



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

**eProject Document**

Online Banking

| **Group 1** | |
| --- | --- |
| **Members** | Nguyen Tuan Minh - Student1240542  Ngo Van Thuyet - Student1227191  Nguyen Manh Tuan Anh - Student1227199 |
| **Instructor** | Hoang Van Chung |
| **Batch** | C1905M |
| **Semester** | III |

**🙠 Date of issue 07-29-2021 🙢**

**Table of Content**

1. **Problem definition**
   1. ***Introduction***
   2. ***Proposed scenario***
2. **Customer Requirement Specification**
   1. ***Customer Requirement***
   2. ***Hardware – Software Requirements***
3. **Architecture and Design of the System**
4. **Flow Chart**
5. **Data Flow Diagram**
6. **Entity Relationship Diagram**
   1. ***Entity relationship diagram***
   2. ***Entity - Attributes***
7. **Database Design/Structure**
   1. ***Database relationship diagram***
   2. ***Table design***
8. **Some typical interface**
9. **Task Sheet**
10. **Check List of Validation**
11. **Submission Checklist**
12. **Problem Definition**
    1. **) Introduction**

Today, digital technology is applied in almost every field. And banks are no exception. Online banking has appeared and received many positive feedbacks.

Online banking offers customers almost every service traditionally available through a local branch including online deposits, transfers and bill payments. Almost every banking institution has some form of online banking, available both on desktop versions .

* 1. **) Proposed Scenario**

1. Currently, during the pandemic, the demand for online payment is extremely large. Thanks to the development of the internet, we have created an online banking application. The application allows users to:

* Balance Enquiry
* Funds Transfer to another account in the same bank
* Request for cheque book/change of address/stop payment of cheques
* Viewing Monthly and annual statements.

1. So they had approached us in order to help them by creating a web application for them meeting their requirements. The application should hold the following functionalities.
2. **Customer Requirement Specification**
   1. **Customer Requirement**
      1. **Users of the system**

* Admin
* User
* Supporter
  + 1. **System functions**

| **User** | **Functions** |
| --- | --- |
| **All Users** | * Login * Change Profile * Register |
| **Admin** | * Account management * BankAccount management * Currency management * Role management * Transaction management * View monthly and annual statement of each account |
| **User** | * Own information manager * Bank Account Manager * Fund Transfer * Cheque Action * View monthly and annual statement * Notification |
| **Support** | * Support for user * View information Data |

* 1. **Common functions**

| **Login** | |
| --- | --- |
| Input | Login to system, user must be enter username and password |
| Process | Access information in database |
| Output | Result of login |
| **Change Profile** | |
| Input | Information details of user such as Name, Phone, Email … |
| Process | Maintain user’s information details |
| Output | Result of Change profile |
| **Register** | |
| Input | Register to system, user must be enter required fields |
| Process | Add new account to database. |
| Output | Result of Register |

* 1. **Admin functions**

| **Account management** | |
| --- | --- |
| Input | Information details of account such as Name, Birthday, Email, Phone, … |
| Process | Access information in database to modify, add new account to database, delete from database. |
| Output | Display or maintain Information detail of Account. |
| **Bank account management** | |
| Input | Information details of bank account such as Name, Balance,… |
| Process | Access information in database to modify, add new bank account to database, delete from database. |
| Output | Display or maintain Information detail of Bank Account. |
| **Currency management** | |
| Input | Information details of currency |
| Process | Access information in database to modify, add new currency to database, delete from database. |
| Output | Display or maintain Information detail of Bank Account. |
| **Transaction management** | |
| Input | Information details of transaction |
| Process | Access information in database to modify, add new transaction to database, delete from database. |
| Output | Display or maintain Information detail of Transaction. |
| **Role management** | |
| Input | Information details of role |
| Process | Access information in database |
| Output | Display or maintain Information detail of Role. |
| **View monthly and annual statement of each account** | |
| Input | Month or year |
| Process | Access information in database to get data. |
| Output | Export to excel |

