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

**Document Submit**

**LibraryManagement**

HCM 06/03/2010

**Signatures Pages**

**Pounder:** Huỳnh Quốc Tuấn <Date>

Admin

Từ Quý Phượng <Date>

Mod

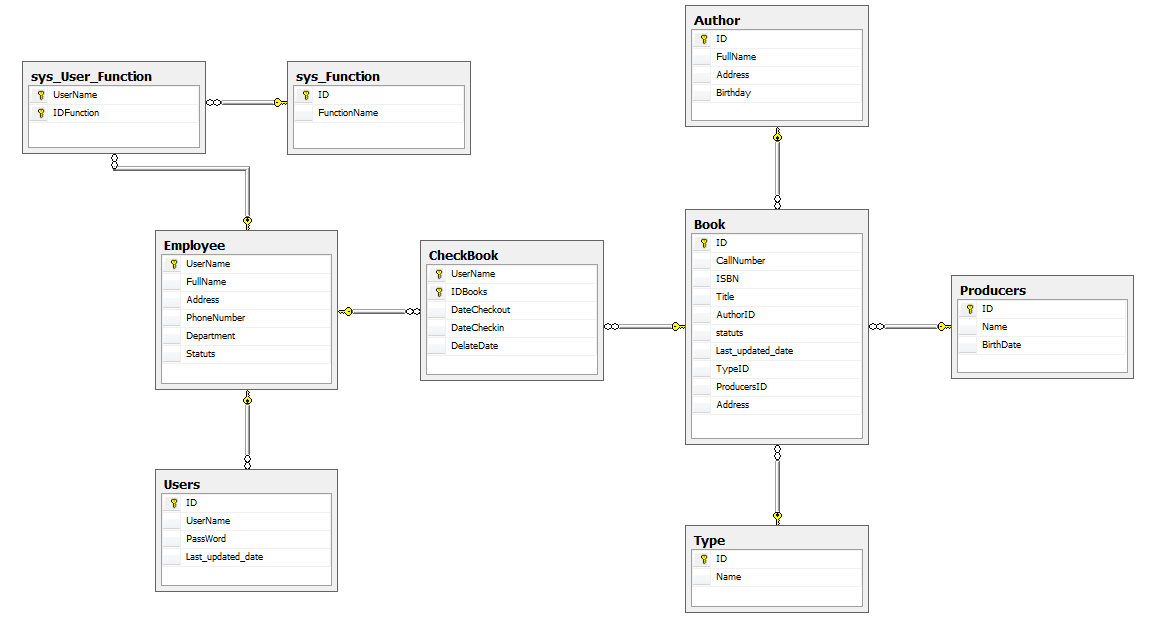
Từ Hùng <Date>

Mod

**Inspector:** Võ Đức Thiện <Date>

Customer

**DATATBASE**

****

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Name | Type | Lengh | Key |
| 1 | Author | | | |
|  | ID | int, (identity) |  | Primary Key |
|  | FullName | Nvarchar | 40 |  |
|  | Address | Nvarchar | 40 |  |
|  | Birthday | Datetime |  |  |
| 2 | Book | | | |
|  | ID | int, (identity) |  | Primary Key |
|  | CallNumber | nvarchar | 40 |  |
|  | ISBN | nvarchar | 40 |  |
|  | Title | nvarchar | 40 |  |
|  | AuthorID | int |  | Foreign key |
|  | Statuts | bit |  |  |
|  | Last\_updated\_date | datetime |  |  |
|  | TypeID | int |  | Foreign key |
|  | ProducersID | int |  | Foreign key |
|  | Address | nvarchar | 40 |  |
| 3 | Employee | | | |
|  | UserName | nvarchar | 40 | Primary key |
|  | FullName | nvarchar | 40 |  |
|  | Address | nvarchar | 40 |  |
|  | PhoneNumber | nvarchar | 40 |  |
|  | Department | nvarchar | 40 |  |
|  | Statuts | bit |  |  |
| 4 | CheckBook | | | |
|  | UserName | nvarchar | 40 | Primary key |
|  | IDBooks | int |  | Primary key |
|  | DateCheckout | datetime |  |  |
|  | DateCheckin | datetime |  |  |
|  | DelateDate | nvarchar | 40 |  |
| 5 | Producers | | | |
|  | ID | int, (identity) |  | Primary key |
|  | Name | nvarchar | 40 |  |
|  | Address | nvarchar | 40 |  |
| 6 | Sys\_Function | | | |
|  | ID | int, (identity) |  | Primary key |
|  | FunctionName | nvarchar | 40 |  |
| 7 | Sys\_User\_Function | | | |
|  | UserName | nvarchar | 40 | Primary key |
|  | IDFunction | int |  | Primary key |
| 8 | Type | | | |
|  | ID | int, (identity) |  | Primary key |
|  | Name | nvarchar | 40 |  |
| 9 | Users | | | |
|  | ID | int, (identity) |  | Primary key |
|  | UserName | nvarchar | 40 | Foreign key |
|  | PassWord | nvarchar | 40 |  |
|  | Last\_updated\_date | datetime |  |  |

