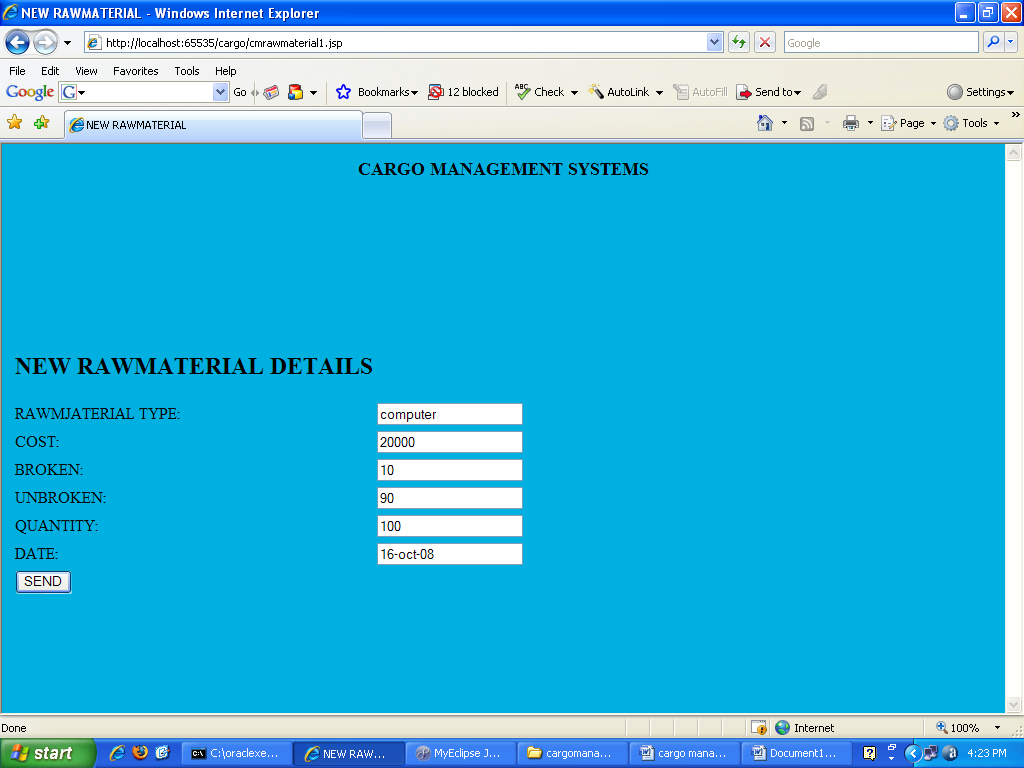
The successful output screens are placed in the output screens section above.

Testing Scneries

Ourput Screens



Reports



Future Scope

*The proposed system scope is limited to Intranet only. In future it can be enhanced to be a global communication medium for multinational companies. We can also implement internationalization (i18n) to support user interface in various/local languages*

Bibliography

* <http://java.sun.com>
* <http://javapassion.com/j2ee>
* <http://java.sun.com/blueprints/corej2eepatterns/Patterns/index.html>
* <http://ibm.com/developerWorks>
* <http://www-136.ibm.com/developerworks/websphere/>