* 1. **User functions**

| **Category Management** | |
| --- | --- |
| Input | User fills required register form |
| Process | Access information in database to modify, add new category to database, delete from database. |
| Output | Display or maintain Product |
| **Bank account management** | |
| Input | Information details of bank account such as Name, Balance,… |
| Process | Access information in database to modify, add new bank account to database, delete from database. |
| Output | Display or maintain Information detail of Bank Account. |
| **Fund Transfer** | |
| Input | User fills required information: name, account, … |
| Process | Access database to process the action fund transfer |
| Output | Display the result of transactions |
| **Cheque action** | |
| Input | Information of cheque |
| Process | Access database to modified |
| Output | Result of action |
| **View monthly and annual statement** | |
| Input | Month or year |
| Process | Access information in database to get data. |
| Output | Export to excel |

* 1. **Hardware – Software requirement**

### Minimum Hardware Requirements

* CPU: Pentium IV 2.00 GHz.
* RAM: 512 MB.
* Hard Disk: 5 GB.
* OS: Window 2000/XP/Server 2003.

### Recommended Hardware Requirements

* CPU: Dou Core 2.00 GHz or higher.
* RAM: 2 Gb or higher.
* Hard Disk: 80 GB or higher.
* OS: Window XP SP3 / Window 7

### Software Requirements

* Environment: Microsoft .NET Framework 2.0 or later
* Database Management System: Microsoft SQL Server 2005 or later
* IDE: Microsoft Visual Studio 2008 or later
* Web Server: Internet Information Services (IIS 7.0) or later

1. **Architecture and design of the system**

The Application will be made of a Web-based distributed three-tier architecture.

