**Software Requirement**

**Specifications**

Prince Labels System

(Version 1.0.0.0)

**Alexander M. Dela Cruz**

Software Developer

Table of Contents

**Introduction**

* 1. User Requirements and Design Specifications

1. **Access and Security**
2. Types of Users1
3. User Levels1
4. Page Access Level2
5. User Name and Password.2
6. User Logging in Different Workstations2
7. Automatic Logoff When Idle2
8. **Functional Specifications**
9. Product Label Generation2
10. QA Approval3
11. Rejecting Labels3
12. Printing of Labels3
13. Voiding and Adding Labels3
14. Reading Labels3
15. Product Takeout3
    1. System Flow and Procedures4
16. **Database Design and Specifications**
    1. Master Files**7**
    2. Transaction Files**7**
    3. System Administration Files8
    4. Audit Files8
    5. Database Management System8
    6. File Structures9
17. **Back Up and Recovery**
18. Backup Maintenance Plan12
19. Manual Backup13
20. Restoring Database14
21. Archiving Database14

**Introduction**

**Overview**

The business of sterilizing and selling products require printing of labels that are affixed to the products to provide proper identification and tracking. Currently, the process of printing and monitoring the application of labels are done manually and Management finds the need to have a better and stringent system of controls to ensure accuracy and eliminate errors associated with a manual system.

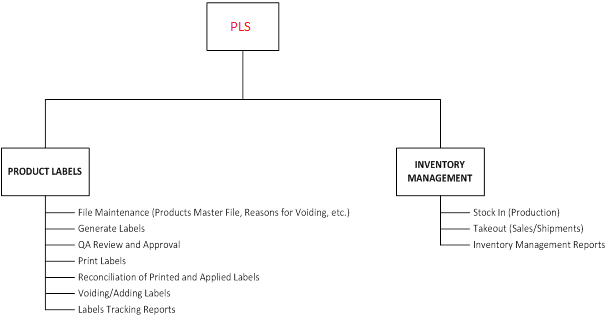
An automated system of printing labels that will ensure the integrity and accuracy of these labels by implementing a strict pre-approval process before printing and conducting reconciliation of printed labels as well as provide inventory management by recording the movement of products being produced and shipped to the customers is highly desired by Management.

With the application of barcode technology, the system shall offer an efficient way of verifying, reconciling or voiding labels by scanning the barcode printed on each label.

The system shall be known as the **Prince Labeling System** or **PLS**.

**Scope of the System**

PLS shall cover two areas of application, namely: 1) Product Labels and 2) Inventory Management. Although PLS is conceptualized to cover product label management only, the movement of products from the storage to their designated destination requires the need to track and monitor the availability of stocks on hand which is covered under inventory management. The following diagram provides an illustration of the scope of the system.



**Figure 1. Scope of PLS**

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Web Application | A program that is designed to be executed through a web browser |
| Web Browser | Commonly known as **browser,** it is a software application for retrieving, presenting and locating information resources from the World Wide Web |
| Intranet | Internal or private network of an organization based on internet technology |
| Menu | A list of commands or options from which a user can choose to execute |
| Menu-driven | Part of a graphical user interface with menu choices that a user can select |
| GUI | Graphical User Interface - way for humans to interact with computers that uses windows, icons and menus and which can be manipulated by a mouse |
| Domain | A group of computers and devices on a network that are administered as a unit with common rules and procedures |
| E-signature | electronic sound, symbol, or process attached to or logically associated with a record and executed or adopted by a person with the intent to sign the record |

**Definitions, Acronyms and Abbreviations**

**Software Requirements**

The following software specifications are the minimum requirements for PLS to run in a networked environment. It is recommended to obtain the latest version of the required software for more advance features and functions that would contribute to better PLS performance.

