**Project Name: Automated Teller Machine (ATM) System**

**SUBMITTED TO**

**Dr. Colmenares**

**Submitted by**

Samuel Weems

Richard Oliver

Scott Gordon

**Problem Statement**

A Real-Time ATM (Automatic Transaction Machine) system is needed to provide users with basic ATM services, such as deposit and withdraw funds, transfer funds and check balance. The system should be designed using Object Oriented concepts to be structured into ATM client and server components as well as a synchronous connector that facilitates communication between the client and server. The system should be designed with the below qualities:

* Object Oriented Implementation and Design
* Extensible
* Maintainable
* Modular
* Usable (easy to us and intuitive)
* A Console Driven Application
* Written in the Java programming language
* As Specified in the requirements Document

Below are the features the ATM system components should have:

**ATM Client**

* Takes user input specific to deposit and withdrawing funds, transfer funds and check balance.
* Request services from the ATM server (through the synchronous connector).
* Retrieves server’s response from the synchronous connector.
* Display result of services to the user.

**Synchronous Connector**

* Stores ATM client’s request.
* Stores ATM server’s response.

**ATM Server**

* Reads service request of ATM client from synchronous connector.
* Processes user’s requests.
* Sends the client response to the synchronous connector.

**ATM System Rules**

* User uses their card id and pin number to authenticate and get access to their account. If the card ID and/or pin number does not match an error is generated, and user is not granted access.
* A user can withdraw of maximum of $300 daily from their account. An error is generated is a user tries to withdraw more than the daily limit.
* Whenever money is withdrawn, the balance is updated against the amount withdrawn.
* A user cannot withdraw more than the amount they have in their account. If the amount withdrawn is greater than the balance an error is generated.
* Users can transfer money from one account to another.
* Users can check balance on their account.

**Proposed Solution**

* Figure 1 shows a high level architecture of the ATM system, which comprises an ATM client, a synchronous connector and an ATM server.
* The ATM client and server as active objects, whilst the synchronous connector is passive.
* Synchronous connector is used for communication between active components.
* The features of the ATM client are: deposit and withdraw funds, transfer funds, check balance, take user input, request service from ATM server through synchronous connector, retrieves server response from the synchronous connector and display result of services to user.
* The features of the synchronous connector are: stores the request of the ATM client and stores the ATM server response.
* The features of the ATM server are: retrieve the ATM client’s request from the synchronous connector, process the request and send the client response to the synchronous connector.

Figure 1. High level architecture

ATM Client

Synchronous Connector

ATM Server

**Final Comments**

In this project we combined knowledge from Real-Time systems, Java programming, and Object Oriented design and implementation. At the beginning the group determined the requirements document was unfeasible for the amount of time allotted. After intense negotiation with the customer, we were able to reduce the requirements to a more realistic and feasible size for the project. A large learning curve was associated with learning Java, however in hindsight this was negligible. The largest hurdle with the project was unclear requirements. After several team meetings involved with design, we established a working concept of the system and then implementation was straightforward. Most of the time spent in this project was in the requirements and design phases.