User Interface with HTML

And ASPX pages





C# (Code - behind) file containing

business logic





SQL Server 2014

**Web – based distributed 3 - Tier Architecture of the Project**

1. **Flow chart**
   1. **Login**













**false**



**true**



b. System Logout



1. System Login
   1. **Add / Update Data**











**false**



**true**

**false**



**true**







1. Add data

b. Update data

* 1. **Search data**









**false**



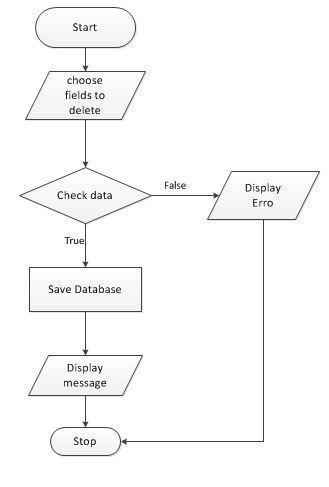
**true**







* 1. **Delete Data**

****

1. **Data flow diagram**
   1. **General Diagram**

**Admin functions**

**Graphical user interface, application

Description automatically generated**

**Supporter functions**

**Diagram

Description automatically generated**

**User functions  
  
Diagram

Description automatically generated**

* 1. **Use case**

1. Login







1. Change Profile







1. Search



1. Account Management







1. Bank Account Management





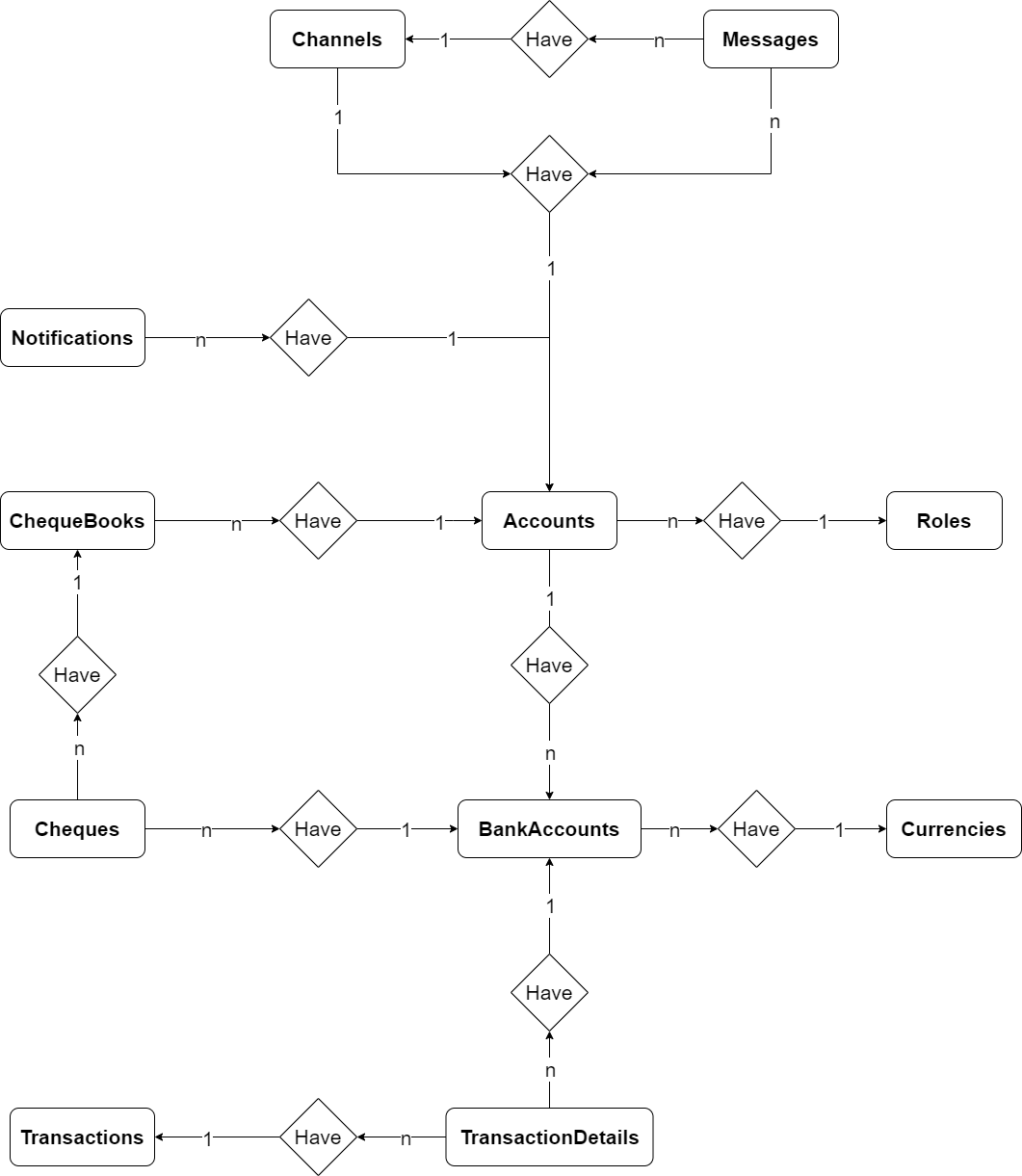
1. Role Management



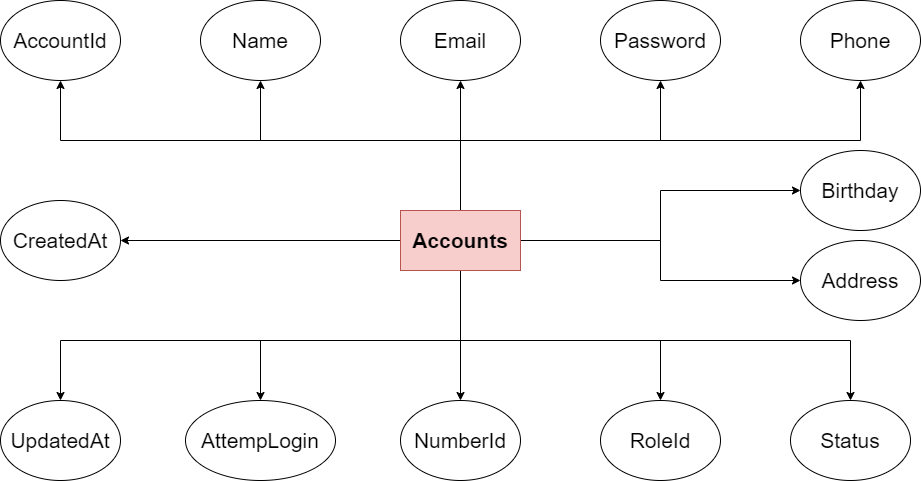


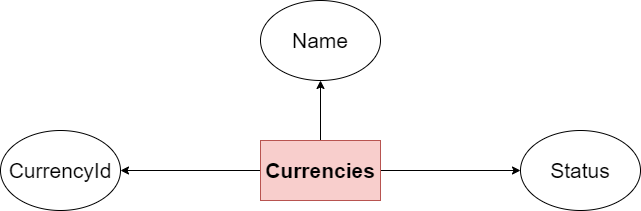


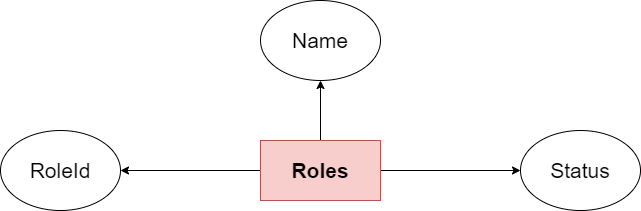
1. **Entity relationship diagram**
   1. **Entity relationship diagram**

