Personal Money Budget App

**INDEX**

**Final Report Contents:**

· Business application description

· Swim Lane Diagrams

· Business process of the application

· Describe business metrics that will be generated

· Logical schema – Data model

· Physical schema – Database dictionary

· DML in form of a SQL script

· Database backup file with sample business data

· SQL queries for each table – create, insert, update, and delete

· SQL queries for business metrics

· Include results of running the query

· Include business metrics reports

· For delete & update queries show rows in table before and after delete query

· Project summary

▪ Summarize your experience with this exercise

▪ What was the hardest part of this project?

▪ What problems did you run against in this project?

▪ How did you solve these problems?

▪ If you were to do this project again, what methodology would you follow ▪ Suggestions for how to refine this project for the next class

**1. Business application description:**

Budget Management Application Domain​: Financial management service

The project is about the database for a mobile app that will help its users in personal financial budget management. The users will be asked to enter personal information like name, age, financial goals, etc. Also, we will have options to store user's bank details, credit card details, and the loan details so any transactions from any of these can be fed into the app automatically. All this information can be saved at the time of registration (Sign up) to the app and also any other time user wants to update it. The expenses that are made using the bank accounts and cards of the user that are linked with this app will automatically be updated as "Transactions". These transactions are classified in "categories of purchases" (for example, rent, entertainment, utilities, travel expenses, education, etc). All these modes can be saved in the app for monitoring the expenses that a user makes. In addition, the user can save the recurring expenses like example monthly mobile bill payments, credit card dues, and loan installments. Using this data app will give a reminder to users to pay before the due date which we term as "Future Payments". Overall, this business application will give a clear view of the user’s budget and expenses. This can give us many interesting analytical findings like most spent categories, week wise spending’s, etc. Also, we can club many such reports to give the user an interactive dashboard of the transaction for a current and recent month.

**2. Swim Lane Diagrams:**

**Use cases:**

**Case 1: Login Process:**

User 1 downloads the app , subscribe to the service and then enters his personal details and financial information. User can maintain a passbook which can have multiple payments such as Wallet, net banking, credit cards so that he can track his accounts or balances credits and debits.

**Case 2:**​ ​**User Transaction Process:**

John goes to Target to purchase grocery for his home. He buys milk, bread which totalled to amount $20. John pays the bill with his credit card and immediately receives a SMS and Email from his credit card company. The app recognises from the transaction SMS/ Email and automatically categorises as ‘Grocery Expense’. The App reduces the grocery budget set by john by $20.

**Case 3:**​ ​**New Categorization Process:**

User\_ 3 made a donation at GoodWill. The transaction is done through bank wire transfer and the details related to the payment is not specifically categorized in the system. In such a scenario, user can create his own category of spending area, so that transactions related to donations or any service related payments will fall into the specific section.

**Case 4** ​: ​**User Split Bill Process:**

John and his three friends go out lunch where their total bill was $80 which he pays using his credit card. This bill should be equally divided among all of them. They all decided to pay him later. The app recognizes from the transactions sms received on John’s mobile phone and marks it as expense. John updates the transaction to mark it as split among contacts and selects his friends from his contact list. App marked that amount as loan given for these contacts.

**Analytics use case:**

1. Create a report to demonstrate the category wise spendings for the user
2. Create a report to display the summary of previous months spendings.
3. Create a chart to display the week wise spendings.
4. Report demonstrating the most spent categories and its comparison with previous months.
5. Create a chart to display the months in which user overrun the budget.

**SQL queries for each table – create, insert, update, and delete:**

CREATE TABLE `user\_details` (

`User\_id` int(11) NOT NULL,

`First\_Name` varchar(45) DEFAULT NULL,

`Last\_name` varchar(45) DEFAULT NULL,

`Phone` varchar(15) DEFAULT NULL,

`Email` varchar(45) DEFAULT NULL,

`Address` varchar(100) DEFAULT NULL,

`Created\_Date` datetime DEFAULT NULL,

`Last\_Updated\_Date` datetime DEFAULT NULL,

`monthly\_budget` decimal(10,3) DEFAULT NULL,

PRIMARY KEY (`User\_id`)

)

CREATE TABLE `user\_login\_details` (

`User\_id` int(11) NOT NULL,

`Login\_id` varchar(45) NOT NULL,

`Password` varchar(45) DEFAULT NULL,

`Security\_Question\_1` int(11) DEFAULT NULL,

`Answer\_1` varchar(45) DEFAULT NULL,

`Security\_Question\_2` int(11) DEFAULT NULL,

`Answer\_2` varchar(45) DEFAULT NULL,

`Password\_Change\_Date` date DEFAULT NULL,

`Created\_Date` date DEFAULT NULL,

`Last\_Updated\_Date` date DEFAULT NULL,

PRIMARY KEY (`User\_id`,`Login\_id`),

CONSTRAINT `R\_6` FOREIGN KEY (`User\_id`) REFERENCES `user\_details` (`User\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `user\_categories` (

`User\_id` int(11) NOT NULL,

`User\_Category\_ID` int(11) NOT NULL,

`Category\_Desc` varchar(50) DEFAULT NULL,

`Created\_Date` date DEFAULT NULL,

`Last\_Updated\_Date` date DEFAULT NULL,

PRIMARY KEY (`User\_Category\_ID`),

KEY `R\_8` (`User\_id`),

CONSTRAINT `R\_8` FOREIGN KEY (`User\_id`) REFERENCES `user\_details` (`User\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `transaction\_split\_details` (

