**Authored By**

Divya Jayaprakash (dxj160830)

MIS 6308.0W1 - SAPM - Summer 2017

Dr. Srinivasan Raghunathan

**System Analysis**

**And**

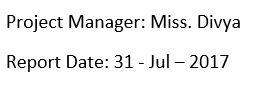
**Project Management**

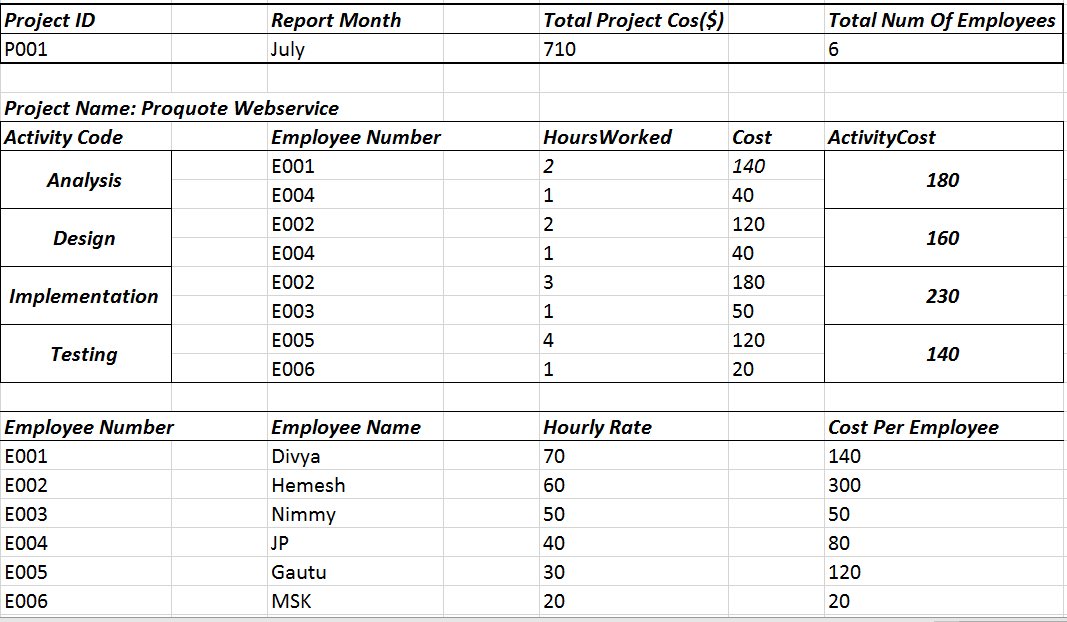
1. Consider the following use case used by a project manager to prepare project cost report for various projects he/she is managing

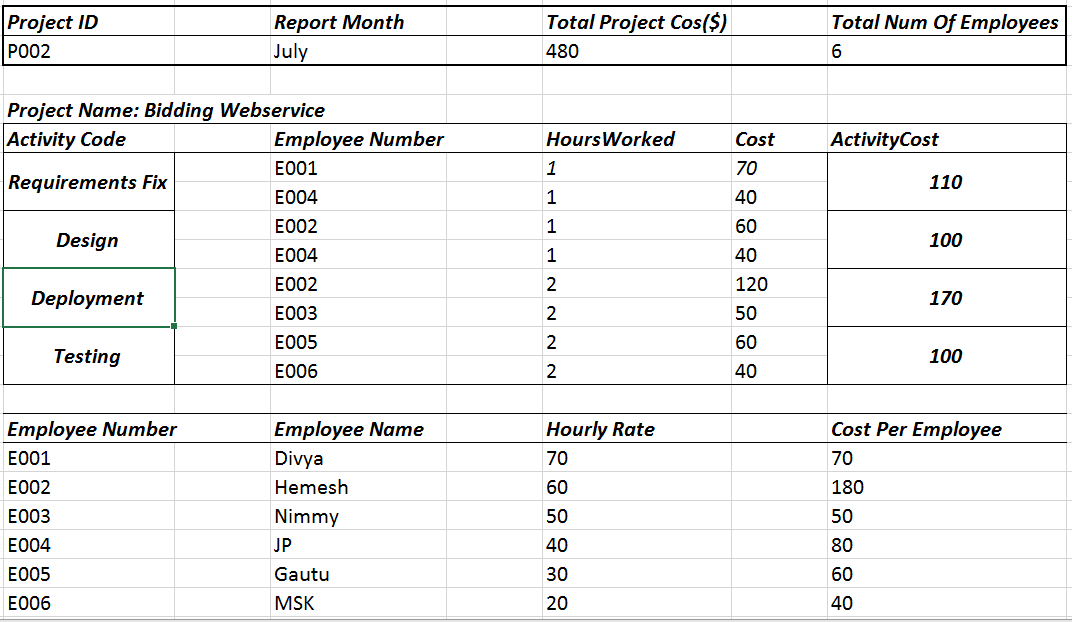
a) Suggest a layout for the report. The layout should indicate what data will be shown in the report.



