

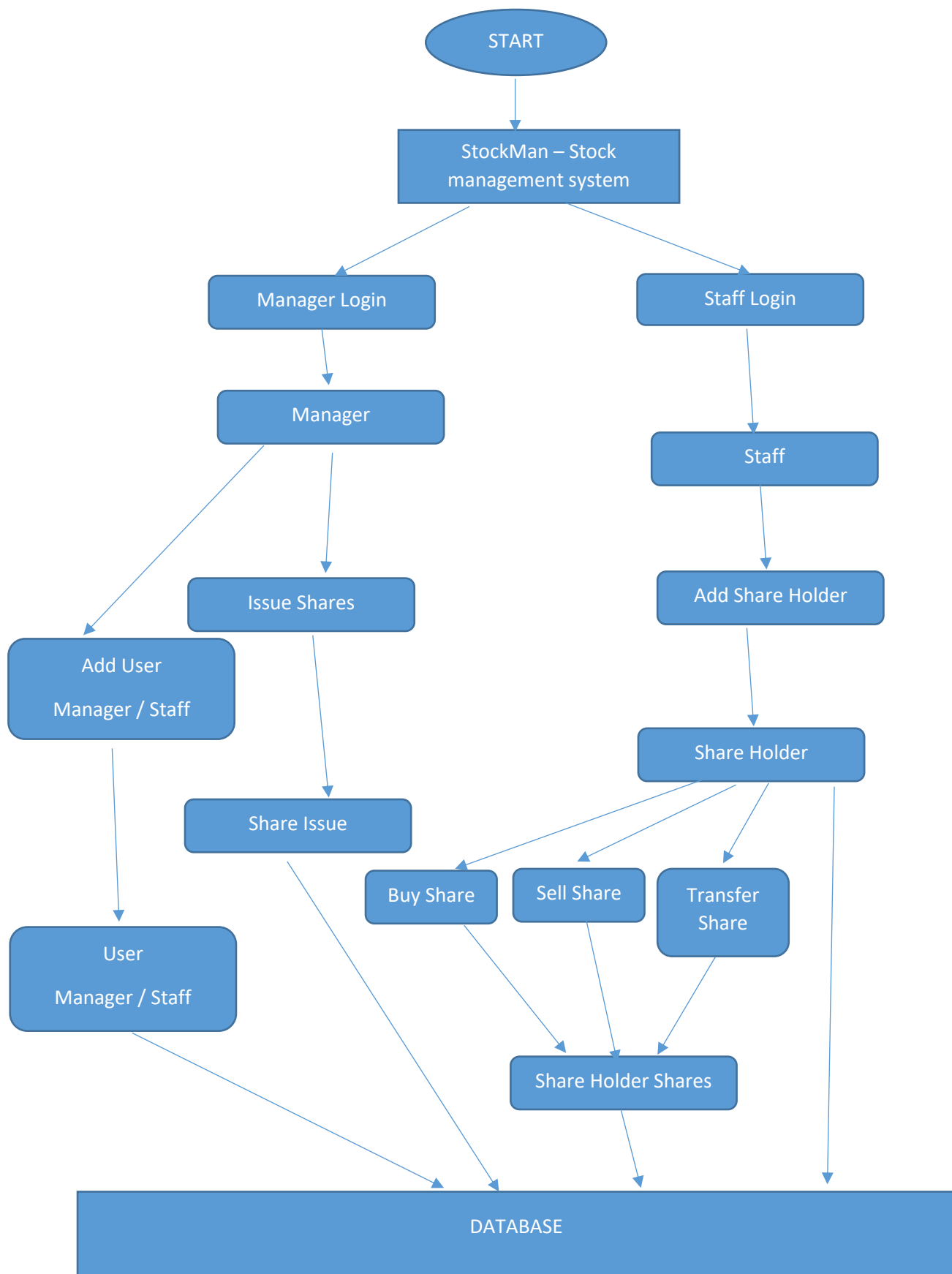
# **SYSTEM DESIGN**

## **INDRODUCTION TO SYSTEM DESIGN**

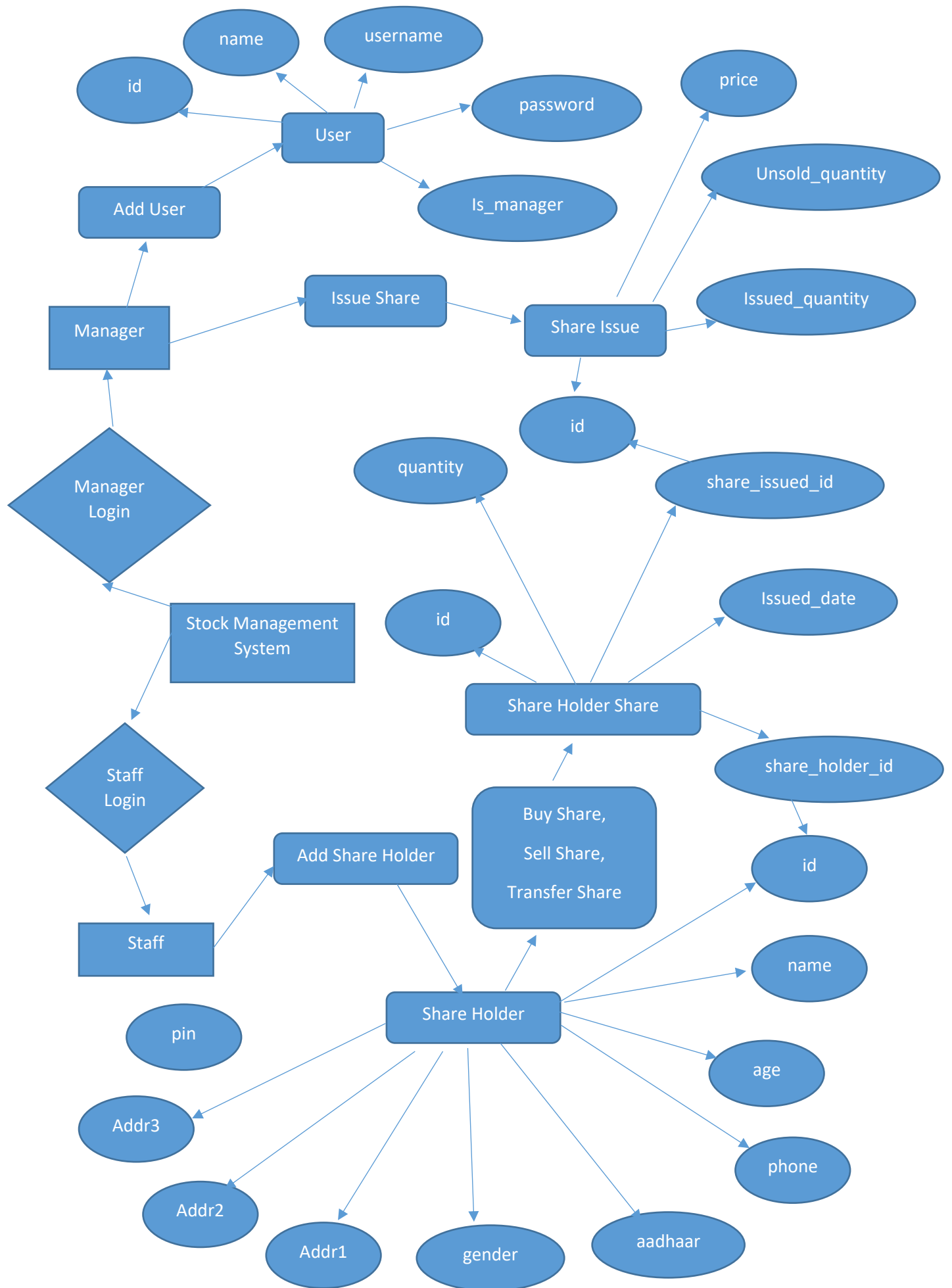
The design phase is the life cycle phase in which the detailed design of the selected system in the study phase is accomplished. In the design phase, the technical specifications are prepared for the performed of all allocated tasks. It also includes the construction of programs and programs testing. In the design phase, the first step is to determine the output is to produced and in what format.