**Use Case Diagram**

****

**Flow**



When libraryManagement login success :

1. Can view book :
   1. Update books in system.
   2. Check in books to outside in system.
2. Can view member :
   1. Add new member in system.
   2. Update member in system.
   3. Lock member in system.
3. Can view repost

Else system retrun the login pages.

****When member login success :

1. Can view books and search books.
2. View informational this member.
3. Check in books and check out books.



**CODE CREATE DATABASE**

1. Create database : create database librarymanagement
2. Use this database : use librarymanagement
3. Creat table Employee :

CREATE TABLE [Employee] (

[UserName] NVARCHAR(40) NOT NULL,

[FullName] NVARCHAR(40),

[Address] NVARCHAR(40),

[PhoneNumber] NVARCHAR(40),

[Department] NVARCHAR(40),

[Statuts] BIT NOT NULL,

CONSTRAINT [PK\_Employee] PRIMARY KEY ([UserName])

)

1. Create table books :

CREATE TABLE [Book] (

[ID] INTEGER IDENTITY(0,1) NOT NULL,

[CallNumber] NVARCHAR(40) NOT NULL,

[ISBN] NVARCHAR(40) NOT NULL,

[Title] NVARCHAR(40),

[AuthorID] INTEGER NOT NULL,

[statuts] BIT NOT NULL,

[Last\_updated\_date] DATETIME,

[TypeID] INTEGER NOT NULL,

[ProducersID] INTEGER NOT NULL,

[Address] NVARCHAR(40),

CONSTRAINT [PK\_Book] PRIMARY KEY ([ID])

)

reate table check book :

CREATE TABLE [CheckBook] (

[UserName] NVARCHAR(40) NOT NULL,

[IDBooks] INTEGER NOT NULL,

[DateCheckout] DATETIME,

[DateCheckin] DATETIME,

[DelateDate] NVARCHAR(40),

CONSTRAINT [PK\_CheckBook] PRIMARY KEY ([UserName], [IDBooks])

)

1. Create table users :

CREATE TABLE [Users] (

[ID] INTEGER IDENTITY(0,1) NOT NULL,

[UserName] NVARCHAR(40) NOT NULL,

[PassWord] NVARCHAR(40),

[Last\_updated\_date] DATETIME,

CONSTRAINT [PK\_Users] PRIMARY KEY ([ID])

)

1. Create table sys\_User\_Function :

CREATE TABLE [sys\_User\_Function] (

[UserName] NVARCHAR(40) NOT NULL,

[IDFunction] INTEGER NOT NULL,

CONSTRAINT [PK\_sys\_User\_Function] PRIMARY KEY ([UserName], [IDFunction])

)

1. Create table Author :

CREATE TABLE [Author] (

[ID] INTEGER IDENTITY(0,1) NOT NULL,

[FullName] NVARCHAR(40),

[Address] NVARCHAR(40),

[Birthday] DATETIME,

CONSTRAINT [PK\_Author] PRIMARY KEY ([ID])

)

1. Create table Type :

CREATE TABLE [Type] (

[ID] INTEGER IDENTITY(0,1) NOT NULL,

[Name] NVARCHAR(40),

CONSTRAINT [PK\_Type] PRIMARY KEY ([ID])

)