Web Browser : Google Chrome (preferred), Mozilla Firefox or

Internet Explorer (ver. 11 or higher)

Site Management : Internet Information Services (IIS 7.5)

Database Management : MS-SQL Server 2008 or higher

Operating System : Client – Windows XP Professional or higher

Server – Windows Server 2003 or higher

**Hardware Requirements**

The following hardware specifications are the minimum requirements for PLS to run in a networked environment. Higher hardware specifications is highly recommended for optimum performance and efficiency

**Client**

Processor : Pentium Type – Dual Core or higher

Memory : 2GB or higher

Hard Disk Capacity : 100GB or higher capacity

Monitor Resolution : 1280 x 1024 or higher

Mouse/Keyboard : Standard mouse/keyboard devices

Bar Code Scanner : Standard Bar Code Scanner

**Server**

Processor : Pentium Type – Dual Core or higher

Memory : 8GB or higher

Hard Disk Capacity : 100GB or higher capacity

Monitor Resolution : 1280 x 1024 or higher

Mouse/Keyboard : Standard mouse/keyboard devices

1. **User Requirements Specifications**
   1. **Access and Security**

PLS shall utilize the Windows login credentials where the user must use the same user name and password when logging in to the Windows Operating System. The login process is covered by the same Windows login policy defined by the IT System Administrator, e.g. locking out the user after 3 failed attempts and requires unlocking the user’s account by the IT Administrator, logon hours and password expiration.

The user is also required to login again when there is inactivity for a set period of time and is currently logged in to PLS. Internal control within the PLS system also prevents the user from logging in to other computer when the user is already logged in to another computer.

1. **Types of Users**

The system shall have 2 types of users, namely:

* Prince Domain Users

This type of users refers to all who have access to the Prince Sterilization computer network. This may generally refer to all employees of Prince Sterilization who are granted access to use the Prince Sterilization computer network.

* Guest Users

Any person who will request access to the system mainly for auditing purposes belongs to this group. They will have Read-Only (RO) access to all the modules of the system.

1. **User Levels**

The system shall have four levels of users, as follows:

* Administrator

The Administrator has the highest level of access to the system. Users under this group can perform system administration functions like adding users and defining their level of access, deactivating users and execute tasks on all modules of the system.

* Manager

The Manager level handles specific roles such as maintaining the Products Master File (adding, editing and deactivating products), generating, printing and reconciling labels to ensure that the labels printed are affixed to the products and generating reports.

* Quality Assurance (QA)

The QA group ensures that all labels being printed are accurate and accounted for. They have the sole authority to approve the labels to be printed and in cases where there are labels that need to be voided or rejected for a specific reason, they are the only users who can void or disapprove printing of labels and add new replacement labels.

* User

This level applies to technicians who are authorized to generate, print and reconcile labels.

1. **Page Access Levels**

The system shall provide three levels of page access, namely:

* Full-Access (FA)
  + - * highest level of access which include full file maintenance access – add, edit or delete/deactivate records and other special permissions
* Read-Only (RO)
* no access to file maintenance routines – cannot add, edit or delete/deactivate records but maybe granted special permissions and other tasks specific to this page
  + 1. **User Name and Password**

The user’s Windows login credentials, i.e. user name and password, which are maintained by the Network System Administrator, shall be used to gain access to PLS. When a new user is registered to the network, the System Administrator defines the role and the appropriate level of access for the new user including access to files and folders, allowed time of access to the system, etc. Rules governing user accounts would affect access to the system. If the user fails to login in three consecutive attempts, Windows automatically locks the user account and PLS is terminated. The user must request the System Administrator to unlock his/her accounts form the Windows Domain Controller to be able to log in again to PLS.

* + 1. **User Logging in Different Workstations**

A user shall not be allowed to open PLS in two different workstations. If a user is still logged in a computer different from the computer he/she is currently logging in, the system shall display an alert message like “**You are still logged in PSRDS01.corp.princesterilization!**”.

1. **Automatic Log Off When System is Idle**

PLS shall automatically logoff the current user when there is no activity within a span of 3 minutes to ensure that no intruder will be able to use the system when the current user leaves the system open for a long period of time. The user must log in again to PLS when the automatic logoff is executed.

* 1. **Functional Specifications**
     1. **Product Label Generation**

The system shall provide a process to generate product labels that can be fully verified, tracked and accounted for. Managers and Technicians shall have the access to create labels but will not be able to print them without the prior approval by a member of the Quality Assurance (QA) group.

The system shall cover three label printing categories, namely:

* + Component/Product Kit Labels
  + Insert Box Labels
  + Case Labels

An e-mail shall be sent to the QA group to notify them that labels have been generated from the system and are waiting to be approved for printing by a member of the QA group.

* + 1. **QA Approval**

A member of the QA group shall be responsible in reviewing and verifying the accuracy of the data and approving the printing of the generated labels upon receipt of the notification e-mail. The system shall ensure that the image of the actual label to be printed has been previewed with a “**proof of label**” before the QA can user approve the printing of the generated labels.