Systems design is the process of defining elements of a system like modules, architecture, components and their interfaces and data for a system based on the specified requirements. It is the process of defining, developing and designing systems which satisfies the specific needs and requirements of a business or organization.

# **DATA FLOW DIAGRAM**



# **ER DIAGRAM**



**CODE**

## Add Manager Page

```
Public Class AddManager
    Private Sub CreateManagerButton_Click(sender As Object, e As EventArgs)
Handles CreateManagerButton.Click
        Dim Name = NameInput.Text
        Dim Username = UsernameInput.Text
        Dim Password = PasswordInput.Text

        ManagerModule.AddManager(Name, Username, Password)
    End Sub
End Class
```

## Manager Login Page

```
Public Class ManagerLogin
    Private Sub CreateManagerButton_Click(sender As Object, e As EventArgs)
Handles LoginButton.Click
        Dim Username = UsernameInput.Text
        Dim Password = PasswordInput.Text

        ManagerModule.ManagerLogin(Username, Password)
    End Sub

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        Form1.LoadLoginPage()
    End Sub
End Class
```

## Manager Page

```
Public Class Manager
    Dim EquityPage As ManagerEquity
    Dim IssueSharePage As ManagerIssueShare

    Dim UsersPage As ManagerUsers
    Dim AddUserPage As ManagerAddUser
    Private Sub Manager_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        LoadEquityPage()
    End Sub

    Private Sub LoadDashboardPage()

    End Sub

    Public Sub LoadEquityPage()
        EquityPage = New ManagerEquity

        ManagerAppPanel.Controls.Clear()
        ManagerAppPanel.Controls.Add(EquityPage)
    End Sub

    Public Sub LoadIssueSharePage()
        IssueSharePage = New ManagerIssueShare

        ManagerAppPanel.Controls.Clear()
        ManagerAppPanel.Controls.Add(IssueSharePage)
    End Sub
```



```

Public Sub LoadUsersPage()
    UsersPage = New ManagerUsers

    ManagerAppPanel.Controls.Clear()
    ManagerAppPanel.Controls.Add(UsersPage)
End Sub

Public Sub LoadAddUserPage()
    AddUserPage = New ManagerAddUser

    ManagerAppPanel.Controls.Clear()
    ManagerAppPanel.Controls.Add(AddUserPage)
End Sub

Private Sub Label3_Click(sender As Object, e As EventArgs) Handles
Label3.Click
    Form1.ManagerPage.LoadEquityPage()
End Sub

Private Sub Label4_Click(sender As Object, e As EventArgs) Handles
Label4.Click
    Form1.ManagerPage.LoadUsersPage()
End Sub

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
    Form1.LoggedInUserName = ""
    Form1.LoadLoginPage()
End Sub
End Class

```

## Manager Users Page

```

Public Class ManagerUsers
    Private Sub ManagerUsers_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        UserModule.LoadUserData()
        Dim CurrentItem = 0
        For Each User As UserModel In UserModule.Users
            UsersListView.Items.Add(User.Id)
            UsersListView.Items(CurrentItem).SubItems.Add(User.Name)
            CurrentItem += 1
        Next

        CurrentItem = 0
        For Each User As UserModel In UserModule.Users
            UsersListView.Items(CurrentItem).SubItems.Add(User.Username)
            CurrentItem += 1
        Next

        CurrentItem = 0
        For Each User As UserModel In UserModule.Users
            If User.IsManager Then
                UsersListView.Items(CurrentItem).SubItems.Add("Manager")
            Else
                UsersListView.Items(CurrentItem).SubItems.Add("Staff")
            End If
            CurrentItem += 1
        Next
    End Sub
End Class

```

```

    Private Sub GoToAddUserPageButton_Click(sender As Object, e As EventArgs)
Handles GoToAddUserPageButton.Click
    Form1.ManagerPage.LoadAddUserPage()
End Sub
End Class

```

## Manager Add User Page

```

Public Class ManagerAddUser
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        Form1.ManagerPage.LoadUsersPage()
    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
        Dim Name = NameInput.Text
        Dim Username = UsernameInput.Text
        Dim Password = PasswordInput.Text

        Dim IsManager As Boolean

        If (IsManagerInput.Text = "Manager") Then
            IsManager = True
        Else
            IsManager = False
        End If

        UserModule.AddUser(Name, Username, Password, IsManager)
    End Sub
End Class

```

## Manager Equity Page

```

Public Class ManagerEquity
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        Form1.ManagerPage.LoadIssueSharePage()
    End Sub

    Private Sub ManagerEquity_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        EquityModule.LoadEquityData()
        Dim CurrentItem = 0
        For Each Issue As ShareIssueModel In EquityModule.IssuedShareList
            ShareIssueList.Items.Add(Issue.IssuedDate.ToString("dd-MM-yyyy"))
            ShareIssueList.Items(CurrentItem).SubItems.Add(Issue.IssuedQty)
            CurrentItem += 1
        Next

        CurrentItem = 0
        For Each Issue As ShareIssueModel In EquityModule.IssuedShareList
            ShareIssueList.Items(CurrentItem).SubItems.Add(Issue.Price)
            CurrentItem += 1
        Next

        CurrentItem = 0
        For Each Issue As ShareIssueModel In EquityModule.IssuedShareList
            ShareIssueList.Items(CurrentItem).SubItems.Add(Issue.UnsoldQty)

```

```

        CurrentItem += 1
    Next

    CurrentItem = 0
    For Each Issue As ShareIssueModel In EquityModule.IssuedShareList
        Dim Sum = (Issue.IssuedQty - Issue.UnsoldQty) * Issue.Price
        ShareIssueList.Items(CurrentItem).SubItems.Add(Sum)
        CurrentItem += 1
    Next
End Sub
End Class

```

## **Manager Issue Share Page**

```

Public Class ManagerIssueShare
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        Form1.ManagerPage.LoadEquityPage()
    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
        Dim Qty = QtyInput.Value
        Dim Price = PriceInput.Value

        If (Qty < 1) Then
            MessageBox.Show("Invalid Quantity")
        ElseIf (Price < 1) Then
            MessageBox.Show("Invalid Price")
        Else
            EquityModule.IssueShares(Qty, Price)
        End If
    End Sub
End Class

```

## **Staff Login Page – UserLogin.vb**

```

Public Class LoginForm
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        Form1.LoadManagerLoginPage()
    End Sub

    Private Sub LoginButton_Click(sender As Object, e As EventArgs) Handles
LoginButton.Click
        Dim Username = UsernameInput.Text
        Dim Password = PasswordInput.Text

        StaffModule.StaffLogin(Username, Password)
    End Sub
End Class

```

## **Staff Page**

```

Public Class Staff

    Public SelectedShareHolderId As Integer
    Dim ShareHoldersPage As StaffShareHolders
    Dim AddShareHolderPage As StaffAddShareHolder

```

```

Dim ShareHolderAccountPage As StaffShareHolderAc
Dim BuySharePage As StaffBuyShare

Private Sub Staff_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    LoadShareHoldersPage()
End Sub
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
    Form1.LoggedInUserName = ""
    Form1.LoadLoginPage()
End Sub

Public Sub LoadShareHoldersPage()
    ShareHoldersPage = New StaffShareHolders

    StaffAppPanel.Controls.Clear()
    StaffAppPanel.Controls.Add(ShareHoldersPage)
End Sub

Public Sub LoadAddShareHolderPage()
    AddShareHolderPage = New StaffAddShareHolder

    StaffAppPanel.Controls.Clear()
    StaffAppPanel.Controls.Add(AddShareHolderPage)
End Sub

Public Sub LoadShareHolderAccountPage()
    ShareHolderAccountPage = New StaffShareHolderAc

    StaffAppPanel.Controls.Clear()
    StaffAppPanel.Controls.Add(ShareHolderAccountPage)
End Sub

Public Sub LoadBuySharePage()
    BuySharePage = New StaffBuyShare

    StaffAppPanel.Controls.Clear()
    StaffAppPanel.Controls.Add(BuySharePage)
End Sub

Private Sub Label3_Click(sender As Object, e As EventArgs) Handles
Label3.Click
    Form1.StaffPage.LoadShareHoldersPage()
End Sub
End Class

