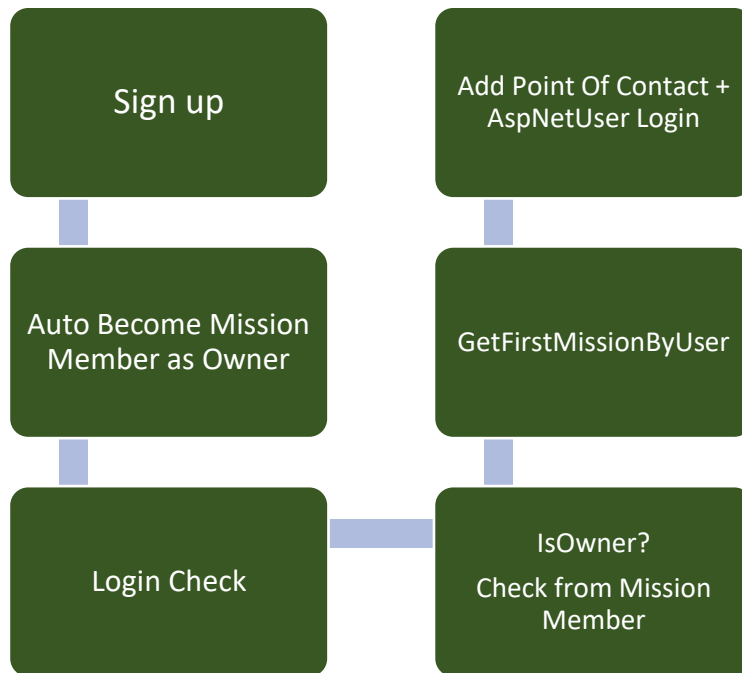
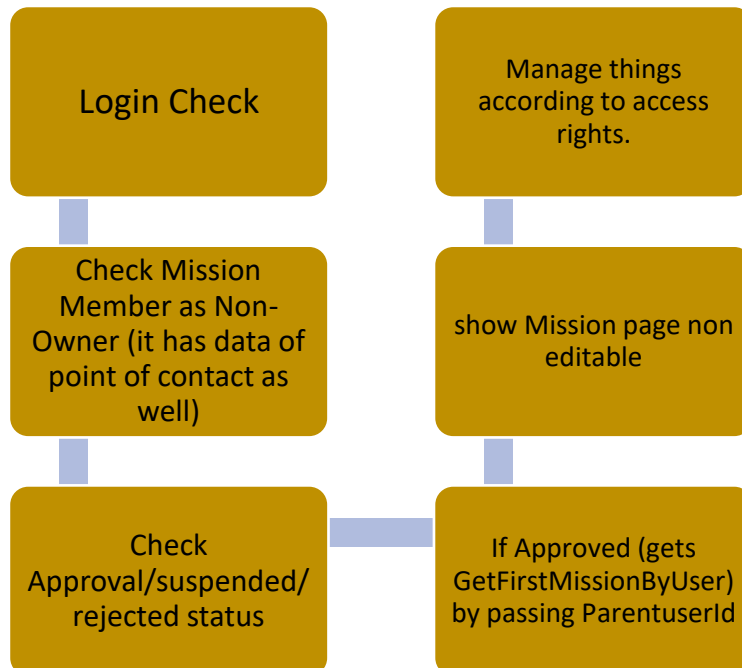


When Missionary (Owner) Will login system will check below flow?



When Non-Owner (point of contact) will login the process will be?



APIs Details in flow (for Owner – Missionary user)

After success login check is the current user is owner from “MissionMember” table, for that call this API.

| | |
|---------|---|
| URL: | Baseurl/api/ Missions/MissionMembersByUser/currentuserid |
| Method: | GET |
| Body: | NULL Just send current logged In user Id in URL |

1. If you get response from above API, that mean this user is the owner (how added from main sign up screen) then keep previous process this API will return you ‘**ParentUserID**’ use that to get this user first mission API
2. Baseurl/api/**Missions/GetUsersFirstMission/ParentUserID**
3. Now this user can have access to change public non public toggle (on/off) and can add point of contacts from there

| | |
|---------|--|
| URL: | Baseurl/api/ Missions/PointofContactsCreate |
| Method: | POST |
| Body: | <p>Model class of PointOfContact table</p> <p>Send these values</p> <pre>{ PointofContactName = Osman; ContactEmail= Osman@gmail.com; ContactPhone= 123123123; Position = Founder (value from dropdown of positions); MissionID = currentmissionid of logged In User; PublicID = current public-non-public id of logged In User; FromScreen = "Public-Private"; ApprovalStatus = false; AutoApprovalStatus = false; AddedByEmail = currentuserId (example 79); }</pre> |

Notes: Once you will send this information to above API that will auto create point of contact along record in Mission Member table as non-Owner profile.

Notes: If same missionary try to add same point of contact user again it will return 'Email Already Exist.' means show message on screen "Point of contact account already added by you."

Model Class of PointOfContact

```
public int POCID { get; set; }
public string PointofContactName { get; set; }
public string ContactEmail { get; set; }
public string ContactPhone { get; set; }
public Nullable<int> MissionID { get; set; }
public string AddedByEmail { get; set; }
public Nullable<int> PublicID { get; set; }
public Nullable<int> UserID { get; set; }
public Nullable<int> ParrentID { get; set; }
public string FromScreen { get; set; }
public string ApprovalCounter { get; set; }
public Nullable<bool> ApprovalStatus { get; set; }
public Nullable<bool> AutoApprovalStatus { get; set; }
public string Position { get; set; }
```

Once Point of contact added show the grid to missionary (owner) by calling below API

| | |
|---------|--|
| URL: | Baseurl/api/Missions/MissionMembersByMission/missionId |
| Method: | GET |
| Body: | NULL Just send current mission Id in URL |

- It will return mission member data along ASPNETUSER data and POINTOFCONTACT data, if user is owner (missionary) than POINTOFCONTACT will be null in return and if user is non-owner that record will have detail of POINTOFCONTACT with it.