Upon approval by the QA user, an e-mail shall be sent to users who have permission to print the labels to proceed with the printing of labels.

* + 1. **Rejecting Labels**

A member of the QA group shall be able to disapprove the printing of generated labels when errors were found during the review and verification of the information to be printed. When labels are not approved for printing, the system shall send an email notification to the technicians that the labels were rejected and will not be printed.

* + 1. **Printing of Labels**

Upon receipt of the e-mail notification from the QA group, the user who is authorized to print labels can then proceed to print the labels from the system using a specialized label printer. The label shall have an embedded 2D barcode that represents a unique identifier for each printed label.

If for some reason, a problem occurred during printing and the printed label is unusable, the label must be surrendered to the QA group for verification and voiding.

* + 1. **Voiding and Adding Labels**

Only a QA user shall have access to void labels. All labels that are rendered to be unusable shall be surrendered to the QA group. When voiding is to be done, the system shall require the QA user to indicate the reason why the label is being voided for tracking purposes.

Voiding labels shall require the use of scanner to scan the embedded barcode in the label for speed and efficiency.

When the number of labels is defined and the labels are generated, the system shall be able to restrict non-QA user to generate additional labels as replacement for voided labels. Only a QA user shall be authorized to add new labels to replace voided labels.

* + 1. **Reading Labels**

The system shall provide a process to ensure that all printed labels are applied to the product packaging and all are accounted for. This process of reconciliation shall require scanning of the embedded barcode on the label when it is affixed to the packaging of the product for speed and efficiency.

* + 1. **Product Takeout**

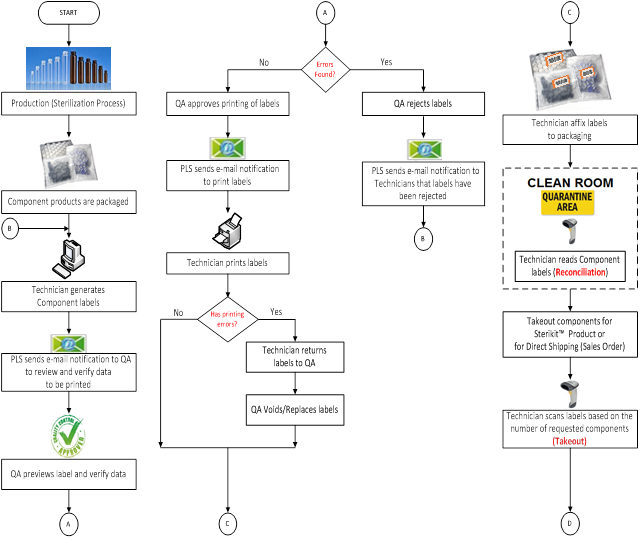
When component products (vials, stoppers and seals) with the labels affixed on their packaging will be taken out from the clean room quarantine area, the system shall provide a scanning process from the embedded barcode on the label to account for the products that will be taken out. This will result in the calculation of the remaining stock on hand (ending inventory) after the products are taken out. It is also important that the system shall ensure that expired products cannot be taken out or shipped out to customers.

This scanning process shall also apply to the warehouse quarantine area for products that are being shipped out to their respective destinations.

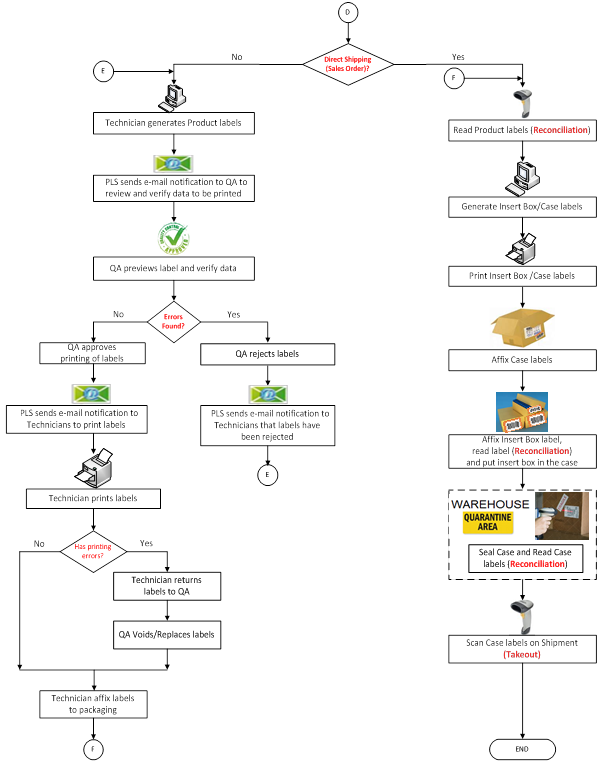
* + 1. **System Flow and Procedures**

The following list describes the processes that must be performed under the PLS. Figures 2 and 3 illustrate the system flow and procedures.