**Monthly Project Cost Report**





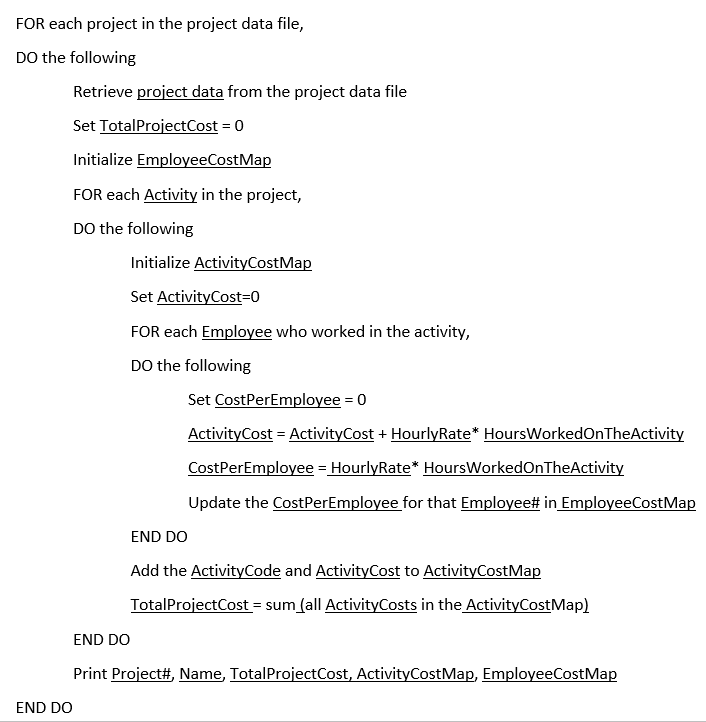


b) Show the revised Use case description and data dictionary to generate the new cost report.

# Use Case for PrepareProjectCostReport

**Trigger:** End of each month

**Normal Flow of Events:**



**Data Dictionary:**

Project data = **Project#** + Name + ProjectBeginDate + ProjectEndDate + {**Employee#** + HourlyRate} +

{**ActivityCode** + short Description + {**Employee#Worked** + HoursWorkedOnTheActivity}}

**2.** A company keeps the following database table to track account balance of customers: (5 points)

(a) Suggest a set of one or more database tables to produce the aged balance report.

***Tables to Produce The Aged Balance Report***

CustomerDetails (**Customer#**, CustomerName)