```

## **Staff Share Holders Page**

```

Public Class StaffShareHolders
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        Form1.StaffPage.LoadAddShareHolderPage()
    End Sub

    Private Sub StaffShareHolders_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        ShareHolderModule.LoadShareHoldersData()

        Dim CurrentItem = 0
        For Each SH As ShareHolderModel In ShareHolderModule.ShareHolders
            ShareHoldersList.Items.Add(SH.Name)
            ShareHoldersList.Items(CurrentItem).SubItems.Add(SH.Id)
            CurrentItem += 1
        End For
    End Sub
End Class

```

```

        Next

        CurrentItem = 0
        For Each SH As ShareHolderModel In ShareHolderModule.ShareHolders
            ShareHoldersList.Items(CurrentItem).SubItems.Add(SH.Phone)
            CurrentItem += 1
        Next
    End Sub

    Private Sub ShareHoldersList_SelectedIndexChanged(sender As Object, e As
EventArgs) Handles ShareHoldersList.SelectedIndexChanged
        If (ShareHoldersList.SelectedItems.Count > 0) Then
            Form1.StaffPage.SelectedShareHolderId =
ShareHoldersList.SelectedItems(0).SubItems(1).Text

ShareHolderModule.GetShareHolderById(Form1.StaffPage.SelectedShareHolderId)

            If (ShareHolderExists) Then
                Form1.StaffPage.LoadShareHolderAccountPage()
            Else
                MessageBox.Show("Invalid Id")
            End If
        End If
    End Sub
End Class

```

## **Staff Add Share Holder Page**

```

Public Class StaffAddShareHolder
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        Form1.StaffPage.LoadShareHoldersPage()
    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
        Validation.IsValidName(NameInput.Text)
        If (Validation.IsNameInvalid) Then
            MessageBox.Show(Validation.ErrMessage)

            Exit Sub
        End If

        Validation.IsValidPhoneNumber(PhoneInput.Text)
        If (Validation.IsPhoneNumberInvalid) Then
            MessageBox.Show(Validation.ErrMessage)

            Exit Sub
        End If

        Validation.IsValidAadhaarNumber(AadhaarInput.Text)
        If (Validation.IsAadhaarNumberInvalid) Then
            MessageBox.Show(Validation.ErrMessage)

            Exit Sub
        End If

        Validation.IsValidPinNumber(PINInput.Text)
        If (Validation.IsPinNumberInvalid) Then
            MessageBox.Show(Validation.ErrMessage)

```

```

        Exit Sub
    End If

    AddNewShareHolder()
End Sub

Private Sub AddNewShareHolder()
    Dim ShareHolder = New ShareHolderModel With {
        .Name = NameInput.Text,
        .Age = AgeInput.Value,
        .Gender = GenderInput.Text,
        .Phone = PhoneInput.Text,
        .Aadhaar = AadhaarInput.Text,
        .Addr1 = Addr1Input.Text,
        .Addr2 = Addr2Input.Text,
        .Addr3 = Addr3Input.Text,
        .PIN = PINInput.Text
    }

    ShareHolderModule.AddNewShareHolder(ShareHolder)
End Sub
End Class

```

## **Staff Share Holder Ac Page**

```

Public Class StaffShareHolderAc
    Private Sub StaffShareHolderAc_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        ShareHolderModule.LoadShareHolderShares(Form1.StaffPage.SelectedShareHolderId)

        Dim CurrentCount = 0
        Dim Total = 0
        For Each Share As ShareHolderShareModel In
            ShareHolderModule.ShareHolderShares
                ShareHolderSharesList.Items.Add(Share.Qty)

            ShareHolderSharesList.Items(CurrentCount).SubItems.Add(Share.ShareIssuedPrice)
            Total = Total + (Share.Qty * Share.ShareIssuedPrice)
            CurrentCount += 1
        Next
        TotalLabel.Text = Total
    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
        Form1.StaffPage.LoadBuySharePage()
    End Sub

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        Form1.StaffPage.LoadShareHoldersPage()
    End Sub
End Class

```

## **Edit Share Holder Page**

```

Public Class StaffEditShareHolder

```

```
Private Sub StaffEditShareHolder_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
ShareHolderModule.GetShareHolderById(Form1.StaffPage.SelectedShareHolderId)
```

```
    NameInput.Text = ShareHolderModule.SelectedShareHolder.Name  
    AgeInput.Value = ShareHolderModule.SelectedShareHolder.Age  
    GenderInput.Text = ShareHolderModule.SelectedShareHolder.Gender  
    PhoneInput.Text = ShareHolderModule.SelectedShareHolder.Phone  
    AadhaarInput.Text = ShareHolderModule.SelectedShareHolder.Aadhaar  
    Addr1Input.Text = ShareHolderModule.SelectedShareHolder.Addr1  
    Addr2Input.Text = ShareHolderModule.SelectedShareHolder.Addr2  
    Addr3Input.Text = ShareHolderModule.SelectedShareHolder.Addr3  
    PINInput.Text = ShareHolderModule.SelectedShareHolder.PIN  
End Sub
```

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
```

```
    Validation.IsValidName(NameInput.Text)  
    If (Validation.IsNameInvalid) Then  
        MessageBox.Show(Validation.ErrorMessage)
```

```
    Exit Sub  
End If
```

```
    Validation.IsValidPhoneNumber(PhoneInput.Text)  
    If (Validation.IsPhoneNumberInvalid) Then  
        MessageBox.Show(Validation.ErrorMessage)
```

```
    Exit Sub  
End If
```

```
    Validation.IsValidAadhaarNumber(AadhaarInput.Text)  
    If (Validation.IsAadhaarNumberInvalid) Then  
        MessageBox.Show(Validation.ErrorMessage)
```

```
    Exit Sub  
End If
```

```
    Validation.IsValidPinNumber(PINInput.Text)  
    If (Validation.IsPinNumberInvalid) Then  
        MessageBox.Show(Validation.ErrorMessage)
```

```
    Exit Sub  
End If
```

```
    EditShareHolder()  
End Sub
```

```
Private Sub EditShareHolder()  
    Dim ShareHolder = New ShareHolderModel With {  
        .Id = Form1.StaffPage.SelectedShareHolderId,  
        .Name = NameInput.Text,  
        .Age = AgeInput.Value,  
        .Gender = GenderInput.Text,  
        .Phone = PhoneInput.Text,  
        .Aadhaar = AadhaarInput.Text,  
        .Addr1 = Addr1Input.Text,  
        .Addr2 = Addr2Input.Text,  
        .Addr3 = Addr3Input.Text,  
        .PIN = PINInput.Text  
    }  
}
```

```

        ShareHolderModule.UpdateShareHolder(ShareHolder)
    End Sub

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        Form1.StaffPage.LoadShareHolderAccountPage()
    End Sub
End Class

```

## **Staff Buy Share Page**

```

Public Class StaffBuyShare
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        Form1.StaffPage.LoadShareHolderAccountPage()
    End Sub

    Private Sub StaffBuyShare_Load(sender As Object, e As EventArgs) Handles
MyBase.Load
        EquityModule.LoadEquityData()

        For Each Share As ShareIssueModel In EquityModule.IssuedShareList
            ShareInput.Items.Add(Share.IssuedDate.ToString("dd-MM-yy") + " -
Rs." + Share.Price.ToString() + " - " + Share.UnsoldQty.ToString() + " nos.")
        Next
    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
        If (ShareInput.SelectedIndex < 0) Then
            MessageBox.Show("Please select a share")
        ElseIf QtyInput.Value < 1 Then
            MessageBox.Show("Please enter a valid quantity")
        ElseIf QtyInput.Value >
EquityModule.IssuedShareList.ElementAt(ShareInput.SelectedIndex).UnsoldQty Then
            MessageBox.Show("Only " +
EquityModule.IssuedShareList.ElementAt(ShareInput.SelectedIndex).UnsoldQty.ToStri
ng() + " shares available in this category")
        Else
            BuyShares()
        End If
    End Sub

    Private Sub BuyShares()
        Dim ShareIssueId =
EquityModule.IssuedShareList.ElementAt(ShareInput.SelectedIndex).Id
        Dim ShareHolderId = Form1.StaffPage.SelectedShareHolderId
        Dim Quantity = QtyInput.Value

        ShareHolderModule.BuyShares(ShareIssueId, ShareHolderId, Quantity)
    End Sub
End Class

```

## **Validation**

```

Module Validation
    Public IsNameInvalid = False
    Public IsPhoneNumberInvalid = False
    Public IsAadhaarNumberInvalid = False

```



```

Public IsPinNumberInvalid = False
Public ErrMessage As String

Public Sub IsValidName(Name As String)
    IsNameInvalid = False
    If Len(Name) = 0 Then
        IsNameInvalid = True
        ErrMessage = "Please enter a name"

        Exit Sub
    End If

    If Text.RegularExpressions.Regex.Match(Name, "\d").Success Then
        IsNameInvalid = True
        ErrMessage = "Name can only contain alphabets"

        Exit Sub
    End If
End Sub

Public Sub IsValidPhoneNumber(Phone As String)
    IsPhoneNumberInvalid = False

    If Len(Phone) = 0 Then
        IsPhoneNumberInvalid = True
        ErrMessage = "Please enter a phone number"

        Exit Sub
    End If

    If IsNumeric(Phone) = False Then
        IsPhoneNumberInvalid = True
        ErrMessage = "Phone number can only contain numeric characters"

        Exit Sub
    End If

    If (Len(Phone) <> 10) Then
        IsPhoneNumberInvalid = True
        ErrMessage = "Please enter a valid 10 digit phone number"

        Exit Sub
    End If
End Sub

Public Sub IsValidAadhaarNumber(Aadhaar As String)
    IsAadhaarNumberInvalid = False

    If Len(Aadhaar) = 0 Then
        IsAadhaarNumberInvalid = True
        ErrMessage = "Please enter aadhaar number"

        Exit Sub
    End If

    If IsNumeric(Aadhaar) = False Then
        IsAadhaarNumberInvalid = True
        ErrMessage = "Aadhaar number can only contain numeric characters"
        Exit Sub
    End If

    If (Len(Aadhaar) <> 12) Then
        IsAadhaarNumberInvalid = True
    End If

```

```

        ErrorMessage = "Please enter a valid 12 digit aadhaar number"

        Exit Sub
    End If

End Sub

Public Sub IsValidPinNumber(Pin As String)
    IsPinNumberInvalid = False

    If Len(Pin) = 0 Then
        IsPinNumberInvalid = True
        ErrorMessage = "Please enter a pin number"

        Exit Sub
    End If

    If IsNumeric(Pin) = False Then
        IsPinNumberInvalid = True
        ErrorMessage = "Pin number can only contain numeric characters"
        Exit Sub
    End If

    If (Len(Pin) <> 6) Then
        IsPinNumberInvalid = True
        ErrorMessage = "Please enter a valid 6 char pin number"

        Exit Sub
    End If

End Sub

End Sub
End Module

```

## **Staff Sell Shares Page**

```

Public Class StaffSellShares
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
        Button1.Click
            Form1.StaffPage.LoadShareHolderAccountPage()
        End Sub

    Private Sub StaffSellShares_Load(sender As Object, e As EventArgs) Handles
        MyBase.Load
            ShareHolderModule.LoadShareHolderShares(Form1.StaffPage.SelectedShareHolderId)

            For Each Share As ShareHolderShareModel In
                ShareHolderModule.ShareHolderShares
                    ShareInput.Items.Add("Rs. " + Share.ShareIssuedPrice.ToString() + " -
" + Share.Qty.ToString() + " No.")
                Next
            End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
        Button2.Click
            If (ShareInput.SelectedIndex < 0) Then
                MessageBox.Show("Please select a share")
            ElseIf QtyInput.Value < 1 Then
                MessageBox.Show("Please enter a valid quantity")
            ElseIf QtyInput.Value >
                ShareHolderModule.ShareHolderShares.ElementAt(ShareInput.SelectedIndex).Qty Then

```

```

        MessageBox.Show("Only " +
ShareHolderModule.ShareHolderShares.ElementAt(ShareInput.SelectedIndex).Qty.ToString() + " shares available")
    Else
        SellShares()
    End If
End Sub

Private Sub SellShares()
    Dim ShareHolderShareId =
ShareHolderModule.ShareHolderShares.ElementAt(ShareInput.SelectedIndex).Id
    Dim ShareIssueId =
ShareHolderModule.ShareHolderShares.ElementAt(ShareInput.SelectedIndex).ShareIssueId

    Dim ShareHolderId = Form1.StaffPage.SelectedShareHolderId
    Dim Quantity = QtyInput.Value
    Dim DeleteRequired = False

    If (QtyInput.Value =
ShareHolderModule.ShareHolderShares.ElementAt(ShareInput.SelectedIndex).Qty) Then
        DeleteRequired = True
    End If

    ShareHolderModule.SellShares(ShareHolderShareId, ShareIssueId, Quantity,
DeleteRequired)
End Sub
End Class

```

## Staff Transfer Shares Page

```

Public Class StaffTransferShares
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        Form1.StaffPage.LoadShareHolderAccountPage()
    End Sub

    Private Sub StaffTransferShares_Load(sender As Object, e As EventArgs)
Handles MyBase.Load
        ShareHolderModule.LoadShareHolderShares(Form1.StaffPage.SelectedShareHolderId)

        For Each Share As ShareHolderShareModel In
ShareHolderModule.ShareHolderShares
            ShareInput.Items.Add("Rs. " + Share.ShareIssuedPrice.ToString() + " -
" + Share.Qty.ToString() + " No.")
        Next

        ShareHolderModule.LoadShareHoldersData()
        For Each SH As ShareHolderModel In ShareHolderModule.ShareHolders
            TargetShareHolderInput.Items.Add(SH.Id.ToString() + " - " +
SH.Name.ToString())
        Next
    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
        If (ShareInput.SelectedIndex < 0) Then
            MessageBox.Show("Please select a share")
        ElseIf QtyInput.Value < 1 Then
            MessageBox.Show("Please enter a valid quantity")
        End If
    End Sub
End Class

```

```

        ElseIf QtyInput.Value >
ShareHolderModule.ShareHolderShares.ElementAt(ShareInput.SelectedIndex).Qty Then
            MessageBox.Show("Only " +
ShareHolderModule.ShareHolderShares.ElementAt(ShareInput.SelectedIndex).Qty.ToString() + " shares available")
        ElseIf (TargetShareHolderInput.SelectedIndex < 0) Then
            MessageBox.Show("Please select a share holder to transfer shares")
        Else
            TarnsferShares()
        End If
    End Sub

    Private Sub TarnsferShares()
        Dim ShareHolderShareId =
ShareHolderModule.ShareHolderShares.ElementAt(ShareInput.SelectedIndex).Id
        Dim ShareIssueId =
ShareHolderModule.ShareHolderShares.ElementAt(ShareInput.SelectedIndex).ShareIssueId
        Dim ShareHolderId = Form1.StaffPage.SelectedShareHolderId
        Dim Quantity = QtyInput.Value
        Dim DeleteRequired = False
        Dim TargetShareHolderId =
ShareHolderModule.ShareHolders.ElementAt(TargetShareHolderInput.SelectedIndex).Id

        If (QtyInput.Value =
ShareHolderModule.ShareHolderShares.ElementAt(ShareInput.SelectedIndex).Qty) Then
            DeleteRequired = True
        End If

        ShareHolderModule.TransferShares(ShareHolderShareId, ShareIssueId,
TargetShareHolderId, Quantity, DeleteRequired)
    End Sub
End Class

```

## Manager Module

```
Imports MySql.Data.MySqlClient
```

```
Module ManagerModule
```

```

    Sub AddManager(Name As String, Username As String, Password As String)
        Dim Sql = "insert into users (name, username, password, is_manager)
values(@Name, @Username, @Password, true)"
        Try
            Form1.DbConnection.Open()
            Dim Cmd = New MySqlCommand(Sql, Form1.DbConnection)

            Cmd.Parameters.AddWithValue("@Name", Name)
            Cmd.Parameters.AddWithValue("@Username", Username)
            Cmd.Parameters.AddWithValue("@Password", Password)

            Dim Res = Cmd.ExecuteScalar()

            Form1.DbConnection.Close()
            ' Navigate to admin page
            Form1.LoadManagerPage()

        Catch ex As Exception
            MessageBox.Show(ex.Message)
        Finally
            Form1.DbConnection.Close()
        End Try
    End Sub

```

```

End Sub

Sub ManagerLogin(Uname As String, password As String)
    Dim Sql = "select name, username, password, is_manager from users where
username=@Username"
    Try
        Form1.DbConnection.Open()
        Dim Cmd = New MySqlCommand(Sql, Form1.DbConnection)
        Cmd.Parameters.AddWithValue("@Username", Uname)

        Dim Reader = Cmd.ExecuteReader

        If Reader.Read() = True Then
            If Reader("is_manager") = True Then
                If Reader("password") = password Then
                    Dim Name = Reader("name")
                    Form1.LoggedInUserName = Name

                    Form1.DbConnection.Close()
                    Form1.LoadManagerPage()
                Else
                    MessageBox.Show("Invalid password")
                End If
            Else
                MessageBox.Show("Invalid username")
            End If
        Else
            MessageBox.Show("Invalid username")
        End If

        Catch ex As Exception
            MessageBox.Show(ex.Message)
        Finally
            Form1.DbConnection.Close()
        End Try
    End Sub
End Module

```

## User Module

```

Imports MySql.Data.MySqlClient

Module UserModule
    Public UserExists = False
    Public Users As New List(Of UserModel)

    Sub IsUsersExists()
        Dim Cmd = New MySqlCommand("select * from users", Form1.DbConnection)

        Try
            Form1.DbConnection.Open()
            Dim Reader = Cmd.ExecuteReader()

            If Reader.HasRows() = True Then
                UserExists = True
            End If
        Catch ex As Exception
            MessageBox.Show(ex.Message)
        End Try
    End Sub
End Module

```

```

        Finally
            Form1.DbConnection.Close()
        End Try
    End Sub

    Sub LoadUserData()
        Users.Clear()
        Dim Cmd = New MySqlCommand("select * from users", Form1.DbConnection)

        Try
            Form1.DbConnection.Open()
            Dim Reader = Cmd.ExecuteReader()

            If Reader.HasRows Then
                While Reader.Read()
                    Users.Add(New UserModel With {
                        .Id = Reader("id"),
                        .Name = Reader("name"),
                        .Username = Reader("username"),
                        .IsManager = Reader("is_manager")
                    })
                End While
            End If
        Catch ex As Exception
            MessageBox.Show(ex.Message)
        Finally
            Form1.DbConnection.Close()
        End Try
    End Sub

    Sub AddUser(Name As String, Username As String, Password As String, IsManager
As Boolean)
        Dim Sql1 = "select name, username, password, is_manager from users where
username=@Username"
        Dim Sql2 = "insert into users (name, username, password, is_manager)
values (@Name, @Username, @Password, @IsManager)"

        Try
            Form1.DbConnection.Open()

            ' Check if the user already exists
            Dim CheckCmd = New MySqlCommand(Sql1, Form1.DbConnection)
            CheckCmd.Parameters.AddWithValue("@Username", Username)
            Dim Reader = CheckCmd.ExecuteReader

            If Reader.Read() = True Then
                MessageBox.Show("User laready exists")
                Return
            End If
            Form1.DbConnection.Close()

            ' Adding new user
            Form1.DbConnection.Open()
            Dim Cmd = New MySqlCommand(Sql2, Form1.DbConnection)
            Cmd.Parameters.AddWithValue("@Name", Name)
            Cmd.Parameters.AddWithValue("@Username", Username)
            Cmd.Parameters.AddWithValue("@Password", Password)
            Cmd.Parameters.AddWithValue("@IsManager", IsManager)
            Cmd.ExecuteScalar()
        End Try
    End Sub

```

```

        Form1.DbConnection.Close()

        ' Navigate to user management page
        Form1.ManagerPage.LoadUsersPage()

    Catch ex As Exception
        MessageBox.Show(ex.Message)

    Finally
        Form1.DbConnection.Dispose()
    End Try
End Sub
End Module

Public Class UserModel
    Public Id As Integer
    Public Name As String
    Public Username As String
    Public IsManager As Boolean
End Class

```

## Equity Module

```

Imports MySql.Data.MySqlClient

Module EquityModule
    Public IssuedShareList As New List(Of ShareIssueModel)
    Sub IssueShares(Qty As Integer, Price As Integer)
        Dim Sql = "insert into share_issues (issued_quantity, unsold_quantity,
price) values (@Qty, @Qty, @Price)"

        Dim Cmd = New MySqlCommand(Sql, Form1.DbConnection)
        Cmd.Parameters.AddWithValue("Qty", Qty)
        Cmd.Parameters.AddWithValue("Price", Price)

        Try
            Form1.DbConnection.Open()
            Cmd.ExecuteNonQuery()

            Form1.DbConnection.Close()
            Form1.ManagerPage.LoadEquityPage()
        Catch ex As Exception
            MessageBox.Show(ex.Message)
        Finally
            Form1.DbConnection.Close()
        End Try
    End Sub

    Sub LoadEquityData()
        IssuedShareList.Clear()
        Dim Sql = "select * from share_issues"
        Dim Cmd = New MySqlCommand(Sql, Form1.DbConnection)

        Try
            Form1.DbConnection.Open()
            Dim Reader = Cmd.ExecuteReader()

            If (Reader.HasRows()) Then
                While Reader.Read()
                    Dim ShareIssue = New ShareIssueModel With {

```

```

        .Id = Reader("id"),
        .IssuedDate = Reader("issued_date"),
        .IssuedQty = Reader("issued_quantity"),
        .UnsoldQty = Reader("unsold_quantity"),
        .Price = Reader("price")
    }

    IssuedShareList.Add(ShareIssue)
End While
End If
Catch ex As Exception
    MessageBox.Show(ex.Message)
Finally
    Form1.DbConnection.Close()
End Try
End Sub
End Module

Class ShareIssueModel
    Public Id As Integer
    Public IssuedDate As Date
    Public IssuedQty As Integer
    Public UnsoldQty As Integer
    Public Price As Integer
End Class

```

## Staff Module

```

Imports MySql.Data.MySqlClient
Module StaffModule
    Sub StaffLogin(Username As String, password As String)
        Dim Sql = "select name, username, password, is_manager from users where
username=@Username"
        Try
            Form1.DbConnection.Open()
            Dim Cmd = New MySqlCommand(Sql, Form1.DbConnection)
            Cmd.Parameters.AddWithValue("@Username", Username)

            Dim Reader = Cmd.ExecuteReader

            If Reader.Read() = True Then
                If Reader("is_manager") = False Then
                    If Reader("password") = password Then
                        Dim Name = Reader("name")
                        Form1.LoggedInUserName = Name

                        Form1.DbConnection.Close()
                        Form1.LoadStaffPage()
                    Else
                        MessageBox.Show("Invalid password")
                    End If
                Else
                    MessageBox.Show("Invalid username")
                End If
            Else
                MessageBox.Show("Invalid username")
            End If

        Catch ex As Exception
            MessageBox.Show(ex.Message)
        End Try
    End Sub
End Module

```



```

        Finally
            Form1.DbConnection.Close()
        End Try
    End Sub
End Module

```

## ShareHolderModule

```
Imports MySql.Data.MySqlClient
```

```
Module ShareHolderModule
```

```

    Public ShareHolders As New List(Of ShareHolderModel)
    Public ShareHolderExists As Boolean
    Public SelectedShareHolder As ShareHolderModel
    Public ShareHolderShares As New List(Of ShareHolderShareModel)

```

```

    Sub AddNewShareHolder(SH As ShareHolderModel)
        Dim Sql = "insert into share_holders (name, age, gender, phone, aadhaar,
addr1, addr2, addr3, pin) values(@Name, @Age, @Gender, @Phone, @Aadhaar, @Addr1,
@Addr2, @Addr3, @PIN )"
        Dim Cmd = New MySqlCommand(Sql, Form1.DbConnection)
        Cmd.Parameters.AddWithValue("@Name", SH.Name)
        Cmd.Parameters.AddWithValue("@Age", SH.Age)
        Cmd.Parameters.AddWithValue("@Gender", SH.Gender)
        Cmd.Parameters.AddWithValue("@Phone", SH.Phone)
        Cmd.Parameters.AddWithValue("@Aadhaar", SH.Aadhaar)
        Cmd.Parameters.AddWithValue("@Addr1", SH.Addr1)
        Cmd.Parameters.AddWithValue("@Addr2", SH.Addr2)
        Cmd.Parameters.AddWithValue("@Addr3", SH.Addr3)
        Cmd.Parameters.AddWithValue("@PIN", SH.PIN)

        Try
            Form1.DbConnection.Open()
            Cmd.ExecuteNonQuery()

            Form1.DbConnection.Close()
            Form1.StaffPage.LoadShareHoldersPage()
        Catch ex As Exception
            MessageBox.Show(ex.Message)
        Finally
            Form1.DbConnection.Close()
        End Try
    End Sub

```

```

    Sub UpdateShareHolder(SH As ShareHolderModel)
        Dim Sql = "update share_holders set name = @Name, age = @Age, gender =
@Gender, phone = @Phone, aadhaar = @Aadhaar, addr1 = @Addr1, addr2 = @Addr2,
addr3 = @Addr3, pin = @PIN where id = @Id"
        Dim Cmd = New MySqlCommand(Sql, Form1.DbConnection)
        Cmd.Parameters.AddWithValue("Id", SH.Id)
        Cmd.Parameters.AddWithValue("@Name", SH.Name)
        Cmd.Parameters.AddWithValue("@Age", SH.Age)
        Cmd.Parameters.AddWithValue("@Gender", SH.Gender)
        Cmd.Parameters.AddWithValue("@Phone", SH.Phone)
        Cmd.Parameters.AddWithValue("@Aadhaar", SH.Aadhaar)
        Cmd.Parameters.AddWithValue("@Addr1", SH.Addr1)
        Cmd.Parameters.AddWithValue("@Addr2", SH.Addr2)
        Cmd.Parameters.AddWithValue("@Addr3", SH.Addr3)
        Cmd.Parameters.AddWithValue("@PIN", SH.PIN)

        Try
            Form1.DbConnection.Open()

```

```

        Cmd.ExecuteNonQuery()

        Form1.DbConnection.Close()
        Form1.StaffPage.LoadShareHoldersPage()
    Catch ex As Exception
        MessageBox.Show(ex.Message)
    Finally
        Form1.DbConnection.Close()
    End Try
End Sub
Sub GetShareHolderById(Id As Integer)
    Dim Sql = "select * from share_holders where id = @Id"

    Dim Cmd = New MySqlCommand(Sql, Form1.DbConnection)
    Cmd.Parameters.AddWithValue("@Id", Id)

    Try
        Form1.DbConnection.Open()
        Dim Reader = Cmd.ExecuteReader

        Reader.Read()
        If (Reader.HasRows()) Then
            ShareHolderExists = True
            SelectedShareHolder = New ShareHolderModel With {
                .Id = Reader("id"),
                .Name = Reader("name"),
                .Age = Reader("age"),
                .Gender = Reader("gender"),
                .Phone = Reader("phone"),
                .Aadhaar = Reader("aadhaar"),
                .Addr1 = Reader("addr1"),
                .Addr2 = Reader("addr2"),
                .Addr3 = Reader("addr3"),
                .PIN = Reader("pin")
            }
        End If
    Catch ex As Exception
        MessageBox.Show(ex.Message)
    Finally
        Form1.DbConnection.Close()
    End Try
End Sub

Sub LoadShareHoldersData()
    ShareHolders.Clear()

    Dim Sql = "select * from share_holders"
    Dim Cmd = New MySqlCommand(Sql, Form1.DbConnection)

    Try
        Form1.DbConnection.Open()
        Dim Reader = Cmd.ExecuteReader()

        If (Reader.HasRows()) Then
            While Reader.Read()
                Dim SH = New ShareHolderModel With {
                    .Id = Reader("id"),
                    .Name = Reader("name"),
                    .Phone = Reader("phone")
                }

                ShareHolders.Add(SH)
            End While
        End If
    End Try
End Sub

```

```

        End If
    Catch ex As Exception
        MessageBox.Show(ex.Message)
    Finally
        Form1.DbConnection.Close()
    End Try

End Sub

Sub BuyShares(ShareIssueId As Integer, ShareHolderId As Integer, Quantity As Integer)
    Dim Sql1 = "select * from share_holder_shares where share_holder_id = @ShareHolderId and share_issue_id = @ShareIssueId"
    Dim Sql2 = "insert into share_holder_shares (quantity, share_holder_id, share_issue_id) values (@Quantity, @ShareHolderId, @ShareIssueId)"
    Dim Sql3 = "update share_holder_shares set quantity = (@Quantity + quantity) where share_holder_id = @ShareHolderId and share_issue_id = @ShareIssueId"
    Dim Sql4 = "update share_issues set unsold_quantity = (unsold_quantity - @Quantity) where id = @Id"

    Dim Cmd1 = New MySqlCommand(Sql1, Form1.DbConnection)
    Cmd1.Parameters.AddWithValue("@ShareHolderId", ShareHolderId)
    Cmd1.Parameters.AddWithValue("@ShareIssueId", ShareIssueId)

    Try
        Form1.DbConnection.Open()
        Dim R = Cmd1.ExecuteReader()

        R.Read()

        If (R.HasRows()) Then
            Dim Cmd3 = New MySqlCommand(Sql3, Form1.DbConnection)
            Cmd3.Parameters.AddWithValue("@Quantity", Quantity)
            Cmd3.Parameters.AddWithValue("@ShareHolderId", ShareHolderId)
            Cmd3.Parameters.AddWithValue("@ShareIssueId", ShareIssueId)

            R.Close()
            Cmd3.ExecuteNonQuery()

            Dim Cmd4 = New MySqlCommand(Sql4, Form1.DbConnection)
            Cmd4.Parameters.AddWithValue("Quantity", Quantity)
            Cmd4.Parameters.AddWithValue("@id", ShareIssueId)

            Cmd4.ExecuteNonQuery()

            Form1.DbConnection.Close()
            Form1.StaffPage.LoadShareHolderAccountPage()
        Else
            Dim Cmd2 = New MySqlCommand(Sql2, Form1.DbConnection)
            Cmd2.Parameters.AddWithValue("@Quantity", Quantity)
            Cmd2.Parameters.AddWithValue("@ShareHolderId", ShareHolderId)
            Cmd2.Parameters.AddWithValue("@ShareIssueId", ShareIssueId)

            R.Close()
            Cmd2.ExecuteNonQuery()

            Dim Cmd4 = New MySqlCommand(Sql4, Form1.DbConnection)
            Cmd4.Parameters.AddWithValue("Quantity", Quantity)
            Cmd4.Parameters.AddWithValue("@id", ShareIssueId)

            Cmd4.ExecuteNonQuery()
        End If
    End Try
End Sub

```

```

        Form1.DbConnection.Close()
        Form1.StaffPage.LoadShareHolderAccountPage()
    End If
Catch ex As Exception
    MessageBox.Show(ex.Message)
Finally
    Form1.DbConnection.Close()
End Try
End Sub

Sub SellShares(ShareHolderShareId As Integer, ShareIssueId As Integer,
Quantity As Integer, DeleteRequired As Boolean)
    If (DeleteRequired) Then
        Dim Sql1 = "delete from share_holder_shares where id = @Id"
        Dim Cmd1 = New MySqlCommand(Sql1, Form1.DbConnection)
        Cmd1.Parameters.AddWithValue("@Id", ShareHolderShareId)

        Dim Sql2 = "update share_issues set unsold_quantity =
(unsold_quantity + @Qty) where id = @ShareIssueId"
        Dim Cmd2 = New MySqlCommand(Sql2, Form1.DbConnection)
        Cmd2.Parameters.AddWithValue("@Qty", Quantity)
        Cmd2.Parameters.AddWithValue("@ShareIssueId", ShareIssueId)

        Try
            Form1.DbConnection.Open()
            Cmd1.ExecuteNonQuery()
            Cmd2.ExecuteNonQuery()

            Form1.DbConnection.Close()
            Form1.StaffPage.LoadShareHolderAccountPage()
        Catch ex As Exception
            MessageBox.Show(ex.Message)
        Finally
            Form1.DbConnection.Close()
        End Try
    Else
        Dim Sql1 = "update share_holder_shares set quantity = (quantity -
@Qty) where id = @Id"
        Dim Cmd1 = New MySqlCommand(Sql1, Form1.DbConnection)
        Cmd1.Parameters.AddWithValue("@Qty", Quantity)
        Cmd1.Parameters.AddWithValue("@Id", ShareHolderShareId)

        Dim Sql2 = "update share_issues set unsold_quantity =
(unsold_quantity + @Qty) where id = @ShareIssueId"
        Dim Cmd2 = New MySqlCommand(Sql2, Form1.DbConnection)
        Cmd2.Parameters.AddWithValue("@Qty", Quantity)
        Cmd2.Parameters.AddWithValue("@ShareIssueId", ShareIssueId)

        Try
            Form1.DbConnection.Open()
            Cmd1.ExecuteNonQuery()
            Cmd2.ExecuteNonQuery()

            Form1.DbConnection.Close()
            Form1.StaffPage.LoadShareHolderAccountPage()
        Catch ex As Exception
            MessageBox.Show(ex.Message)
        Finally
            Form1.DbConnection.Close()
        End Try
    End If
End Sub

```

```

Sub TransferShares(ShareHolderShareId As Integer, ShareIssueId As Integer,
TargetShareHolderId As Integer, Quantity As Integer, DeleteRequired As Boolean)
    If (DeleteRequired) Then
        Dim Sql1 = "delete from share_holder_shares where id = @Id"
        Dim Cmd1 = New MySqlCommand(Sql1, Form1.DbConnection)
        Cmd1.Parameters.AddWithValue("@Id", ShareHolderShareId)

        Dim Sql2 = "select * from share_holder_shares where share_holder_id =
@TargetShareHolderId and share_issue_id = @ShareIssueId"
        Dim Cmd2 = New MySqlCommand(Sql2, Form1.DbConnection)
        Cmd2.Parameters.AddWithValue("@TargetShareHolderId",
TargetShareHolderId)
        Cmd2.Parameters.AddWithValue("@ShareIssueId", ShareIssueId)

        Dim Sql3 = "update share_holder_shares set quantity = (quantity +
@Qty) where share_holder_id = @TargetShareHolderId and share_issue_id =
@ShareIssueId"
        Dim Cmd3 = New MySqlCommand(Sql3, Form1.DbConnection)
        Cmd3.Parameters.AddWithValue("@Qty", Quantity)
        Cmd3.Parameters.AddWithValue("@TargetShareHolderId",
TargetShareHolderId)
        Cmd3.Parameters.AddWithValue("@ShareIssueId", ShareIssueId)

        Dim Sql4 = "insert into share_holder_shares (quantity,
share_holder_id, share_issue_id) values (@Quantity, @TargetShareHolderId,
@ShareIssueId)"
        Dim Cmd4 = New MySqlCommand(Sql4, Form1.DbConnection)
        Cmd4.Parameters.AddWithValue("@Quantity", Quantity)
        Cmd4.Parameters.AddWithValue("@TargetShareHolderId",
TargetShareHolderId)
        Cmd4.Parameters.AddWithValue("@ShareIssueId", ShareIssueId)

        Try
            Form1.DbConnection.Open()
            Cmd1.ExecuteNonQuery()

            Dim R1 = Cmd2.ExecuteReader()

            If (R1.HasRows()) Then
                R1.Close()
                Cmd3.ExecuteNonQuery()
            Else
                R1.Close()
                Cmd4.ExecuteNonQuery()
            End If

            Form1.DbConnection.Close()
            Form1.StaffPage.LoadShareHolderAccountPage()
        Catch ex As Exception
            MessageBox.Show(ex.Message)
        Finally
            Form1.DbConnection.Close()
        End Try
    Else
        Dim Sql1 = "update share_holder_shares set quantity = (quantity -
@Qty) where id = @Id"
        Dim Cmd1 = New MySqlCommand(Sql1, Form1.DbConnection)
        Cmd1.Parameters.AddWithValue("@Qty", Quantity)
        Cmd1.Parameters.AddWithValue("@Id", ShareHolderShareId)

        Dim Sql2 = "select * from share_holder_shares where share_holder_id =
@TargetShareHolderId and share_issue_id = @ShareIssueId"
        Dim Cmd2 = New MySqlCommand(Sql2, Form1.DbConnection)

```

```

        Cmd2.Parameters.AddWithValue("@TargetShareHolderId",
TargetShareHolderId)
        Cmd2.Parameters.AddWithValue("@ShareIssueId", ShareIssueId)

        Dim Sql3 = "update share_holder_shares set quantity = (quantity +
@Qty) where share_holder_id = @TargetShareHolderId and share_issue_id =
@ShareIssueId"
        Dim Cmd3 = New MySqlCommand(Sql3, Form1.DbConnection)
        Cmd3.Parameters.AddWithValue("@Qty", Quantity)
        Cmd3.Parameters.AddWithValue("@TargetShareHolderId",
TargetShareHolderId)
        Cmd3.Parameters.AddWithValue("@ShareIssueId", ShareIssueId)

        Dim Sql4 = "insert into share_holder_shares (quantity,
share_holder_id, share_issue_id) values (@Quantity, @TargetShareHolderId,
@ShareIssueId)"
        Dim Cmd4 = New MySqlCommand(Sql4, Form1.DbConnection)
        Cmd4.Parameters.AddWithValue("@Quantity", Quantity)
        Cmd4.Parameters.AddWithValue("@TargetShareHolderId",
TargetShareHolderId)
        Cmd4.Parameters.AddWithValue("@ShareIssueId", ShareIssueId)

    Try
        Form1.DbConnection.Open()
        Cmd1.ExecuteNonQuery()

        Dim R1 = Cmd2.ExecuteReader()
        If (R1.HasRows()) Then
            R1.Close()
            Cmd3.ExecuteNonQuery()
        Else
            R1.Close()
            Cmd4.ExecuteNonQuery()
        End If

        Form1.DbConnection.Close()
        Form1.StaffPage.LoadShareHolderAccountPage()
    Catch ex As Exception
        MessageBox.Show(ex.Message)
    Finally
        Form1.DbConnection.Close()
    End Try
End If
End Sub

Sub LoadShareHolderShares(ShareHolderId As Integer)
    ShareHolderShares.Clear()

    Dim Sql = "select * from share_holder_shares where share_holder_id =
@ShareHolderId"
    Dim Cmd = New MySqlCommand(Sql, Form1.DbConnection)
    Cmd.Parameters.AddWithValue("@ShareHolderId", ShareHolderId)

    Dim Sql2 = "select * from share_issues where id = @Id"

    Try
        Form1.DbConnection.Open()
        Dim R = Cmd.ExecuteReader()
        If R.HasRows() Then
            While R.Read()
                Dim Share = New ShareHolderShareModel With {
                    .Id = R("id"),
                    .Qty = R("quantity"),

```

```

        .ShareIssueId = R("share_issue_id")
    }
    ShareHolderShares.Add(Share)
End While
End If

R.Close()
For Each Share As ShareHolderShareModel In ShareHolderShares
    Dim Cmd2 = New MySqlCommand(Sql2, Form1.DbConnection)
    Cmd2.Parameters.AddWithValue("@Id", Share.ShareIssueId)

    Dim R2 = Cmd2.ExecuteReader()

    R2.Read()
    Share.ShareIssuedPrice = R2("price")
    R2.Close()
Next

Catch ex As Exception
    MessageBox.Show(ex.Message)
Finally
    Form1.DbConnection.Close()
End Try
End Sub

End Module

Class ShareHolderModel
    Public Id As Integer
    Public Name As String
    Public Age As Integer
    Public Gender As String
    Public Phone As String
    Public Aadhaar As String
    Public Addr1 As String
    Public Addr2 As String
    Public Addr3 As String
    Public PIN As String
End Class

Class ShareHolderShareModel
    Public Id As Integer
    Public Qty As Integer
    Public ShareHolderId As Integer
    Public ShareIssueId As Integer
    Public ShareIssuedPrice As Integer
End Class

```

## **DbHelper**

```
Imports MySql.Data.MySqlClient
```

```
Module DbHelper
```

```
    Sub CreateTables()
```

```
        Dim UsersTableSchema = "create table if not exists users (id int not null
auto_increment primary key, name varchar(25) not null, username varchar(25) not
null, password varchar(20) not null, is_manager bool not null )"

```

```
        Dim ShareIssuesTableSchema = "create table if not exists share_issues (id
int not null auto_increment primary key, issued_date datetime default now(),
issued_quantity int not null, unsold_quantity int not null, price int)"

```

```

        Dim ShareHoldersTableSchema = "create table if not exists share_holders
(id int not null auto_increment primary key, name varchar(40) not null, age int
not null, gender varchar(6) not null, phone varchar(10) not null, aadhaar
varchar(12) , addr1 varchar(20) not null, addr2 varchar(20), addr3 varchar(20),
pin varchar(6) not null)"
        Dim ShareHolderSharesTableSchema = "create table if not exists
share_holder_shares (id int not null auto_increment primary key, quantity int not
null, share_holder_id int not null, share_issue_id int not null)"
        Try
            Form1.DbConnection.Open()
            Dim cmd = New MySqlCommand(UsersTableSchema, Form1.DbConnection)
            Dim cmd2 = New MySqlCommand(ShareIssuesTableSchema,
Form1.DbConnection)
            Dim cmd3 = New MySqlCommand(ShareHoldersTableSchema,
Form1.DbConnection)
            Dim cmd4 = New MySqlCommand(ShareHolderSharesTableSchema,
Form1.DbConnection)
            cmd.ExecuteNonQuery()
            cmd2.ExecuteNonQuery()
            cmd3.ExecuteNonQuery()
            cmd4.ExecuteNonQuery()

            Catch ex As Exception
                MessageBox.Show(ex.Message)
            Finally
                Form1.DbConnection.Close()
                Form1.DbConnection.Dispose()
            End Try
        End Sub
End Module

```



# **SCREENSHOTS**

## Add Manager Page

Form1

### ShareMarket Management

Please add the manager details

Name

Username

Password

Create Manager

## Manager Login

Form1


### Manager Panel

Please login.

Username

Password

Login

A colorful illustration depicting business collaboration and growth. It features several large puzzle pieces in blue and orange, with a stack of gold coins on top of one piece. A person in a blue suit is placing an orange puzzle piece. In the background, there are icons for a lightbulb, a magnifying glass, a heart, and a thumbs up. At the bottom, three people are working together at a desk with a laptop. The scene is set against a backdrop of stylized clouds and a tree.

Go To Staff Login

## Manager Equity Page

Form1

Management Panel

Equity

Users

Logout

Equity

Issue Share

| Issue Date | Issued Qty | Price per share | Unsold qty | Amount raised |
|------------|------------|-----------------|------------|---------------|
| 11-03-2022 | 100        | 20              | 100        | 0             |
| 11-03-2022 | 50         | 3               | 50         | 0             |
| 11-03-2022 | 30         | 8               | 30         | 0             |
| 11-03-2022 | 60         | 12              | 60         | 0             |

## Manager Issue Share Page

Form1

Management Panel

Equity

Users

Logout

Issue Share

Go Back

Quantity

0

Price per share

0

Issue

## Manager Users Page

Form1

Management Panel

Equity

Users

Logout

Add User

| Id | Name         | Username    | User Type |
|----|--------------|-------------|-----------|
| 1  | James Bond   | Jamesbond   | Manager   |
| 2  | Rabecca John | rabeccajohn | Staff     |
| 3  | Tom Clancy   | tomclancy   | Staff     |
| 4  | Ricky Ben    | rickyben    | Staff     |

## Manager Add User Page

Form1

Management Panel

Equity

Users

Logout

Add User

Name

Username

Password

User Type

Add User

Go Back

## Staff Login Page

Form1

Welcome to *StockMan*

#1 stock management platform.

Please login to continue.

Username

Password

Login

Go To Manager

## Staff Share Holders Page

Form1

Staff Panel

Share Holders

Logout

Share Holders

Add Share Holder

| Name      | id | Phone      |
|-----------|----|------------|
| Mary      | 1  | 9878765678 |
| Raphel    | 2  | 9988776678 |
| Jim Carry | 3  | 6787656789 |

## Staff Add Share Holder Page

Form1

Staff Panel

Share Holders

Logout

Add Share Holder

Name

Age

0

Gender

Phone

Aadhaar No.

Address 1

Address 2

Address3

PIN

Go Back

Add Share Holder

## Staff Share Holder Account Page

Form1

Staff Panel

Share Holders

Logout

Share Holder Account

| Quantity | Price |
|----------|-------|
| 12       | 20    |
| 4        | 3     |
| 20       | 12    |
| 4        | 8     |

Total Worth (Rs.) : 524

Buy Share

## Staff Edit Share Holder Page

Form1

Staff Panel

Share Holders

Logout

Edit Share Holder

Name

Mary Clara

Age

34

Gender

Female

Phone

9878765600

Aadhaar No.

456744567546

Address 1

#20/234

Address 2

Ramapuram

Address3

Bangalore

PIN

598767

Go Back

Save

## Staff Buy Share Page

Form1

Staff Panel

Share Holders

Logout

Buy Share

Go Back

Share

Quantity

0

Buy

## Staff Sell share

Form1

Staff Panel

Share Holders

Logout

Sell Shares

Go Back

Share

Quantity

Sell

## Staff Transfer Share

Form1

Staff Panel

Share Holders

Logout

Transfer Shares

Go Back

Share

Quantity

Target ShareHolder

Transfer



# **REPORTRS**

## User Report

Add User

| Id | Name         | Username    | User Type |
|----|--------------|-------------|-----------|
| 1  | James Bond   | Jamesbond   | Manager   |
| 2  | Rabecca John | rabeccajohn | Staff     |
| 3  | Tom Clancy   | tomclancy   | Staff     |
| 4  | Ricky Ben    | rickyben    | Staff     |

## Share Issue Report

### Equity

Issue Share

| Issue Date | Issued Qty | Price per share | Unsold qty | Amount raised |
|------------|------------|-----------------|------------|---------------|
| 11-03-2022 | 100        | 20              | 88         | 240           |
| 11-03-2022 | 50         | 3               | 46         | 12            |
| 11-03-2022 | 30         | 8               | 26         | 32            |
| 11-03-2022 | 60         | 12              | 40         | 240           |

## Share Holder Report

## Share Holders

[Add Share Holder](#)

| Name      | id | Phone      |
|-----------|----|------------|
| Mary      | 1  | 9878765678 |
| Raphel    | 2  | 9988776678 |
| Jim Carry | 3  | 6787656789 |

# **DATABASE TABLES**

### **Users Table (users)**

| Column Name | Data Type |
|-------------|-----------|
| id          | int       |
| name        | varchar   |
| username    | varchar   |
| password    | varchar   |
| Is_manager  | bool      |

### **Share Issues Table (share\_issues)**

| Column Name     | Data Type |
|-----------------|-----------|
| id              | int       |
| issued_date     | datetime  |
| issued_quantity | int       |
| unsold_quantity | int       |
| price           | int       |

### **Share Holders Table (share\_holders)**

| Column Name | Data Type |
|-------------|-----------|
| id          | int       |
| name        | varchar   |
| age         | int       |
| gender      | varchar   |
| phone       | varchar   |
| aadhaar     | varchar   |
| addr1       | varchar   |
| addr2       | varchar   |
| addr3       | varchar   |
| pin         | varchar   |

### **Share Holder Shares Table (share holder shares)**

| Column Name     | Data Type |
|-----------------|-----------|
| id              | int       |
| quantity        | int       |
| share_holder_id | int       |
| Share_issue_id  | int       |