**PERSON TO CONTACT**

* + **jmware@mtu.edu**

Joseph

Atharva

Reyn

Emily

John

Irene

Lucas

Trent

Our team SPECIFICALLY

* Bank
  + ~~Open savings account~~
  + ~~Make a deposit~~
  + ~~Make a withdrawal~~
  + ~~Open a bank account~~
  + ~~Close a bank account~~
* ~~Credit Scores~~ 
  + ~~Open line of credit~~
  + ~~Close line of credit~~
  + ~~Request credit score~~
  + ~~Request a credit review~~
* Loan / Borrowing
  + ~~Apply for a Loan~~
  + ~~Make Loan payment~~
  + ~~Defer payment on my loan~~
  + ~~Finish paying off my Loan~~
* ~~Buying~~ 
  + ~~Buy property / large investments~~
  + ~~Get a mortgage~~

# Credit Activities

## Credit activity diagram

Need to be done

## Credit activities

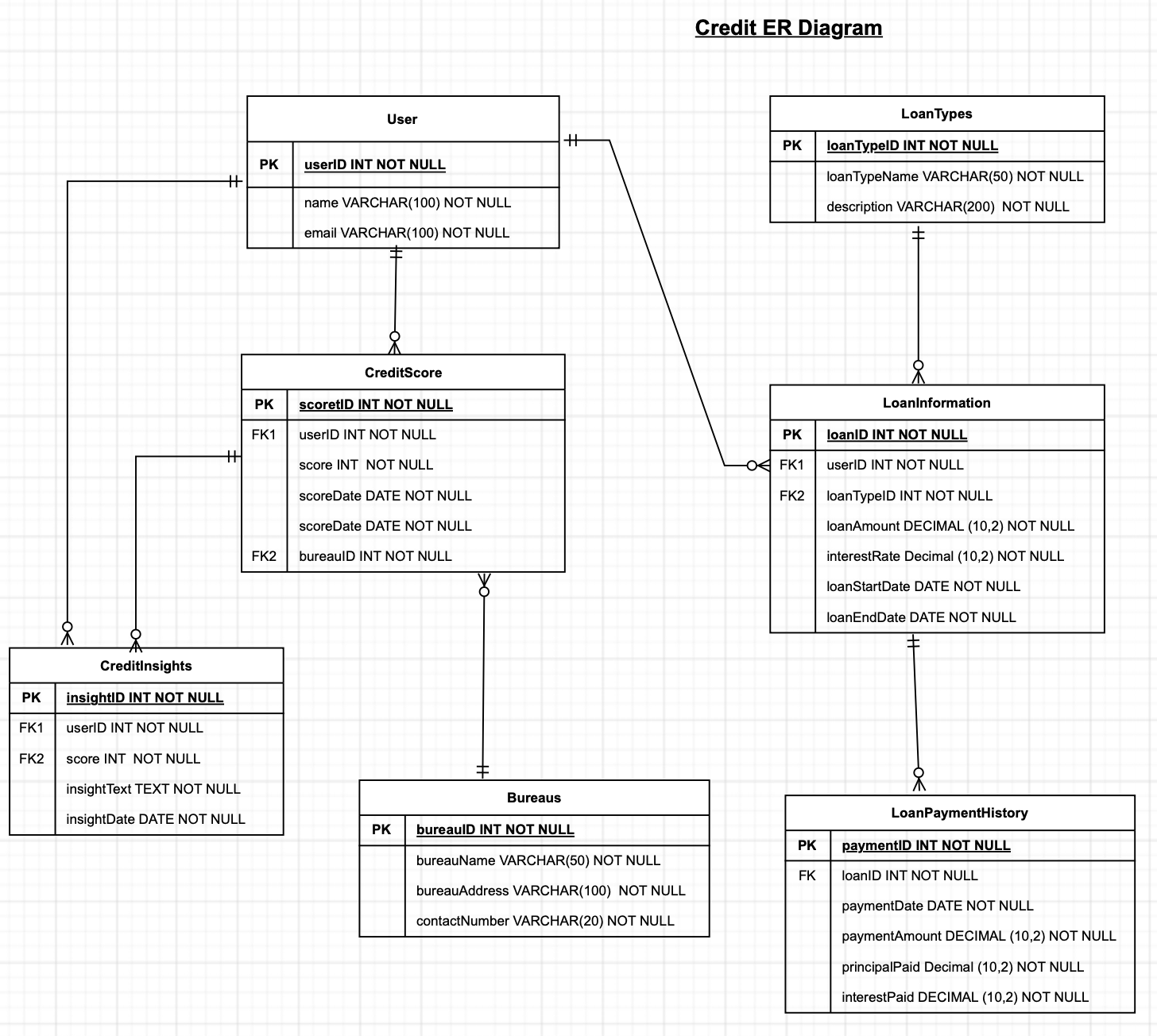
A summary of activities required.

