**Report**

**Introduction**

**Objective**

The purpose pf this document is to explain and justify the decisions made during the construction of the UML diagram for the banking application. This includes the types of relationships, classes and attributes and methods the classes have.

**Scope**

The model will be based on 2 base classes for customers/users and accounts, with an extra class for storing transactions. The report will describe the decisions and justify them.

**Class Design and Inheritance**

For this model I use two base classes: Account and Customer.

**Account**

Account class is an abstract class that will have the attributes of accountID and balance as these are common attributes in every class. As for the methods, as every account should be able to deposit or withdraw money, these two will be in the base account class.

There are three classes that will inherit from the Account class.

**Everyday account**

Everyday account is a simple account that has no overdraft limit, no transaction failure fee, or interest. This account inherits everything it needs from the base class.

**Investment account**

Investment account is the second type of account that has no overdraft limit, but failing to complete a transaction incurs a fixed amount of fee. Lastly this account adds interest to balance over time.

**Omni account**

Omni account is the last account that has the most complexity overall. It has an overdraft limit, and interest rates only apply as long as there is more than $1000 in the account. Failed transactions incur a fee.

**Customer**

Customer class holds customerNumber, name, contactDetails and staffDiscount as attributes. The staffDiscount is initially zero. It has two child classes:

**Regular customer**

Regular customer inherits everything from the base class

**Bank Staff**

Bank staff receive a 50% discount instead of the 0 default.