* The production phase involves a process of sterilization for the components (Vials, Stoppers and Seals).
* The materials are packaged into a specialized packaging material.
* Authorized users then generate labels for the component materials. Control ID for the product master label is generated but no labels are printed in this phase. The label numbers are then generated based on the product master label’s Control ID and number of labels to be generated.
* An email is sent to QA to request for review and verification of the information on the labels to be printed and approval for printing if everything is in order.
* QA verifies information on the label using the **Proof of Label** as basis. If everything is in order, QA user e-signs the labels. An email is sent to users who are authorized to print labels notifying them that the labels are approved for printing.
* Otherwise, QA rejects the labels and an email notification is sent to Technician that an error was found and has to be rejected. Technician has to generate another set of labels. The process is repeated while errors are found in the label information.
* Labels are printed in a specialized device for printing labels. If errors occurred during printing, the labels must be returned to QA office and voided. Labels that are voided may be replaced as needed.
* Labels are affixed to the packaging of the component materials.
* Barcode embedded in the packaging is scanned to confirm label application and to register the physical existence (inventory count) of the materials. Component materials with labels are stored in the Clean Room (Quarantine Area).
* A request is made to take out components from the Clean Room for direct shipment (Sales Order) or for assembling a Sterikit™ product.
* Barcode embedded in the packaging of the product to be taken out is scanned to record the takeout of the materials (inventory stock out).
* Sterikit™ product labels will be generated and processed in the same way with that of the component materials – QA review and approval or rejection, voiding, etc.
* Insert box labels are generated at the same time with the Case labels.
* For Insert Box and Case labels, there is no need for QA approval and they can be printed consecutively.
* Labels are affixed to the insert box and the embedded barcode is scanned to confirm application of the label into the insert box or packaging and will be transferred into a case box according to the defined size per product.
* Case labels are affixed to the case box and the embedded barcode is scanned to confirm application of the label and to register physical existence of the product in the Warehouse (Quarantine Area).
* During shipment, Case labels are scanned for takeout from the Warehouse.



**Figure 2. System Flow Diagram – Part 1**

****

**Figure 3. System Flow Diagram – Part 2**

* 1. **Database Design and Specifications**
     1. **Master Files**
* **Products Master File**

The products master file stores information about the product that does not change frequently and are likely to be permanent. This includes the SKU number, storage condition, case size, etc. This is the initial file that needs to be setup before any other processing activities can be started.

This file specifically stores information about the Prince Laboratories Sterikit™ products including its individual components - SteriVials™, SteriStoppers™ and SteriSeals™. The Sterikit™ products represent one category and the individual components as another category.

* **Reasons for Voiding Labels**

To keep track of the reason when labels are voided, this file is populated with the different reasons why a particular label must be voided. These include printing error, damaged product, wrong label requested/printed, etc.

* **Reasons for Rejecting Labels**

To keep track of the reason when labels are rejected, this file is populated with the different reasons why a particular label must be rejected. These include wrong number of labels requested, wrong product description, etc.

* + 1. **Transaction Files**
* **Product Labels**

All generated labels are saved in the product labels file. This is considered as the control file which contains the master data of the labels to be generated and processed which includes the control ID number, number of labels generated, manufacturing and expiration date, lot number and other related information.

* **Label Numbers**

Each individual labels that are generated are stored in this file and contains the detailed information like the label ID number, category code, date of approval and other pertinent information for controls.

* **Case Label Size**

This file provides a way to print actual product size or number of products in a case label when the number of products will indicate a lower number of product sizes per case.

* + 1. **System Administration Files**
* **Users Master File**

PLS shall have an independent Users Master File from the other existing Prince applications. This contains the basic user information like user name, type of user, level of access, etc.

* **System Pages**

All the available pages to be accessed by the users are to be stored in this file. It shall contain the Page ID, name and description of the page.