1. **Create table Producers:**

CREATE TABLE [Producers] (

[ID] INTEGER IDENTITY(0,1) NOT NULL,

[Name] NVARCHAR(40),

[Address] NVARCHAR(40),

CONSTRAINT [PK\_Producers] PRIMARY KEY ([ID])

)

1. **Create foreign for all table :**

ALTER TABLE [Book] ADD CONSTRAINT [Author\_Book]

FOREIGN KEY ([AuthorID]) REFERENCES [Author] ([ID])

GO

ALTER TABLE [Book] ADD CONSTRAINT [Type\_Book]

FOREIGN KEY ([TypeID]) REFERENCES [Type] ([ID])

GO

ALTER TABLE [Book] ADD CONSTRAINT [Producers\_Book]

FOREIGN KEY ([ProducersID]) REFERENCES [Producers] ([ID])

GO

ALTER TABLE [CheckBook] ADD CONSTRAINT [Book\_CheckBook]

FOREIGN KEY ([IDBooks]) REFERENCES [Book] ([ID])

GO

ALTER TABLE [CheckBook] ADD CONSTRAINT [Employee\_CheckBook]

FOREIGN KEY ([UserName]) REFERENCES [Employee] ([UserName])

GO

ALTER TABLE [Users] ADD CONSTRAINT [Employee\_Users]

FOREIGN KEY ([UserName]) REFERENCES [Employee] ([UserName])

GO

ALTER TABLE [sys\_User\_Function] ADD CONSTRAINT [Employee\_sys\_User\_Function]

FOREIGN KEY ([UserName]) REFERENCES [Employee] ([UserName])

GO

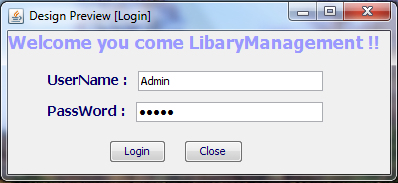
ALTER TABLE [sys\_User\_Function] ADD CONSTRAINT [sys\_Function\_sys\_User\_Function]

FOREIGN KEY ([IDFunction]) REFERENCES [sys\_Function] ([ID])

GO

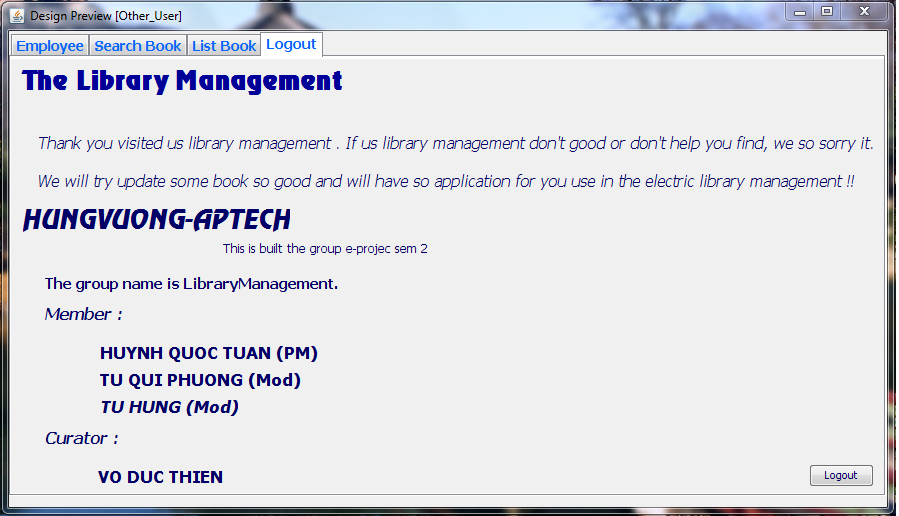
**Playout Design**

**Login :**

****

You can input username and password , click buttom login and login in us system.