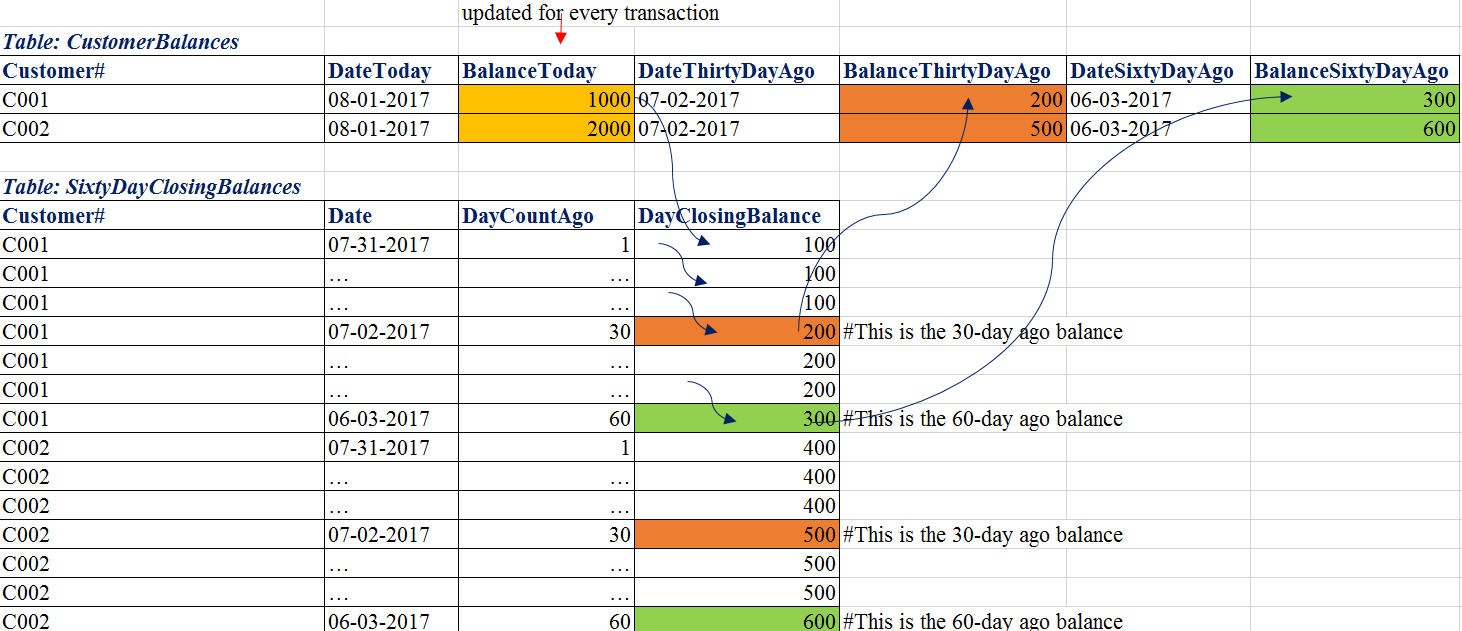
CustomerBalances (**Customer#**, DateToday, BalanceToday, DateThirtyDayAgo, BalanceThirtyDayAgo,

DateSixtyDayAgo, BalanceSixtyDayAgo)

SixtyDayClosingBalances (**Customer#**, **Date**, DayCountAgo, DayClosingBalance)

/\* This table has exactly 60 records per customer for the sixty-day closing balances \*/

|  |  |
| --- | --- |
|  |  |
|  | |



(b) Briefly describe how these tables will be updated and how the aged balance report can be generated using these tables.

***The blue arrow marks on the tables indicate the automated nightly batch process.***

***The whole CustomerBalances table can be used to generate short aged balance report.***

***Both CustomerBalances table and SixtyDayClosingBalances table can be used to generate the detailed aged balance report.***

**The original process shall still be carried out:**

**Pseudocode:**

For every transaction DO,

IF (new order Placed)

THEN in CustomerBalances table, BalanceToday = BalanceToday + Order Amount

END IF

IF (payment Made)

THEN in CustomerBalances table, BalanceToday = BalanceToday - Payment Amount

END IF

**Additionally, a midnight automated batch process shall be run when each day begins to do the following:**

The blue arrow marks on the tables indicate the nightly batch process

**Pseudocode:**

FOR each new day,

DO the following

Update DateToday, DateThirtyDayAgo, DateSixtyDayAgo

Remaining “BalanceToday” of yesterday = Opening “BalanceToday” of today

FOR each Customer,

DO the following

(In SixtyDayClosingBalances table)

Move all DayClosingBalance and its Date by one step to one previous date

Update yesterday date and yesterday balance where DayCountAgo = 1

Take the 30-day ago balance and the 60-day ago balance

(In the CustomerBalances table)

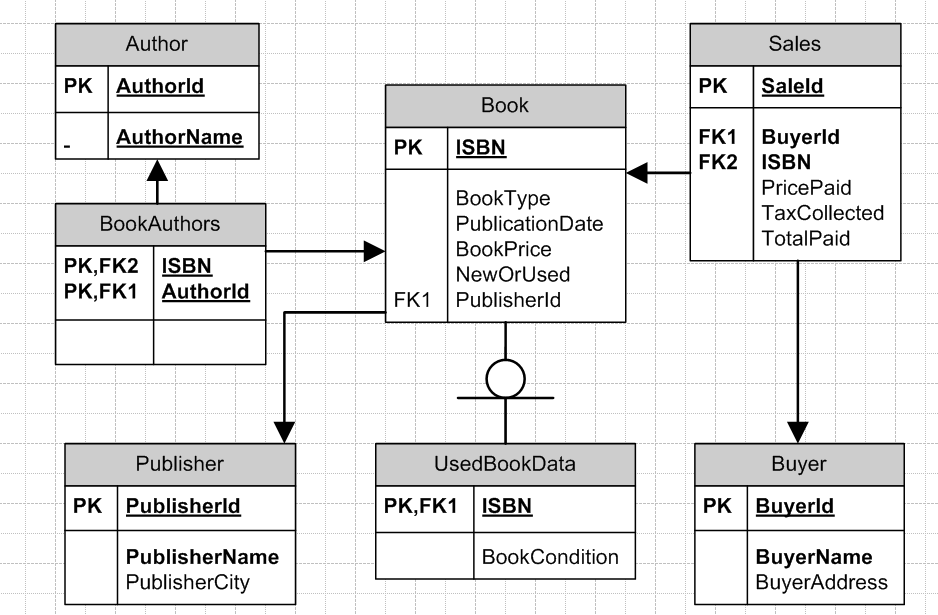
Update “BalanceThirtyDayAgo” and “BalanceSixtyDayAgo”

