

HIT3119/8119 Enterprise Java
Assignment 1 Final version
Mark Sheet

Group ID: _____

Marker: _____

| | Student ID | Student Name |
|---|------------|--------------|
| 1 | | |
| 2 | | |
| 3 | | |
| 4 | | |

Points to note for students (Extracted from the Assignment doc.):

1. A design document (In this document, you need to identify the roles and responsibilities of each component in your system, remember to include both 3-Tier and 4-Tier architectures in your system and put the right “module” in the right place. It is strongly recommended that you use diagram to illustrate your design.)
2. A simple instruction on running the functions from the application clients (This is a set of simple steps for people to execute your software. Simple descriptions like those in the lab exercises will be sufficient. There is no need to capture screen dumps.)
3. The source code

[As a reminder, please submit both hard and soft copies of Items 1 and 2 above; for Item 3, just submit the soft copy.]

| Software | | | |
|--|--|-------|----|
| Functionality | Items | Marks | |
| Login | [Internet Banking, IB] | | |
| | Login as a Customer | /1 | |
| | Check using invalid customer credential | /0.25 | |
| | Check using bank teller’s staff credential | /0.25 | |
| | [ATM] | | |
| | Login as a Customer | /1 | |
| | Check using invalid customer credential | /0.25 | |
| | Check using bank teller’s staff credential | /0.25 | |
| | [Teller Machine, TM] | | |
| | Login as a Bank Teller using staff credentials | /1 | |
| Create a bank account [TM Only] | Check using invalid staff credentials | /0.25 | |
| | Check using valid customer credentials (login as customer!) | /0.5 | |
| | Check using invalid customer credentials | /0.25 | /5 |
| | [IB and ATM] - Should not have this feature | /1 | |
| | [TM] | | |
| Remove a bank account [TM Only] | Create a new account for an existing Customer | /1 | |
| | Create a new account for a new Customer; and after creation, the customer will be able to login | /2 | |
| | Check using an existing bank account | /1 | /5 |
| | [IB and ATM] - Should not have this feature | /1 | |
| | [TM] | | |
| View Account Balance | Remove an existing bank account | /1 | |
| | Remove an existing bank account whose owner has one account in the bank; and after the removal, the owner cannot login as a customer anymore | /2 | |
| | Check using a non-existing bank account | /1 | /5 |
| | [IB and ATM] | | |
| | Customers view their own account balance | /2 | |
| | Check using other people’s account | /1 | |

HIT3119/8119 Enterprise Java
Assignment 1 Final version
Mark Sheet

| Software | | | |
|-------------------------------------|--|-------|----|
| Functionality | Items | Marks | |
| | [TM] Bank tellers view any account balance (TM) | /1 | |
| | [Common] Check using a non-existing account | /1 | /5 |
| View Transaction History | [IB and ATM] Customers view their own transaction history within a period in the past | /1 | |
| | Check customer viewing other people's transaction history | /1 | |
| | [TM] Bank tellers view customer's transaction history within a period of time in the past | /1 | |
| | [Common] Check using a non-existing account | /1 | |
| | Check for invalid period (e.g. invalid date, future date – current time is the demo time or when the tutors run the system on their own) | /1 | /5 |
| Deposit [ATM and TM only] | [IB] - Should not have this feature | /0.5 | |
| | [ATM] Customers deposit money into an existing account which can be other people's account | /1 | |
| | [TM] Bank tellers deposit money into an existing account | /1 | |
| | Check customer accessing this function | /0.5 | |
| | [Common] Check using a non-existing account | /1 | |
| | Check depositing zero or negative amount into an existing account | /1 | /5 |
| Withdrawal [ATM and TM only] | [IB] - Should not have this feature | /0.5 | |
| | [ATM] Customers withdraw money from their own accounts | /1 | |
| | Check customer withdrawing money from other people's account | /0.5 | |
| | [TM] Bank tellers withdraw money from an existing account | /1 | |
| | Check customer accessing this function | /0.5 | |
| | [Common] Check using a non-existing account | /0.5 | |
| | Check for withdrawing zero or negative amount | /0.5 | |
| | Check for withdrawing amount greater than the balance | /0.5 | /5 |
| Money Transfer | [IB and ATM] Customers transfer money from one of their own accounts into another account of their own | /2 | |
| | Check customer transferring money from other people's account | /1 | |
| | [TM] Bank tellers transfer money from one account into another | /2 | |
| | Check customer accessing this function | /1 | |

HIT3119/8119 Enterprise Java
Assignment 1 Final version
Mark Sheet

| Software | | | |
|-----------------------|--|--|-----|
| Functionality | Items | Marks | |
| | [Common] Check for non-existing account (source and target) Check for transferring zero or negative amount Check for transferring amount greater than the current balance | /2 /1 /1 | /10 |
| Bill Payment | [IB] Customers pay bill to a company account Check customer paying bill from other people's account Check using existing personal account as target account Check using non-existing account (source and target) Check paying bill with zero or negative amount Check paying bill with amount greater than the current balance [ATM and TM] - Should not have this feature | /1 /0.5 /1 /0.5 /0.5 /0.5 /1 | /5 |
| Sub-total (a1) | = sum of the above marks | | /50 |
| Deduction (a2) | Missing ATM Application – Deduct 10% of (a) Missing TM Application – Deduct 10% of (a) Missing Web client – Deduct 20% of (a) [no Servlets/JSPs/JSFs] | | /50 |
| Sub-total (a) | = (a1) – (a2) | | /50 |

| Documentation | | | |
|--|---|-----------|-----|
| Design and Coding | Items | Marks | |
| Overall Architecture and Object Orientation | Architecture (Diagrams, Proper use of technologies and Descriptions) Object Orientation (Design of EJBs and other client applications) | /10 /5 | /15 |
| Database Design | Proper Design of Database for the Software (ER Diagram) | | /5 |
| Appropriate use of Web technologies | Proper use of Servlets as the Control and JSPs as the View, or JSFs for managing the flow | | /5 |
| Appropriate use of EJBs | Proper use of Session Beans (as the Façade) and Entity Class (as the Data Model) | | /5 |
| Instructions | User Manual (Simple, clear and precise instructions) | | /5 |
| Code Quality | Randomly pick a Client Application, a Session Bean, an Entity Class, a Servlet, a JSP and/or a JSF; then check the following: Style of programming (e.g. Proper use of variable names) Clarity (Coding comments – in line comments) | /5 /5 | /10 |
| Sub-total (b) | | | /45 |

**HIT3119/8119 Enterprise Java
Assignment 1 Final version
Mark Sheet**

| Deployment | | | |
|-------------------------|--|--------------|-------------|
| Items | | Marks | |
| Deployment | Client and Java EE application server in one machine | | /5 |
| Sub-total (c) | | | /5 |
| Total (d) | = (a) + (b) + (c) | | /100 |
| Late Penalty (e) | Less 10% of (d) per day | | /100 |
| Final mark | = [(d) – (e)]/10*3 | | /30 |

Marker's comments: