Technical Document Login

Table of Contents

Activity

- Log into Calculus

Introduction

- This screen is used to log into the Calculus system

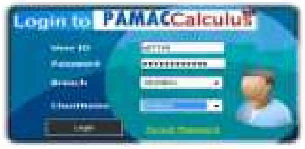
Menu Path

- https://pamaconline.com/calculus/Pages/Login.aspx

Screen Name

- Log in

Screen Description



- Login page – contains
  - User id (input by the user)
  - Password
  - Branch name (select from Drop down)
  - Client (Select from Drop down)
  - Submit Button

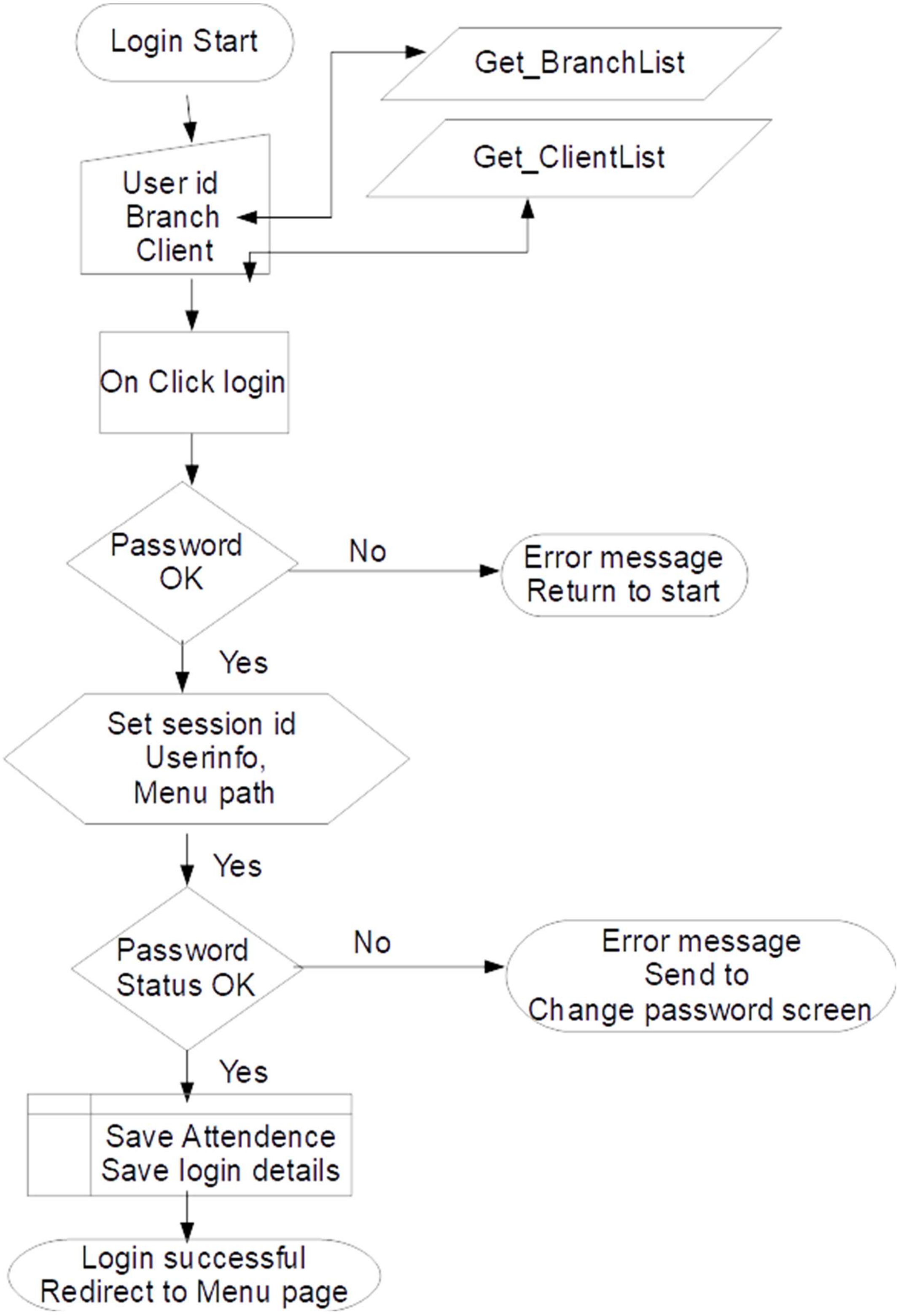
Program name

- **Login.aspx** and **login.aspx.cs**

Process mode

- log in

Process Flow diagram



login-flowchart.odg

Program process description - *login.aspx*

The program ***login.aspx*** is the front end program and the program ***login.aspx.cs*** is the process or back end program. The description given below would deal with the back end process namely *login.aspx.cs.*