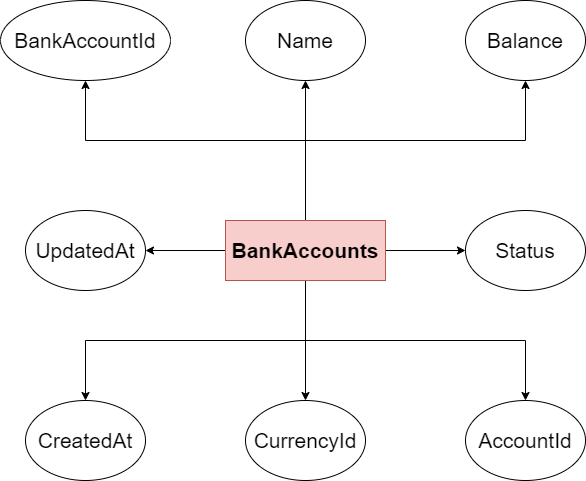
****

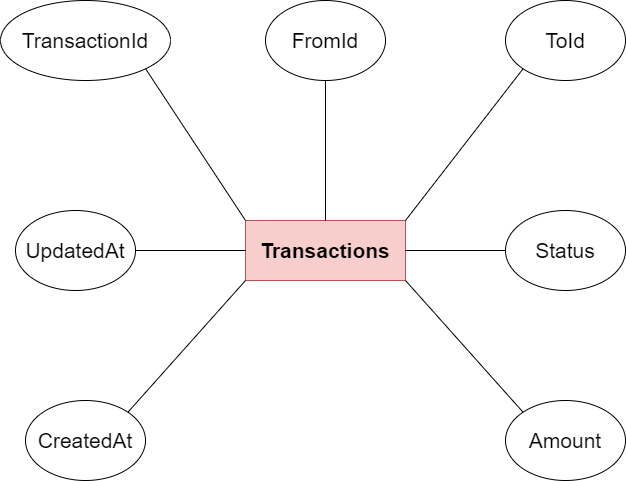
* 1. **Entity - Attributes**

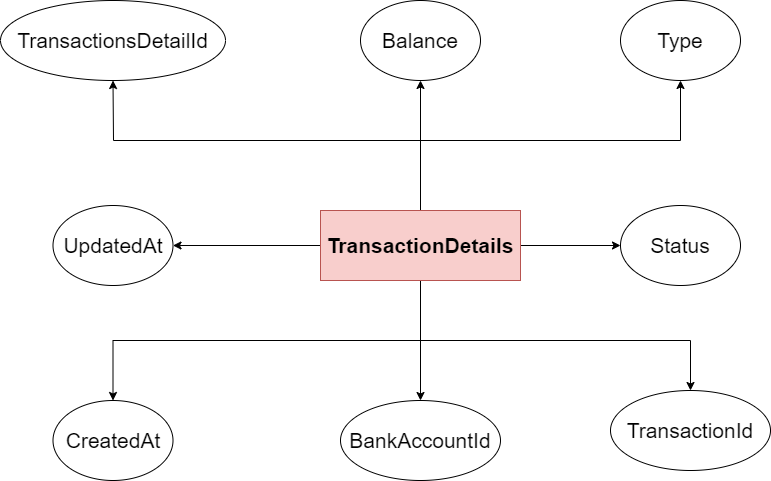


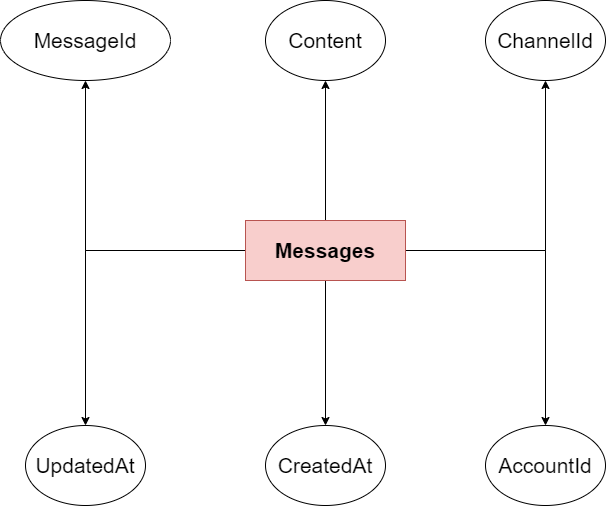


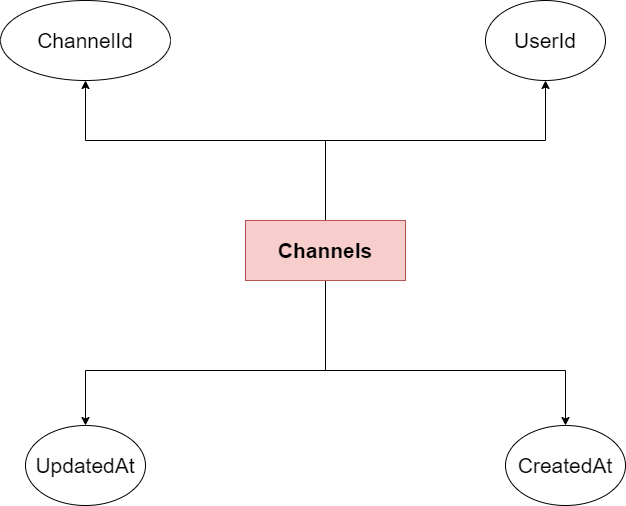
****

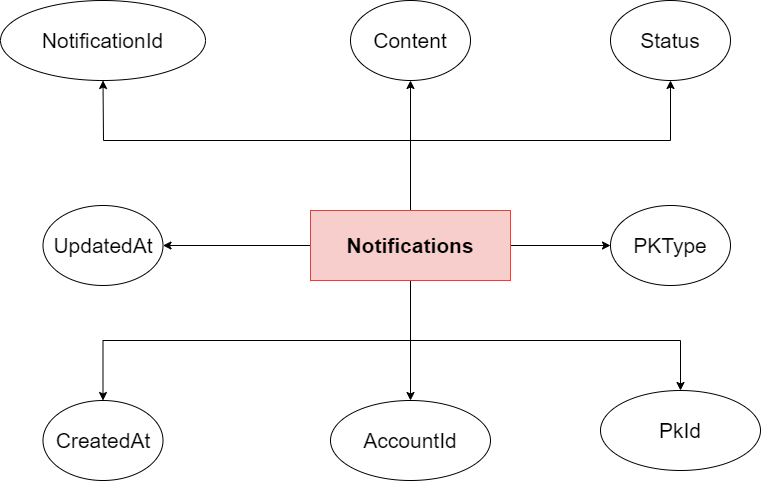
****

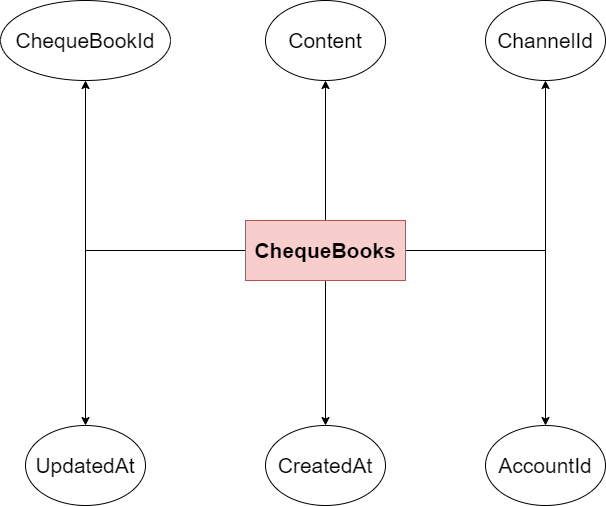
****

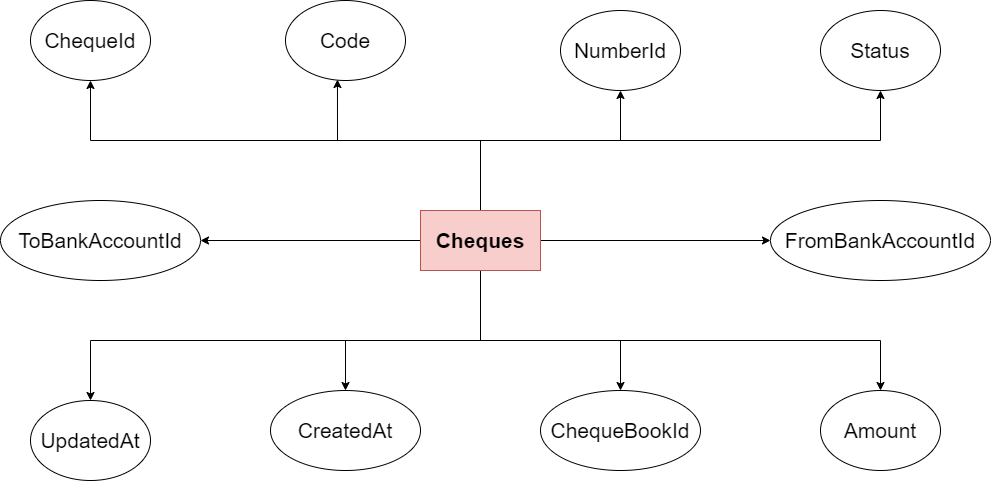
****

****

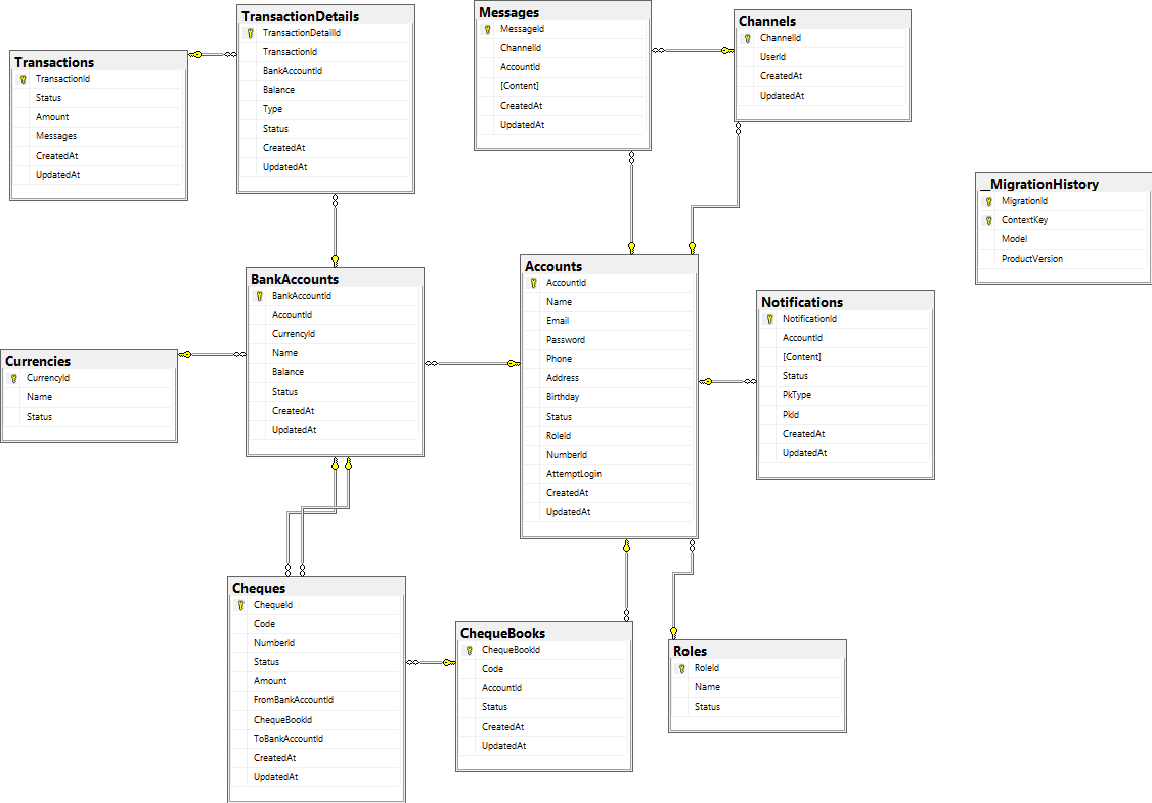
****







1. **Database Design/Structure**
   1. **Database relationship diagram**



* 1. **Table Design**

| **Table:** **Accounts** | | | | |
| --- | --- | --- | --- | --- |
| **Column** | **Data type** | **Allow Nulls** | **Key** | **Description** |
| AccountId | Int | ◻ | PK, Auto | UserId |
| Name | nvarchar(MAX) | ◻ |  | Name |
| Email | nvarchar(MAX) | ◻ |  | Email |