## Credit Entities/Attributes

## 

| Entity | Description | Attributes | Example |
| --- | --- | --- | --- |
| User | Contains user details | UserID,Name,Email | userID: 1  Name: John Mware  email: jmware@mtu.edu |
| Bureaus | Contains Bureau details | bureauID INT PK,  bureauName VARCHAR(100), bureauAddress VARCHAR(150),  contactNumber VARCHAR(15) | bureauID: 201  bureauName: Experian  bureauAddress: 123 Credit St, Finance City  contactNumber: (123) 456-7890 |
| CreditScore | Contains users credit scores. | scoreID INT PK,  userID INT FK,  score INT,  reportDate DATE,  creditBureau INT FK | scoreID: 101  userID: 1  score: 750  reportDate: 2023-10-25  BureauID: 201 |
| loanTypes | Contains loan types details. | loanTypeID INT PK,  loanTypeName VARCHAR(50),  description TEXT | loanTypeID: 401  loanTypeName: Personal Loan  description: A loan for personal use, unsecured and based on the borrower's credit history. |
| LoanInformation | Contains users' loan details. | loanID INT PK,  userID INT FK,  loanType INT FK,  loanAmount DECIMAL(10,2),  interestRate DECIMAL(10,2),  loanStartDate DATE, loanEndDate DATE | loanID: 501  userID: 1  loanType: 401  loanAmount: 5000.00  interestRate: 5.00  loanStartDate: 2023-01-01  loanEndDate: 2024-01-01 |
| LoanPaymentHistory | Stores loan payment history. | paymentID INT PK,  loanID INT FK,  paymentDate DATE,  paymentAmount DECIMAL(10,2),  principalPaid DECIMAL(10,2),  interestPaid DECIMAL(10,2) | paymentID: 1001  loanID: 501  paymentDate: 2023-10-25  paymentAmount: 200.50  principalPaid: 150.00  interestPaid: 50.50 |
| CreditInsights | Stores users credit score insights. | insightID INT PK,  userID INT FK,  scoreID INT FK,  insightText TEXT,  insightDate DATE | insightID: 301  userID: 1  scoreID: 101  insightText: "Your credit score has improved by 20 points in the last month."  insightDate: 2023-10-25 |