**Logout :**

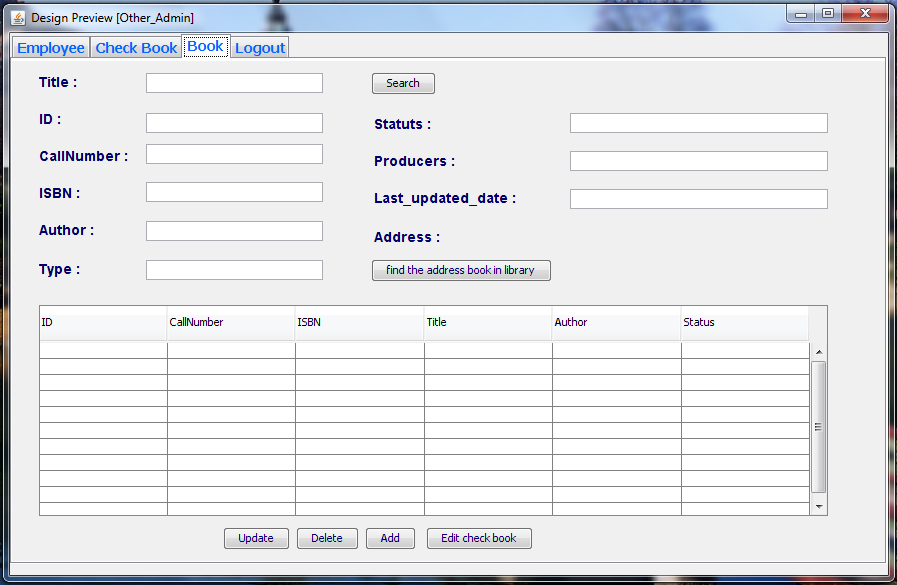
****

When you used, you can clik task Logout and buttom logout.

**Admin login and use**

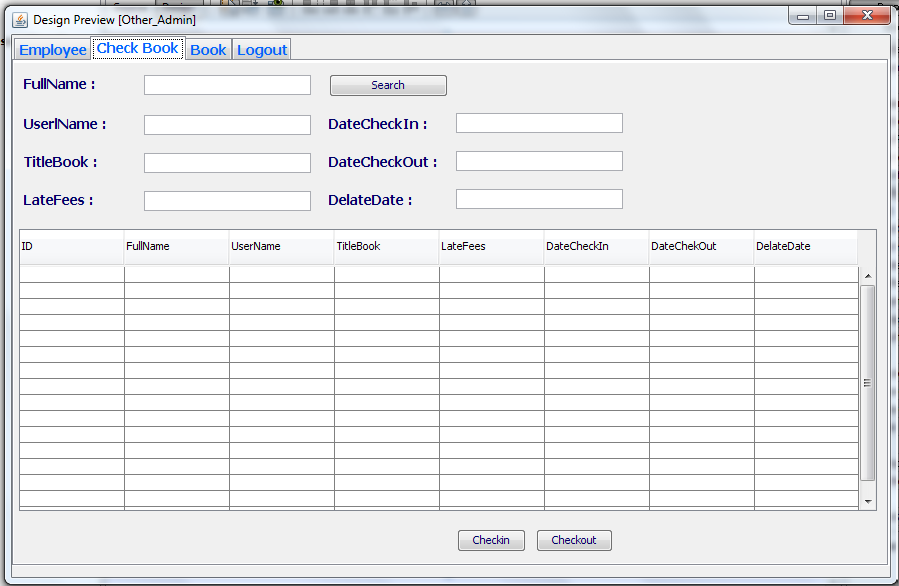
**Task book :**

* you can insert new book.
* You can Search book in system library.