| Password | nvarchar(24) | ◻ |  | Password |
| Phone | nvarchar(MAX) | ◻ |  | Phone |
| Address | nvarchar(MAX) | 🗹 |  | Address |
| Birthday | datetime | 🗹 |  | Birthday |
| Status | int | 🗹 |  | Status |
| RoleId | int | ◻ | FK Roles(RoleId) | Role |
| NumberId | nvarchar(MAX) | ◻ |  | Id card |
| AttempLogin | int | ◻ |  | Login times (Lock account when 3 times) |
| CreatedAt | Datetime | 🗹 |  | Created day |
| UpdatedAt | Datetime | 🗹 |  | Updated day |

| **Table:Roles** | | | | |
| --- | --- | --- | --- | --- |
| **Column** | **Data type** | **Allow Nulls** | **Key** | **Description** |
| RoleId | int | ◻ | PK, Auto | UserId |
| Name | nvarchar(MAX) | 🗹 |  | UserName |
| Status | int | ◻ |  | Status |

| **Table:** **Currencies** | | | | |
| --- | --- | --- | --- | --- |
| **Column** | **Data type** | **Allow Nulls** | **Key** | **Description** |
| CurrencyId | Int | ◻ | PK, Auto | Currency Id |
| Name | nvarchar(MAX) | ◻ |  | Currency Name |
| Status | int | ◻ |  | Status |

| **Table:BankAccounts** | | | | |
| --- | --- | --- | --- | --- |
| **Column** | **Data type** | **Allow Nulls** | **Key** | **Description** |
| BankAccountId | Int | ◻ | PK, Auto | BankAccountId |
| Name | nvarchar(MAX) | 🗹 |  | Bank Account Name |
| Balance | float | ◻ |  | Balance |
| Status | int | ◻ |  | Status |
| AccountId | Int | ◻ | FK Accounts (AccountId) | Account |
| CurrencyId | Int | ◻ | FK Currencies (AccountId) | Currency |
| CreatedAt | Datetime | 🗹 |  | Created day |
| UpdatedAt | Datetime | 🗹 |  | Updated day |

| **Table: Transactions** | | | | |
| --- | --- | --- | --- | --- |
| **Column** | **Data type** | **Allow Nulls** | **Key** | **Description** |
| TransactionId | int | ◻ | PK, Auto | TransactionsId |
| Status | int | ◻ |  | Status |
| Amount | float | ◻ |  | Amount |
| Messages | nvarchar(MAX) |  |  |  |
| CreatedAt | Datetime | ◻ |  | Created day |
| UpdatedAt | Datetime | ◻ |  | Updated day |

| **Table: TransactionDetails** | | | | |
| --- | --- | --- | --- | --- |
| **Column** | **Data type** | **Allow Nulls** | **Key** | **Description** |
| TransactionDetailId | int | ◻ | PK, Auto | TransactionsId |
| Balance | float | ◻ |  | Status |
| Type | int | ◻ |  | Type |
| Status | int | ◻ |  | Status |
| TransactionId | int | ◻ | FK Transactions (TransactionId) | Transaction |
| BankAccountId | int | ◻ | FK BankAccounts (BankAccountId) | BankAccount |
| CreatedAt | Datetime | 🗹 |  | Created day |
| UpdatedAt | Datetime | 🗹 |  | Updated day |