## Credit ER Diagram



## 

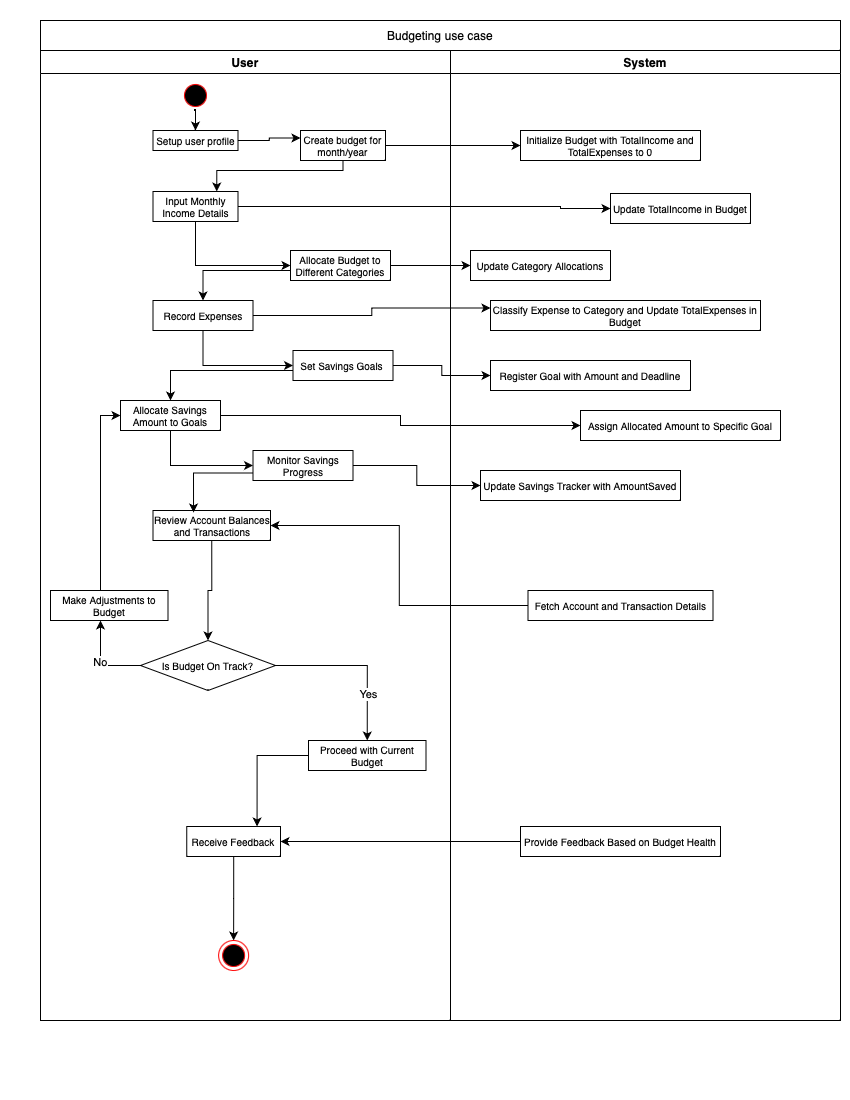
## Credit Entity Relationships

* User - Has -> Credit Score
  + One User can have multiple Credit Scores, but each Credit Score is associated with one User.
* User - Receives -> Credit Insight
  + One User can receive multiple Credit Insights, but each Credit Insight is associated with one User.
* User - Takes -> Loan
  + One User can take multiple Loans, but each Loan is taken by one User.
* Credit Score - Generates -> Credit Insight
  + One Credit Score can generate multiple Credit Insights, but each Credit Insight is generated from one Credit Score.
* Loan - Has -> Payment History
  + One Loan can have multiple Payment Histories, but each Payment History is associated with one Loan.
* Loan Type - Categorizes -> Loan
  + One Loan Type can categorize multiple Loans, but each Loan is categorized by one Loan Type.
* Bureau - Reports -> Credit Score
  + One Bureau can report multiple Credit Scores, but each Credit Score is reported by one Bureau.

## 

# Budgeting Activities

## Budgeting activity diagram



## Budgeting activities

1. Set Up User Profile - The user starts by creating or updating their profile, entering personal information such as name, email, and any other necessary details.
2. Create Budget for a Specific Month/Year - The user initiates a new budget for a specific month and year. At this point, the system initializes the total income and total expenses for this budget to zero.
3. Input Monthly Income Details - The user enters the details of their monthly income. As they provide this information, the system updates the budget's total income.
4. Allocate Budget to Different Categories - The user divides their budget among different expense categories. The system updates these allocations in respective records.
5. Record Expenses - The user records expenses incurred throughout the month. The system classifies each expense to its respective category and updates the budget's total expenses.
6. Set Savings Goals - The user determines savings goals, specifying the goal amount and desired deadline. The system registers each goal with its amount and deadline.
7. Allocate Savings Amount to Goals - Based on the user's financial status and priorities, they allocate a specific amount of savings to each set goal. The system then assigns the allocated amount to the specific goal.
8. Monitor Savings Progress - The user reviews their progress towards achieving their savings goals. The system provides an updated savings tracker, showing the amount saved so far for each goal.
9. Review Account Balances and Transactions - To ensure that they're staying on track, the user reviews their account balances and transactions. The system fetches and displays the account and transaction details.
   1. After the review, the user evaluates whether their budget is on track or if they're overspending.
   2. If the budget is on track, the user proceeds without making adjustments.
   3. If adjustments are needed, the user modifies the budget, revising allocations or cutting unnecessary expenses.
10. Receive Feedback - Finally, based on the user's budgeting activities, decisions, and financial status, the system provides feedback on the health of their budget. This feedback can include suggestions, warnings, or commendations based on how well the user is adhering to their budget.