END DO

END DO

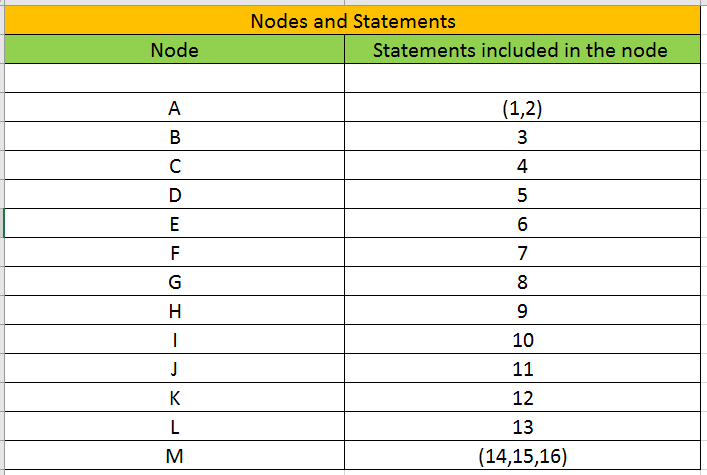
**3. Consider the following data dictionary. (5 points). Develop a database in BCNF to store the data contained in the above data dictionary**

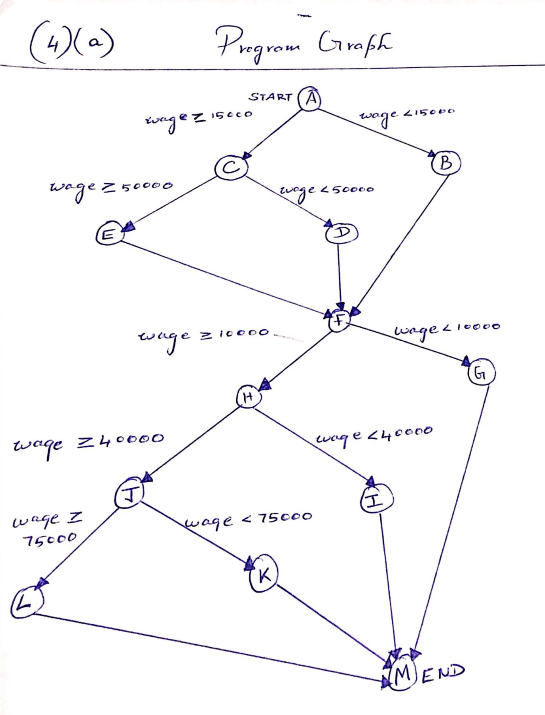
**BOOK SALE DATABASE**



**4. Consider the following program. (10 points)**

**a) Draw the program graph for the above program.**





**b) What test data will you use to test the above program using statement coverage?**

**Statement Coverage:**

**Lines: Test Data**

1,2,3,7,8,14,15,16: **(9300)**

1,2,4,5,7,9,10,14,15,16: **(21300)**

1,2,4,6,7,9,11,12,14,15,16: **(51300)**

1,2,4,6,7,9,11,13,14,15,16: **(81300)**

**c) What test data will you use to test the above program using branch coverage?**

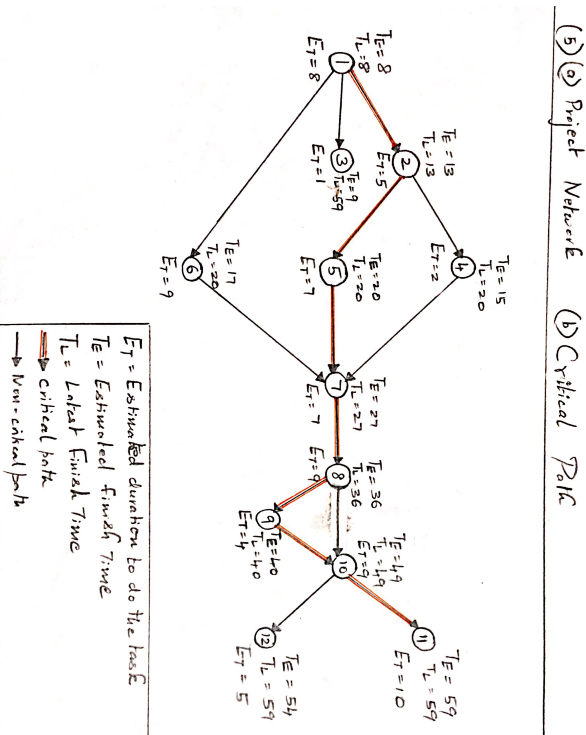
Total Branching nodes are**: A, C, F, H and J**

|  |  |  |
| --- | --- | --- |
| **TEST data for Branch Coverage** | **Branching SPLIT coverage in the Program Graph** | **BRANCH in Program graph** |
| (9300) | A node – Right Branch  F node – Right Branch | A->B->F->G->M |
| (21300) | A node – Left Branch  F node – Left Branch  H node – Right Branch  C node – Right Branch | A->C->D->F->H->I->M |
| (51300) | C node – Left Branch  H node – Left Branch  J node - Right Branch | A->C->E->F->H->J->K->M |
| (81300) | J node – Left Branch | A->C->E->F->H->J->L->M |

**d) What test data will you use to test the above program using path coverage?**

|  |  |  |
| --- | --- | --- |
| Test Data | Nodes in Program graph | Lines |
| (9300) | A->B->F->G->M | 1,2,3,7,8,14,15,16 |
| (11300) | A->B->F->H->I->M | 1,2,3,7,9,10,14,15,16 |