The program is called first to support the front end to get the drop down boxes and also to retain the values selected by the user.

The Branch name is the first drop down – The same is got by the following process.

Get the values from data base using the function ***Get_BranchList ()*** which calls the Stored Procedure - ***Get_AllBranchList*** which gets two fields required for the drop down namely *BranchName* and *BranchId* from the Table - **BranchInfo** and inserts into *ddlBranchList* for the user to select.

The Client name is the next drop down – The same is got by the following process.

Get the values from data base using the function ***Get_ClientList()*** which calls the Stored Procedure - ***Get_ClientList*** which gets two fields required for the drop down namely *ClientName* and *ClientId* from the Table - t**blClientDetails** and inserts into *ddlClientList* for the user to select.

These are the only two drop downs in this screen

The btnLogin_Click

Once the user has entered the information and clicks Login Button - The *btnLogin_Click (object sender, EventArgs e)* function is called.

Check_Auth_user

This function in turn calls *Check_Auth_user()*

It calls the stored procedure *Auth_user12 (varUserId, varPassword (the encrypted value of the password entered, intBranchId* and first checks whether the password is matching with the data in the table.

If password does not match –

a message is shown on screen : "Incorrect UserId or Password,[Please enter Your PAMAC EMPLOYEE Code]!" and calls

savelogindetails12()
Here the stored procedure *save_loginstatus_password* is called and the fact is saved in the table **logindetails**

If password matches

The session variables given below are filled using the *UserInfo (class)*

- BranchID
- Branch Name
- UserID
- GroupID
- GroupName
- VerticalID
- UserName
- Product
- ClientName (from the selected value from drop down)
- ClientID (from the selected value from drop down)
- Get last access time from MasterDBF (system IO file)
- Get createdDate from MasterDBF (system io file)
- Get the Menupaths using the following function

IsAuthorizeUser_ForPage(SaveUSERInfo.GroupName);

IsAuthorizeUser_ForPage(SaveUSERInfo.GroupName);

This returns the menu for this group and stored as part of the UserInfo class.

Check_IsSystem()

- This function is called to verify the status of the user password or in other words check the password correctness using the stored procedure *Get_UserPasswordStatus* (@userid) and get the result in return parameter @return value. The function returns the value of 1 if password has expired and value of 2 if password requires to be changed as set by Admin and 3 if password has reached the days limit (a message to this effect is sent to front end) and if it return value other than 1,2,or3 then the function

Check_Password(string pstrPassword)

This is a redundant para – but as it Is there in the program this is mentioned, If this returns true proceed else return error “not complying with policy”.

- If the return valiue of *Check_IsSystem* is True then redirect to Menu and call

logdetails()

- This function would call stored procedure *Sp_AssignAttendance12* to insert/append created by in the table “**Attendance_master**”

QCTrackAutoAssign

  - This function would call stored procedure ***uspQCTrackAutoAssign*** to store the details for auto assign.
- If the return value of *Check_IsSystem* is false then
  - The user is redirected to “Change_Password.aspx”

savelogindetails12

Here the stored procedure *save_loginstatus_password* is called and the fact is saved in the table **logindetails**

Autorole()
This para is called and the stored procedure *Sp_AutoRole12* is called – This SP saves the roles attached to this user id in a table called ats_rolemaster.

As the response.redirect has ‘false’ after the execution of this line the program will get redirected to Menu page.

Stored Procedures used

Get_AllBranchList

Get_ClientList

Get_UserPasswordStatus

Sp_AssignAttendance12

Sp_AutoRole12

Tables used

BranchInfo

tblClientDetails

logindetails

Attendance_master

UserInfo

ats_role_master

****