

**Version 1.1**

**Date: 2018/12/12**

Customers, especially on-line customers are specifically allocated to one or more sales staff members who need to act in the roll of customer support seeing that customers place their orders on-line

Sales staff to customer allocations

**Table of Contents**

Document approval and distribution list 2

1. Introduction 3

2. Audience 3

3. Objectives 3

4. Business Flow 3

5. Detailed description of functionality 3

6. Dependencies 4

7. Application design philosophy 4

8. Database design philosophy 4

9. Catalogue lookup to sales-order 4

10. Database entities and relationships 6

11. Programs 8

11.1 MS Windows Executables 8

11.2 SQL Stored Procedures 8

12. Acceptance 10

# Document approval and distribution list

|  |  |  |  |
| --- | --- | --- | --- |
| **Description** | **Name / Title** | **Signature** | **Date** |
| **Document Type / purpose** | | | |
| Prepared by |  |  |  |
| Reviewed by |  |  |  |
| Approved by |  |  |  |

# Introduction

The on-line customer base attributes for a large portion of the effective turnover which is non-attributable to sales staff.

Clinically, sales staff should not be earing commission from on-line sales. However, in this conflicting situation, it is known that sales staff would collude with on-line customer to rather pass their orders to the sales team rather than self-helping.

The debate is then to not permit ANY sales commission for designated customers. However, sales staff do not promote worthy customers to on-line status consequently.

The middle of the road solution to this was to apportion on-line customers to sales staff where sales commission was paid based on on-line customer performance.

The solution provides for a percentage-based apportionment of turnover between one or more allocated sales staff to assist in resolving staff conflicts re commission earnings.

The expectation from Engineparts management is for the sales staff to act in the capacity of a sales representative to engender loyalty and improve turnover and profit. Also to ensure proper use of the ePart decision support.

# Audience

* Sales
* Sales management

# Objectives

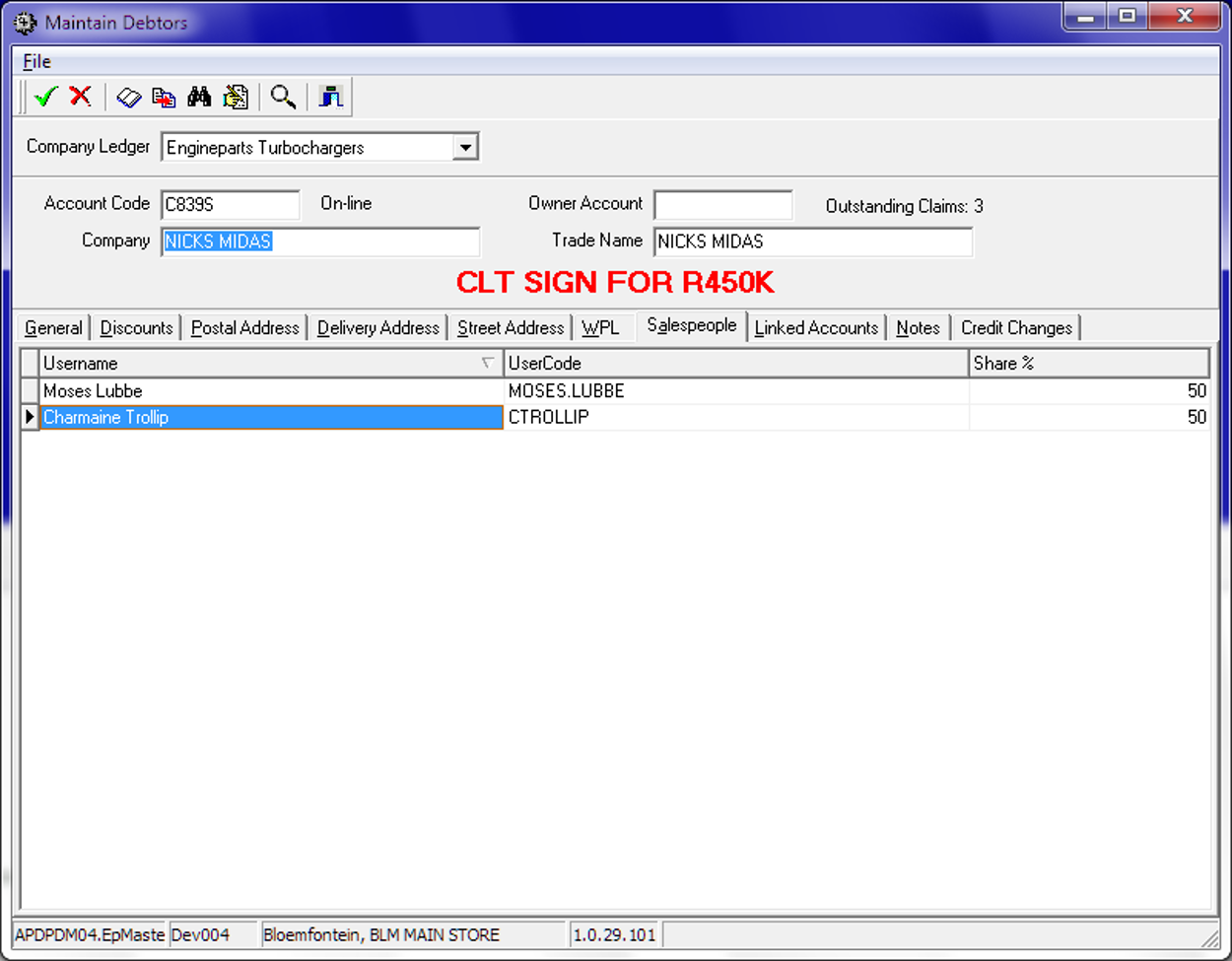
Create and maintain sales staff to customer allocation relationships including percentage share where more than 1 sales person is allocated to a specific customer.