* **Users Page Access**

This file stores all the pages that a particular user can access in the PLS including their level of access. The system administrator shall assign the access of users to the system pages.

* **Users Log**

This log file records a login by a user and the current log status for controlling the access of a particular user to the system and to the entire network.

* **User Groups**

The file will provide a grouping of users as to their company assigned departments. This can include the departments who shall use the system like Calibration and Quality Assurance (QA) groups.

* + 1. **Audit Files**

The audit files shall serve as a source for tracking and identifying changes made to a file and as a reference for restoring records to its previous state and printing audit trail. This shall be available only to QA users who have the main role of monitoring any changes made to a particular file.

* 1. **Database Management System**

The **SQL Server Management Studio** of **SQL Server 2008** shall be used to manage and maintain the PLS database with the following properties:

* + - * Supports the relational database model - inner joins, left outer joins, right outer joins, full outer joins, and multiple joins within a query with multi-user access.
      * Use of indexes for faster sorts and queries based on different parameters and preventing duplicate data.
      * Use of stored codes (stored procedures and triggers) for faster execution of queries.
  1. **File Structures**

Database Name: **PLS**

File Name: **Products**

File Structure:

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Field Name** | **Data Type** | **Allow Nulls** | **Notes** |
| ProductID | int | No | Primary Key, no duplication |
| ProductDesc | nvarchar(150) | Yes | Description of the product |
| SKUNo | nvarchar(50) | Yes | SKU number assigned to a product |
| StorageCondition | nvarchar(50) | Yes | Range of temperature to store the product |
| CategoryCode | tinyint | Yes | Product Category:  1 - Component  2 - Product |
| CaseSize | tinyint | Yes | Number of products assigned for a particular case box |
| DateCreated | datetime | Yes | Date and time the record is created |
| CreatedByID | smallint | Yes | ID of user who created the record |
| LastUpdate | datetime | Yes | Date and time of last update |
| LastUserID | smallint | Yes | ID of user who made the last update |

File Name: **ProductLabels**

File Structure:

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Field Name** | **Data Type** | **Allow Nulls** | **Notes** |
| ControlID | int | No | Primary Key, no duplication |
| ManufacturingDate | smalldatetime | Yes | Date the product is processed |
| ExpirationDate | smalldatetime | Yes | Expiration date calculated 2 years after manufacturing date |
| ProductID | int | Yes | Product ID – linked to Products |
| LotNo | nvarchar(50) | Yes | Lot number assigned to the batch |
| PONo | nvarchar(50) | Yes | Purchase order number |
| LabelCount | int | Yes | Number of labels to be generated |
| Status | bit | Yes | Status of label printing:  1 – Completed  2 – Pending |
| DateCreated | datetime | Yes | Date and time the record is created |
| CreatedByID | smallint | Yes | ID of user who created the record |
| LastUpdate | datetime | Yes | Date and time of last update |
| LastUserID | smallint | Yes | ID of user who made the last update |

File Name: **LabelNumbers**

File Structure:

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Field Name** | **Data Type** | **Allow Nulls** | **Notes** |
| ControID | int | No | } Primary Keys, |
| LabelNo | int | No | } no duplication |
| CategoryCode | tinyint | Yes | Label category:  1 – Product  2 – Insert  3 – Case  4 – Component |
| DateApproved | datetime | Yes | Date and time the label is approved for printing |
| ApprovedByID | smallint | Yes | ID of QA user who approved the label for printing |
| Printed | bit | Yes | Identifier for printed labels |
| PrintedByID | smallint | Yes | ID of user who printed the label |
| DatePrinted | datetime | Yes | Date and time the label is printed |
| LabelStatus | tinyint | Yes | Status of Label:  1 – Original  2 – Added  3 – Voided |
| StatusReason | int | Yes | ID representing the reason for voiding, linked to Voiding Reasons table |
| Voided | bit | Yes | Identifier for voided labels |
| VoidedByID | smallint | Yes | ID of user who voided the label |
| DateVoided | datetime | Yes | Date and time the label is voided |
| DateApplied | datetime | Yes | Date and time the label is reconciled |
| Rejected | bit | Yes | Identifier for rejected labels |
| DateRejected | datetime | Yes | Date and time the label is rejected |
| RejectedByID | smallint | Yes | ID of user who rejected the label |
| AppliedByID | smallint | Yes | ID of user who reconciled the label |
| Previewed | smallint | Yes | Identifier if label is previewed  1 – Previewed  0 or null – Not Yet Previewed |
| DateIssued | datetime | Yes | Date and time a product is taken out from warehouse |
| IssuedByID | smallint | Yes | ID of user who took out the product |
| TakeOutCode | tinyint | Yes | 1 – Taken out  0 or Null – not Taken out |
| DateCreated | datetime | Yes | Date and time the record is created |
| CreatedByID | smallint | Yes | ID of user who created the record |
| LastUpdate | datetime | Yes | Date and time of last update |
| LastUserID | smallint | Yes | ID of user who made the last update |

