# Database Design Requirements

## Client: Carlotta Antique Jewellery

Document Prepared By: Stefan Mazaleigue

## Requirements brief

Currently the sales information is listed in a spreadsheet providing a full name, address, sale date, item sold, item type, price of item, year item was made, total sale value, newsletter request flag and an age category of the customer. There is up to three item sales per line. An increase to volume of input makes these manual entries and searches very inefficient.

The current system is restrictive to access and updating of information due to searches, updates and new entries all being manually input. This leads to inevitable errors such as data input into wrong columns, miscalculating totals an empty fields that should contain data. A transfer of this information from a spreadsheet to a database is required in order to improve efficiency. There is too much information for one single table that creates redundant information (unnecessary repetition of data). The information needs to be split among multiple tables of more specific details. This will be done in the database allowing for reports the client has requested: a list of Sales sorted by item type, a list of sales sorted by customer category, a list of sales by month and a list of customer wanting the newsletter to be sent to them.

The system used will be Microsoft SQL Server. The security login will allow only particular people access to certain information in the database depending on the account type of the user.

## Entity relationship diagram



## Normalisation issues

1. There are multiple multiple customers with the same name requiring a customer ID to distinguish each customer uniquely
2. Name column combines first name and last name that need to be separated

1. Category data has multiple instances of the same name creating redundancy. This needs it’s own table to store the category names to also ensure data integrity (No loss of information)

1. Item Type data has multiple types in the one column. Having this in a separate table removes this redundancy by the item itself determining the item type (by relation)

1. Item fields appear multiple times indicate their own table is necessary to maintain data integrity

1. Amount column is redundant as it can be deciphered from item costs

1. Address should be split up into smaller details in their own columns

## Database diagram

## 

## Approval

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_/\_\_\_/\_\_\_\_\_\_