

- ATM
  - ATM - DO
  - ATM GUI - DO
  - ATMUser - DO
- Teller
  - Teller - HL
    - makeConnection(): connected = true
    - accessCustomer(): connected = true
    - deposit(money): balance = balance + deposit
    - withdraw(money): balance = balance - deposit
    - add\_account(customer): success = true, no error
    - delete\_account(customer): success = true, no error
    - transfer(money): balance = balance + transfer, oldBalance = oldBalance - transfer
    - logOut(): socket.close()
  - Supervisor - HL
    - newEmployee(employee): list.length() + 1
    - removeEmployee(employee): list.length() - 1
    - accessEmployee(id): employee id = id
    - processDividend(account): (3000, 0.05, 36) = 3307.5
    - load(): message.authentication = success
  - Employee - DO
- Server
  - Money - AG
    - sub(): 0.00 - 0.01 equals -0.01
    - sub(): -1.01 - -2.02 = 0.92
    - mult(): (100000 \* 1.016) 3 times = 104877.21
      - getFraction(): = 0.96
    - add(): 34.16 + -20.17 = 13.99
    - new Money(): = 0.00
    - toString(): cents always 2 digits
    - new Money(-23, 16, false): -23.16
      - getIsPositive(): = false
  - Message - AG
    - Two new Messages will each have a different id
    - message.packet = new ATMPacket(), no error
    - message.packet = new CustomerPacket(), no error
    - message.id = 234, error (final type)
    - message.success = true, error (final type)
  - Customer - AG
    - new Customer(234, "Aaron", "sesame", 2, 1, new Date(), null):
      - customer.getPasscode() = "sesame"
      - customer.getName() = "Aaron"
      - customer.getNumSavings() = 2

- customer.getClosed() = null
  - customer.getAccounts(), for loop count, num accounts = 3
  - customer.addAccount(43, AccountType.CHECKING)
    - customer.findAccount(43) != null
- Account - AG
  - setBalance( -23.14), positiveStatus() = false
  -
- Server - AG
  - stop() = server stops
  - test = reserveSessionID()
    - sessionIDs.contains(test) = true
    - reserveSessionID(), 1000 times
      - none are the same
    - ClientHandler
      - validateATM(testGoodMessage) => Message.success = true
      - dismiss(testMessageID) => out message has same id
      - balance(testBalance) => out message has indicated balance
    - ConnectionListener
      - login 1000 clients => success, all connected
- CustomerFile - DO
- EmployeeFile - HL
  - employeeID, returns employee
  - totalEmployee(): (5,5)
  - employeePW, (employeeID)
- CheckValidation - AG
  - validateCheck() x 10000, average = 97.5±.1
- DataBase - AG
  - getCustomerFromCard(card) = customer
    - customer.hasAttachedCard():
    - customer.attachedCard() = card
  - setCustomer(customer)
    - findCustomer(customer.getID()) = customer
  - removeCustomer(customer)
    - findCustomer(customer.getID()) = null
- UniqueIDs - AG
  - addID(4) = true
    - addID(4) = false
    - removeID(4) = true