File Name: **VoidReasons**

File Structure:

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Field Name** | **Data Type** | **Allow Nulls** | **Notes** |
| ReasonID | smallint | No | Primary Key – no duplicate |
| ReasonDesc | nvarchar(50) | Yes | Text of reason for voiding |
| DateCreated | datetime | Yes | Date and time the record is created |
| CreatedByID | smallint | Yes | ID of user who created the record |
| LastUpdate | datetime | Yes | Date and time of last update |
| LastUserID | smallint | Yes | ID of user who made the last update |

File Name: **RejectReasons**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data Field Name** | **Data Type** | **Allow Nulls** | **Notes** |
| ReasonID | smallint | No | Primary Key – no duplicate |
| ReasonDesc | nvarchar(50) | Yes | Text of reason for rejection |
| DateCreated | datetime | Yes | Date and time the record is created |
| CreatedByID | smallint | Yes | ID of user who created the record |
| LastUpdate | datetime | Yes | Date and time of last update |
| LastUserID | smallint | Yes | ID of user who made the last update |

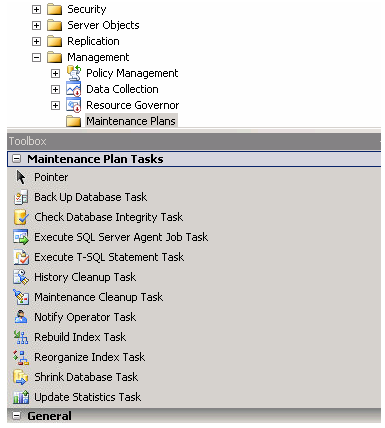
File Structure:

1. **Backup and Recovery**

The backup and recovery procedures for PLS shall be handled within the **SQL Server Database Management System (DBMS**). IT personnel with the role of a **Database Administrator** (**DBA**) can create automated tasks for backing up the database and also perform manual back up and restoration tasks.

1. **Scheduled Backup Maintenance Plan**

Using the features of the SQL Server DBMS, a scheduled backup and other maintenance plan for the PLS database shall be created to endure

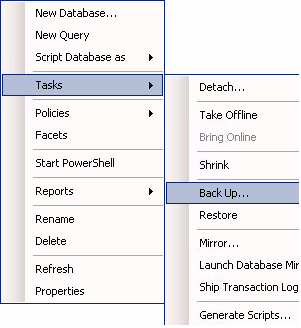


**Figure 4. Back Up Database Maintenance Plan**

1. **Manual Backup Procedure**

A manual back up procedure can also be performed as an option to the automated backup under the maintenance schedule. This can be done in cases where urgent changes to any of the table structure must be undertaken for any reason.

This is to ensure that the latest, working copy of the database can be restored in case of failure in the procedure and the database gets corrupted.

****

**Figure 5. Manual Backup**

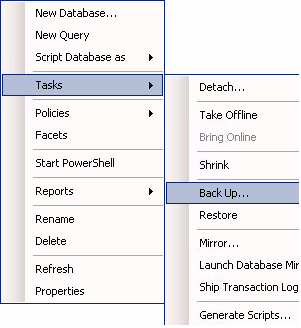
1. **Restoring a Backup Copy**

If, for some reason, the database gets corrupted, a backup copy of the database can be restored within the SQL Server DBMS by executing the **Restore** function. The integrity of the backup copy must be initially confirmed by restoring a copy to a test unit before deploying to the Production Server.

1. **Archiving**

The archiving plan for the PLS database depends on the required retention period set by Management. A separate database prefixed with “Arc”, i.e. Arc\_PLS database, shall store the old records for future retrieval.

The archive data may contain training records below the retention period and would be moved from the original database to the archive database through a code executed as needed.



**Figure 6. Restore Database**



PLS

Arc\_PLS

**Figure 7. Archiving PLS Records**