## Budgeting Entities/Attributes

| Entity | Description | Attributes | Example |
| --- | --- | --- | --- |
| User | Contains user details | UserID,Name,Email | UserID: 101  Name: "John Doe"  Email: "johndoe@email.com" |
| Budget | Contains budget details | BudgetID,UserID,Month,Year,TotalIncome,TotalExpenses | BudgetID: 2023  Month: 10 (October)  Year: 2023  TotalIncome: $5000.00  TotalExpenses: $3200.00 (This would be calculated based on the expenses added to this budget.) |
| Expense | Contains user expenses | ExpenseID,BudgetID, CategoryID, Amount, ExpenseDate | ExpenseID: 456  Amount: $150.00  ExpenseDate: "2023-10-05" (5th October 2023)  Category: "Groceries" (This would be set by the system dynamically) |
| BudgetCategoryAllocation | Manages budget allocations to different categories | AllocationID, BudgetID, CategoryID, AllocatedAmount, AllocationDate | AllocationID: 123,  BudgetID: 2023,  CategoryID: 1,  AllocatedAmount: $400.00,  AllocationDate: "2023-10-01" |
| Income | Contains details of user's income | IncomeID, BudgetID, Source, Amount, IncomeDate | IncomeID: 789,  BudgetID: 2023,  Source: "Salary",  Amount: $4500.00,  IncomeDate: "2023-10-01" |
| *Category* | *Different categories* | *CategoryID, Name* | *CategoryID: 1*  *Name: "Groceries"* |
| Goal | Contains user goals with realistic timelines. | GoalID, UserID, Name, Amount, Deadline | GoalID: 789  Name: "Save for Vacation"  Amount: $2000.00  Deadline: "2023-12-31" (31st December 2023) |
| Allocation | Details the amount allocated to specific goals | AllocationID, GoalID, AmountAllocated, AllocationDate | AllocationID: 2001,  GoalID: 1001,  AmountAllocated: $500.00,  AllocationDate: "2023-10-05" |
| Feedback | Contains user feedback regarding budgeting | FeedbackID, UserID, Details, ReportDate | FeedbackID: 3001,  UserID: 101,  Details: "The budgeting tool is very helpful!",  ReportDate: "2023-10-07" |
| Account | Represents user's different accounts like bank accounts | AccountID, UserID, Type, Balance | AccountID: 4001,  UserID: 101,  Type: "Savings",  Balance: $10000.00 |
| Transaction | Contains details of transactions in user accounts | TransactionID, AccountID, CategoryID, Amount, TransactionDate, Type | TransactionID: 5001,  AccountID: 4001,  CategoryID: 1,  Amount: $200.00,  TransactionDate: "2023-10-10",  Type: "Expense" |
| SavingsTracker | Tracks progress towards achieving savings goals | TrackerID, GoalID, AmountSaved, TrackerDate | TrackerID: 6001,  GoalID: 1001,  AmountSaved: $500.00,  TrackerDate: "2023-10-15" |
|  |  |  |  |

## Budgeting ER Diagram

### 

## 

## Budgeting Entities Relationships

* User - Creates -> Budget
  + One User can create multiple Budgets, but each Budget is created by one User.
* Budget - Contains -> Expense
  + One Budget can contain multiple Expenses, but each Expense belongs to one Budget.
* Budget - Contains -> Income
  + One Budget can have multiple Incomes, but each Income is tied to one Budget.
* Budget - Has -> BudgetCategoryAllocation
  + One Budget can have multiple BudgetCategoryAllocations, but each BudgetCategoryAllocation is tied to one Budget.
* Category - Allocated in -> BudgetCategoryAllocation
  + One Category can be allocated in multiple BudgetCategoryAllocations, but each BudgetCategoryAllocation references one Category.
* Category - Categorized as -> Expense
  + One Category can be associated with multiple Expenses, but each Expense belongs to one Category.
* User - Sets -> Goal
  + One User can set multiple Goals, but each Goal is set by one User.
* Goal - Has -> Allocation
  + One Goal can have multiple Allocations, but each Allocation is tied to one Goal.
* Goal - Tracked by -> SavingsTracker
  + One Goal can be tracked by multiple SavingsTrackers, but each SavingsTracker tracks one Goal.
* User - Provides -> Feedback
  + One User can provide multiple Feedbacks, but each Feedback is provided by one User.
* User - Owns -> Account
  + One User can own multiple Accounts, but each Account is owned by one User.
* Account - Registers -> Transaction
  + One Account can register multiple Transactions, but each Transaction is registered to one Account.
* Category - Categorized as -> Transaction
  + One Category can be associated with multiple Transactions, but each Transaction belongs to one Category.