`Transaction\_ID` int(11) NOT NULL,

`Split\_ID` int(11) NOT NULL,

`Split\_with\_contact` varchar(50) DEFAULT NULL,

`Split\_amount` decimal(10,2) DEFAULT NULL,

`Created\_Date` date DEFAULT NULL,

`Last\_Updated\_Date` date DEFAULT NULL,

PRIMARY KEY (`Transaction\_ID`,`Split\_ID`),

CONSTRAINT `R\_10` FOREIGN KEY (`Transaction\_ID`) REFERENCES `transaction\_details` (`Transaction\_ID`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `transaction\_details` (

`Payment\_method\_id` int(11) NOT NULL,

`Transaction\_ID` int(11) NOT NULL,

`Category\_ID` int(11) DEFAULT NULL,

`Transaction\_Date` date DEFAULT NULL,

`Transaction\_Amt` decimal(10,3) DEFAULT NULL,

`Created\_Date` date DEFAULT NULL,

`Last\_Updated\_date` date DEFAULT NULL,

PRIMARY KEY (`Transaction\_ID`),

KEY `R\_9` (`Payment\_method\_id`),

CONSTRAINT `R\_9` FOREIGN KEY (`Payment\_method\_id`) REFERENCES `payment\_method\_details` (`Payment\_method\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `security\_questions` (

`Question\_ID` int(11) NOT NULL,

`Security\_Q\_Desc` varchar(100) DEFAULT NULL,

`Created\_Date` date DEFAULT NULL,

`Last\_Updated\_Date` date DEFAULT NULL,

PRIMARY KEY (`Question\_ID`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `payment\_method\_details` (

`User\_id` int(11) NOT NULL,

`Payment\_method\_id` int(11) NOT NULL,

`Pymt\_method\_type` varchar(1) DEFAULT NULL,

`Created\_Date` date DEFAULT NULL,

`Last\_Updated\_Date` date DEFAULT NULL,

PRIMARY KEY (`Payment\_method\_id`),

KEY `R\_7` (`User\_id`),

CONSTRAINT `R\_7` FOREIGN KEY (`User\_id`) REFERENCES `user\_details` (`User\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `future\_bills` (

`User\_id` int(11) DEFAULT NULL,

`Bill\_id` int(11) NOT NULL,

`Invoice\_number` varchar(25) DEFAULT NULL,

`Bill\_date` date DEFAULT NULL,

`Due\_date` date DEFAULT NULL,

`Bill\_amount` decimal(10,3) DEFAULT NULL,

`Created\_date` date DEFAULT NULL,

`Last\_update\_date` date DEFAULT NULL,

PRIMARY KEY (`Bill\_id`),

KEY `R\_23` (`User\_id`),

CONSTRAINT `R\_23` FOREIGN KEY (`User\_id`) REFERENCES `user\_details` (`User\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `credit\_card\_details` (

`Payment\_method\_id` int(11) NOT NULL,

`Card\_id` int(11) DEFAULT NULL,

`Credit\_card\_number` varchar(20) DEFAULT NULL,

`Credit\_card\_expiry\_date` date DEFAULT NULL,

`Issuing\_Company` varchar(100) DEFAULT NULL,

`Created\_date` date DEFAULT NULL,

`Last\_Updated\_Date` date DEFAULT NULL,

PRIMARY KEY (`Payment\_method\_id`),

CONSTRAINT `R\_22` FOREIGN KEY (`Payment\_method\_id`) REFERENCES `payment\_method\_details` (`Payment\_method\_id`) ON DELETE CASCADE

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `categories` (

`Category\_id` int(11) NOT NULL,

`Category\_Desc` varchar(50) DEFAULT NULL,

`Created\_Date` date DEFAULT NULL,

`Last\_Updated\_Date` date DEFAULT NULL,

PRIMARY KEY (`Category\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `bank\_details` (

`Payment\_method\_id` int(11) NOT NULL,

`Bank\_id` int(11) DEFAULT NULL,

`Bank\_Name` varchar(50) DEFAULT NULL,

`Account\_Number` varchar(50) DEFAULT NULL,

`Created\_Date` date DEFAULT NULL,

`Last\_Updated\_Date` date DEFAULT NULL,

PRIMARY KEY (`Payment\_method\_id`),

CONSTRAINT `R\_15` FOREIGN KEY (`Payment\_method\_id`) REFERENCES `payment\_method\_details` (`Payment\_method\_id`) ON DELETE CASCADE

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE `audit\_transactions` (

`Payment\_method\_id` int(11) NOT NULL,

`Transaction\_ID` int(11) NOT NULL,

`New\_Transaction\_ID` int(11) NOT NULL AUTO\_INCREMENT,

`Category\_ID` int(11) DEFAULT NULL,

`Transaction\_Date` date DEFAULT NULL,

`Transaction\_Amt` decimal(10,3) DEFAULT NULL,

`New\_Transaction\_Amt` decimal(10,3) DEFAULT NULL,

`Record\_Status` varchar(10) DEFAULT NULL,

`First\_Created\_Date` date DEFAULT NULL,

`updated\_at` timestamp NOT NULL DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

`updated\_by` varchar(30) NOT NULL,

PRIMARY KEY (`New\_Transaction\_ID`),

KEY `Payment\_method\_id` (`Payment\_method\_id`),

CONSTRAINT `audit\_transactions\_ibfk\_1` FOREIGN KEY (`Payment\_method\_id`) REFERENCES `payment\_method\_details` (`Payment\_method\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=2 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;