| (21300) | A->C->D->F->H->I->M | 1,2,4,5,7,9,10,14,15,16 |
| (41300) | A->C->D->F->H->J->K->M | 1,2,4,5,7,11,12,14,15,16 |
| (51300) | A->C->E->F->H->J->K->M | 1,2,4,6,7,9,11,12,14,15,16 |
| (81300) | A->C->E->F->H->J->L->M | 1,2,4,6,7,9,11,13,14,15,16 |

5. Consider a project with the following data (10 points) a) Draw the project network.



|  |  |  |  |
| --- | --- | --- | --- |
| **Activity** | **Estimated Finish Time (TE) week** | **Latest Finish Time (TL) week** | **Slack = (TL) - (TE)**  **weeks** |
| 1 | 8 | 8 | 0 |
| 2 | 13 | 13 | 0 |
| 3 | 9 | 59 | 50 |
| 4 | 15 | 20 | 5 |
| 5 | 20 | 20 | 0 |
| 6 | 17 | 20 | 3 |
| 7 | 27 | 27 | 0 |
| 8 | 36 | 36 | 0 |
| 9 | 40 | 40 | 0 |
| 10 | 49 | 49 | 0 |
| 11 | 59 | 59 | 0 |
| 12 | 54 | 59 | 5 |

**b) Identify the critical activities.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Activity** | **Estimated Finish Time (TE) week** | **Latest Finish Time (TL) week** | **Slack = (TL) - (TE)**  **weeks** |
| 1 | 8 | 8 | 0 |
| 2 | 13 | 13 | 0 |
| 5 | 20 | 20 | 0 |
| 7 | 27 | 27 | 0 |
| 8 | 36 | 36 | 0 |
| 9 | 40 | 40 | 0 |
| 10 | 49 | 49 | 0 |
| 11 | 59 | 59 | 0 |

**Critical Path = 1-> 2 -> 5 -> 7-> 8-> 9-> 10 -> 11**

c) What is the least number of weeks required to complete the project?

**59 weeks (adding the estimated duration of the activities on the above critical path)**

d) Which activity has the greatest slack?

**activity 3**

|  |  |  |  |
| --- | --- | --- | --- |
| **Activity** | **Estimated Finish Time (TE)**  **week** | **Latest Finish Time (TL) week** | **Slack = (TL) - (TE) weeks** |
| 3 | 9 | 59 | 50 |