CS 6400 Database Project

BuzzBuy Data Warehouse

**Team 34**

The first phase of the project deals with the Analysis and Specifications.

Data Entities and Types

**User**

|  |  |  |
| --- | --- | --- |
| EmployeeID |  | Unique, Primary |
| FirstName |  |  |
| Password |  |  |

**District**

|  |  |  |
| --- | --- | --- |
| District\_ID |  | Primary Key |
| DistrictNumber |  |  |

**AuditLog**

|  |  |  |
| --- | --- | --- |
| AuditLog\_ID |  | Primary Key |
| ReportName |  |  |
| Timestamp |  |  |

**Holidays**

|  |  |  |
| --- | --- | --- |
| Holiday\_ID |  | Primary Key |
| HolidayName |  |  |
| Date |  |  |

**Store**

|  |  |  |
| --- | --- | --- |
| StoreNumber |  | Unique, Primary Key |
| PhoneNumber |  |  |
|  |  |  |

**City**

|  |  |  |
| --- | --- | --- |
| City\_ID |  | Primary Key |
| CityName |  |  |
| State |  |  |
| Population |  |  |

**Product**

|  |  |  |
| --- | --- | --- |
| PID |  | Unique, Primary Key |
| ProductName |  |  |
| RetailPrice |  |  |

**Manufacturer**

|  |  |  |
| --- | --- | --- |
| Manufacturer\_ID |  | Primary Key |
| ManufacturerName |  | Unique |

**Discount**

|  |  |  |
| --- | --- | --- |
| Discount\_ID |  | Primary Key |
| DiscountDate |  |  |
| DiscountPrice |  |  |

**Category**

|  |  |  |
| --- | --- | --- |
| Category\_ID |  | Primary Key |
| CategoryName |  |  |

**SellsTransactions**

|  |  |  |
| --- | --- | --- |
| Transaction\_ID |  | Primary Key |
| DateSold |  |  |
| QuantitySold |  |  |
| StoreNumber |  |  |
| PID |  |  |

Relationship Entities and Types

**Users2Districts**

|  |  |  |
| --- | --- | --- |
| EmployeeID |  |  |
| District\_ID |  |  |

**District2Stores**

|  |  |  |
| --- | --- | --- |
| District\_ID |  |  |
| StoreNumber |  |  |

**User2AuditLogs**

|  |  |  |
| --- | --- | --- |
| EmployeeID |  |  |
| AuditLog\_ID |  |  |

**Store2Cities**

|  |  |  |
| --- | --- | --- |
| StoreNumber |  |  |
| City\_ID |  |  |

**Discounts2Products**

|  |  |  |
| --- | --- | --- |
| Discount\_ID |  |  |
| PID |  |  |

**Manufacturer2Products**

|  |  |  |
| --- | --- | --- |
| Manufacturer\_ID |  |  |
| PID |  |  |

**Products2Categories**

|  |  |  |
| --- | --- | --- |
| PID |  |  |
| Category\_ID |  |  |

Business Logic Constraints

Task Decomposition & Abstract Code

Login

* Lock Types: Read-Only **User** Table
* Number of Locks: Single
* Enabling Conditions: None
* Frequency: Frequent
* Consistency: Not Critical, order is not critical
* Subtasks: Subtask is not needed, no further decomposition needed

Login – Abstract Code

* User enters employee id and password (last 4 ssn + “-” + LastName) input fields
* If data validation is success for both employee id and password input fields, then:
  + When Login button is clicked:
    - If User record is found but user password != (last 4 ssn + “-” + LastName)
      * Go back to login form with error message
    - Else:
      * Store login information as session variable (EmployeeID)
      * Go to Main Menu Form

View Statistics

* **Lock Types:** Read-Only Lookups
* **Number of Locks:** Several schema constructs are needed
* **Enabling Conditions:** Successful login
* **Frequency:** Medium (Upon successful login)
* **Consistency:** Not Critical
* **Subtasks:**
  + List available reports with navigation options.

View Holidays

* Lock Types: 2 Read-Only Lookups of HolidayName and HolidayDate from **Holiday** Table
* Number of Locks: Several different schema constructs are needed
* Enabling Conditions: Both enabled to user’s login screen
* Frequency: Occasionally
* Consistency: Not Critical, Even if profile being edited by the user while  another employee is looking at it
* Subtasks:
  + User clicked **View Holidays** from main menu
  + Run View Holiday Task: query information about Holidays

View Holidays – Abstract Code

* Check if user has been granted to access to all districts
* If all districts are accessible, then:
  + View all holidays

Edit Holidays

* Lock Types: Write Locks on Holiday Table
* Number of Locks: Several schema constructs are needed
* Enabling Conditions: User has access to all districts
* Frequency: Infrequent (Only certain users can add holidays)
* Consistency: Critical (To prevent duplicate holidays)
* Subtasks:
  + User clicks "Edit Holidays" from the main menu.
  + Validate user's permission to add holidays.
  + If permission is valid:
    - Check if the holiday already exists for the date.
    - If not, allow user to add a new holiday.
    - Record which user added the holiday.

Edit Holidays – Abstract Code

* Check if user has been granted to access to all districts
* If all districts are accessible, then:
  + View all holidays
  + Select a holiday to edit
  + “Delete” is also supported from the same view “All Holidays”

View Reports

* Lock Types: Read-Only Lookups
* Number of Locks: Several schema constructs are needed
* Enabling Conditions: Successful login
* Frequency: Occasional (Reports viewed based on user needs)
* Consistency: Not Critical
* Subtasks:
  + User selects a report from the list.
  + Query the data warehouse for the selected report.
  + Display the report data to the user.

View Reports – Abstract Code

* Check if user has been granted to access to Reports
* If a user is granted to access, then:
  + View all available reports
  + Choose a report to generate a report
  + Upon completion of generating a report
    - User is able to export the output to different formats of files (PDF or CSV)

View Audit Logs

* Lock Types: Read-Only Lookups on Audit Log Table
* Number of Locks: Several schema constructs are needed
* Enabling Conditions: User has audit log permissions
* Frequency: Infrequent (Only users with permissions can view)
* Consistency: Not Critical
* Subtasks:
  + User clicks "View audit log" from the main menu.
  + Query the most recent audit log records.
  + Display the audit log records in a table format, highlighting entries where the user has access to all districts.

View Audit Logs – Abstract Code

* Check if user has been granted to access to Audit Logs
* If a user is granted to access, then:
  + View all available audit logs
    - Search by Date, User, ReportName
    - Sort by Date
  + Choose an audit log to view
  + A selected log or multiple logs can be printed or exported to other file formats