Provide management reporting reflecting the apportioned turnover per customer and allocated sales staff.

Although not part of the solution; sales staff and on-line customers need to be motivated to use the ePart system to its most effective deployment

# Detailed description of functionality

The following illustration provides a view of the Debtors Maintenance



# Dependencies

* Sales staff management
* Sales order to Invoice
* RFC (return for Credit)
* Debtors account management (sales staff allocation & % allocation)
* On-line customer database.

# Application design philosophy

The same design philosophy is used as in most other ePART applications. It consists of 3 basic components:

* 1. Presentation – A C++Builder application with limited, if any, business logic.
  2. Business logic – Implemented as stored procedures on an MSSQL database server.
  3. Data persistence – The MSSQL database server is responsible for persistence.

This modular approach should increase the longevity of the product by allowing developers to replace the business logic and presentation independently of one another.

# Database design philosophy

The Debtors Maintenance application uses an object-oriented approach to the data and a three-tier approach for presentation, business logic and database persistency.

The sales staff assignment to a debtor requires that an active debtor entry be recalled. The assignment function is a tab on the ***action bar – Sales People***

**<This section will be replaced with a link to the database design philosophy document>**

# image.pngDatabase entities and relationships

# Programs

Notably, is the naming convention where the 1st 2 / 3 letters denote the sub-system i.e. ‘dr’ for debtors, ‘stk’ for stock management etc.

The stored procedures used here represent the ***business logic and data persistence*** of the solution whereas the ***presentation logic*** only performs visual functions.

The presentation logic used the Borland C++ Builder development framework; originally used for its use in tertiary education centres i.e. UOFS. However, the framework was swapped in favour of MS Visual Studio with C# and subsequently Borland as a company has ceased business, making the C++ Builder framework obsolete

***Importantly, the Borland C++ framework MUST be replaced over the long term with potentially re-developing the user interfaces in MS C# in and amongst many other UI opportunities such as web-based development.***

The effort to replace the presentation logic is near enough about 10% of the total code base; consequently represents a smaller percentage of effort.

# MS Windows Executables

|  |  |
| --- | --- |
| **Name** | **Description** |
| DrMaintainDebtors.exe | Compile name to include into EpMenu |

# SQL Stored Procedures

|  |  |
| --- | --- |
| **Name** | **Description** |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# Acceptance

I hereby confirm that I have been fully informed of the documents content and, received training to understand how the detailed instructions are to be applied:

Name …………………………………………………………………………….

Job Title ………………………………………………………………………….

Signed ……………………………………………………………………………

Date ………………………………………………………………………………