****

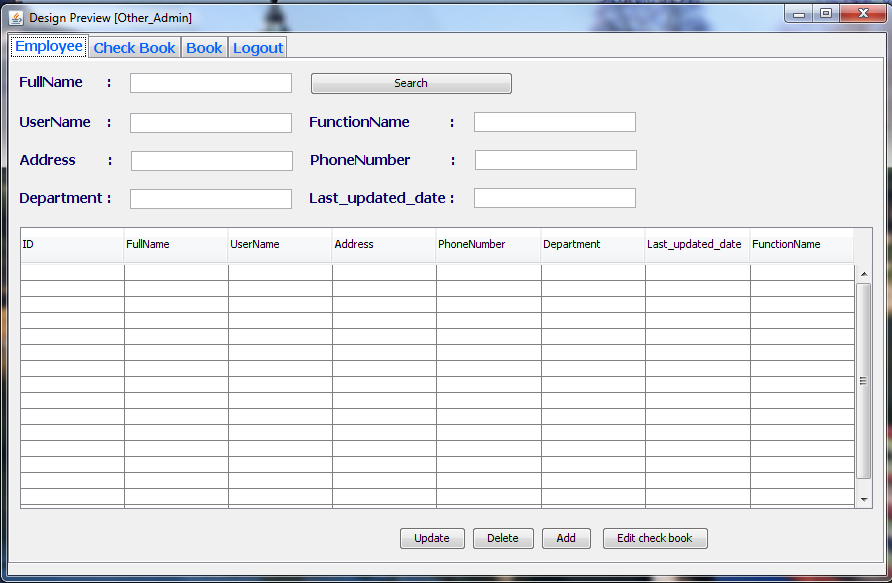
**Task Check Book :**

* you can Checkin or checkout book for user .
* You can Search The use shecked book in system library.

****

**Task Employee :**

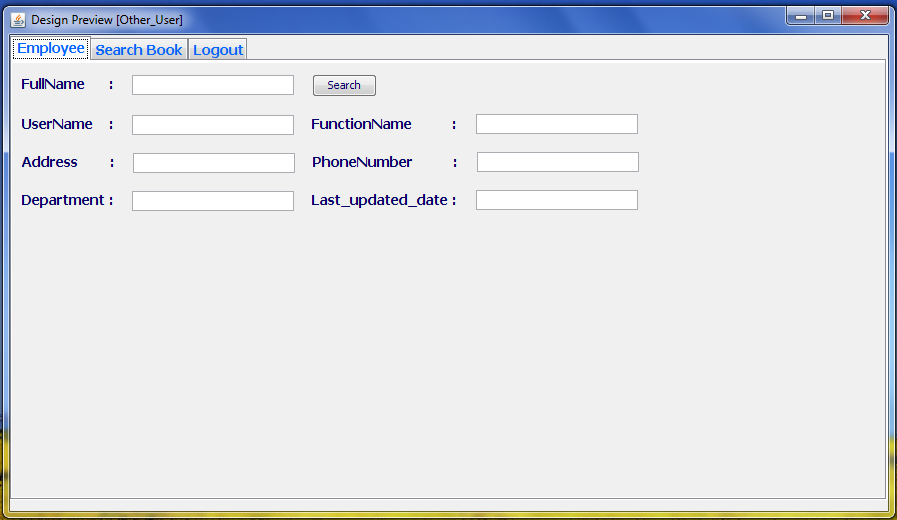
* you insert new Employee .
* You can Search The use shecked book in system library.

****

**Employee login and use**

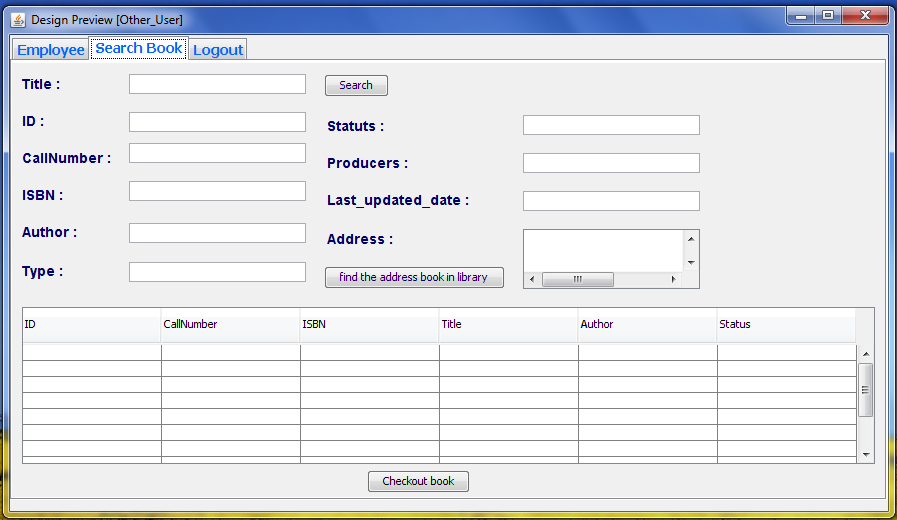
**Task Employee :**

* **You will read your personal information.**

****

**Task search book :**

* **You can find book in laybrary**

****