| **Table: TransactionDetails** | | | | |
| --- | --- | --- | --- | --- |
| **Column** | **Data type** | **Allow Nulls** | **Key** | **Description** |
| TransactionDetailId | int | ◻ | PK, Auto | TransactionDetailId |
| Balance | float | ◻ |  | Status |
| Type | int | ◻ |  | Type |
| Status | int | ◻ |  | Status |
| TransactionId | int | ◻ | FK Transactions (TransactionId) | Transaction |
| BankAccountId | int | ◻ | FK BankAccounts (BankAccountId) | BankAccount |
| CreatedAt | Datetime | 🗹 |  | Created day |
| UpdatedAt | Datetime | 🗹 |  | Updated day |

| **Table: Messages** | | | | |
| --- | --- | --- | --- | --- |
| **Column** | **Data type** | **Allow Nulls** | **Key** | **Description** |
| MessageId | int | ◻ | PK, Auto | MessageId |
| Content | nvarchar(MAX) | ◻ |  | Content |
| ChannelId | int | ◻ | FK Channels (ChannelId) | Channel |
| AccountId | int | ◻ | FK BankAccounts (BankAccountId) | Account |
| CreatedAt | Datetime | 🗹 |  | Created day |
| UpdatedAt | Datetime | 🗹 |  | Updated day |

| **Table: Channels** | | | | |
| --- | --- | --- | --- | --- |
| **Column** | **Data type** | **Allow Nulls** | **Key** | **Description** |
| ChannelId | int | ◻ | PK, Auto | ChannelId |
| UserId | int | ◻ | FK Accounts (AccountId) | Account |
| CreatedAt | Datetime | 🗹 |  | Created day |
| UpdatedAt | Datetime | 🗹 |  | Updated day |

| **Table: Notifications** | | | | |
| --- | --- | --- | --- | --- |
| **Column** | **Data type** | **Allow Nulls** | **Key** | **Description** |
| NotificationId | int | ◻ | PK, Auto | NotificationId |
| [Content] | nvarchar(MAX) |  |  | Content |
| Status | int |  |  | Status |
| PkType | int |  |  | PkType |
| PkId | int |  |  | PkId |
| AccountId | int | ◻ | FK Accounts (AccountId) | Account |
| CreatedAt | Datetime | 🗹 |  | Created day |
| UpdatedAt | Datetime | 🗹 |  | Updated day |

| **Table: ChequeBooks** | | | | |
| --- | --- | --- | --- | --- |
| **Column** | **Data type** | **Allow Nulls** | **Key** | **Description** |
| ChequeBookId | int | ◻ | PK, Auto | ChequeBookId |
| Code | nvarchar(MAX) | 🗹 |  | Code |
| Status | int | ◻ |  | Status |
| AccountId | int | ◻ | FK Accounts (AccountId) | Account |
| CreatedAt | Datetime | 🗹 |  | Created day |
| UpdatedAt | Datetime | 🗹 |  | Updated day |

| **Table: Cheques** | | | | |
| --- | --- | --- | --- | --- |
| **Column** | **Data type** | **Allow Nulls** | **Key** | **Description** |
| ChequeId | int | ◻ | PK, Auto | ChequeId |
| Code | nvarchar(MAX) | 🗹 |  | Code |
| NumberId | nvarchar(MAX) | 🗹 |  | Id card |
| Status | int | ◻ |  | Status |
| Amount | real | ◻ |  | Amount |
| FromBankAccountId | int | ◻ | FK BankAccounts (BankAccountId) | From Bank Account |
| ToBankAccountId | int | 🗹 | FK BankAccounts (BankAccountId) | To Bank Account |
| ChequeBookId | int | ◻ | FK ChequeBooks (ChequeBookId) | Cheque Book |
| CreatedAt | Datetime | 🗹 |  | Created day |
| UpdatedAt | Datetime | 🗹 |  | Updated day |

1. **Some typical interface**

**9.1) Login***A screenshot of a computer

Description automatically generated with medium confidence*

**9.2) Admin**

Graphical user interface, application

Description automatically generated

**9.3) User**

Graphical user interface, application, website

Description automatically generated

**9.4) Profile**

Graphical user interface, application, Teams

Description automatically generated

**THE END**