The model class of Mission Member table

```
public int MissionMemberID { get; set; }
public Nullable<int> UserID { get; set; }
public Nullable<bool> Status { get; set; }
public Nullable<bool> ReadStatus { get; set; }
public string MissionName { get; set; }
public Nullable<System.DateTime> CreatedDate { get; set; }
public Nullable<int> MissionID { get; set; }
public Nullable<int> ParentUserID { get; set; }
public string FromScreen { get; set; }
public Nullable<int> RoleID { get; set; }
public Nullable<int> POCID { get; set; }
public Nullable<bool> DesireTDecline { get; set; }
public Nullable<bool> Decline { get; set; }
public Nullable<bool> Approve { get; set; }
public Nullable<bool> AutoApprove { get; set; }
public Nullable<bool> TempSuspended { get; set; }
```

```

public Nullable<bool> PermanentBlock { get; set; }
public Nullable<bool> SendStandardQuestion { get; set; }
public Nullable<bool> CustomizeQuestion { get; set; }
public string BlockReason { get; set; }
public string SuspendedReason { get; set; }
public string BlockedBy { get; set; }
public string SuspendedBy { get; set; }
public Nullable<System.DateTime> SuspendedDate { get; set; }
public Nullable<System.DateTime> BlockedDate { get; set; }
public Nullable<System.DateTime> ApprovedDate { get; set; }
public virtual ICollection<PointofContact> PointofContacts { get; set; }
public virtual ICollection<AspNetUser> AspNetUsers { get; set; }

```

The list grid will be like this following:

| FOUNDERS, BOARD OF DIRECTORS, OFFICERS(PRESIDENT ,VP) & ADMINS | | | | | | |
|--|--------------------|-----------------|-----------------------------|--|-----------------|---|
| Mission Admin (Position) | usman (Name) | usman@gmail.com | SignUp (comes from) | <input checked="" type="checkbox"/> Auto-Approve | | |
| Founder (Position) | osman amjad (Name) | osman@gmail.com | Public-Private (comes from) | 1 (How many user added) | usman@gmail.com | <input type="checkbox"/> Temporarily Suspended <input type="checkbox"/> Permanent Block |

Notes: The first row will always be owner (missionary) means from response record which have PointofContacts: null in response & rest of rows will be point of contact users.

After showing list user can suspend and block the point of contact, once user will click suspended or blocked checkbox, pop up will appear and asked for 'Reason'.

| | |
|---------|--|
| URL: | Baseurl/api/ Missions /MissionMemberUpdate/MissionMemberID |
| Method: | POST |
| Body: | <pre> { MissionMemberID = Selective row mission member id PermanentBlock = true // if checked true else false BlockedBy = currentUser email // send if user is blocking BlockReason = //if block send reason as note TempSuspended = true // if checked true else false SuspendedBy = currentUser email // send if user is suspending SuspendedReason = // if suspended send reason as note } </pre> |

APIs Details in flow (for non-Owner – Missionary user)

After check if user not owner then check if user is non-owner from “MissionMember” table, for that call this API.

| | |
|---------|---|
| URL: | Baseurl/api/ Missions/MissionMembersByNonOwnerUser/currentuserid |
| Method: | GET |
| Body: | NULL Just send current logged In user Id in URL |

1. If you get response from above API, that mean this user is the non-owner (how added from point of contact screen by some other user) then check all bool status value from this response, if user is approved and not blocked, suspended then keep previous process, this API will return you ‘ParentUserID’use that to get user’s owner first mission API.
2. Check the following after response from above API, check if user is approved or auto approved then send “ParentUserId” from response to get it’s owner mission otherwise show messages like below.

```

if (responseObjectMember != null && responseObjectMember.Count() > 0)
{
    if (responseObjectMember.Count() > 0)
    {
        if (responseObjectMember[0].Status == true && responseObjectMember[0].Approve == true || responseObjectMember[0].AutoApprove == true)
        {
            apiurl = "Missions/GetUsersFirstMission" + "/" + responseObjectMember.ElementAt(0).ParentUserID;
        }
        else if (responseObjectMember[0].Status == false)
        {
            TempData["response"] = "Your Point Of Contact Account Is In Approval Process"; ;
            return RedirectToAction("Index", "Home");
        }
        else if (responseObjectMember[0].Decline == true)
        {
            TempData["response"] = "Your Point Of Contact Account Is Declined By Your Administration"; ;
            return RedirectToAction("Index", "Home");
        }
        else if (responseObjectMember[0].PermanentBlock == true)
        {
            TempData["response"] = "Your Point Of Contact Account Is Permanently Blocked By Your Administration"; ;
            return RedirectToAction("Index", "Home");
        }
        else if (responseObjectMember[0].SendStandardQuestion == true)
        {
            TempData["response"] = "Your Administration Will Asked You Some Standard Questions To Give Access Back"; ;
            return RedirectToAction("Index", "Home");
        }
        else if (responseObjectMember[0].CustomizeQuestion == true)
        {
            TempData["response"] = "Your Administration Will Asked You Some Customized Questions To Give Access Back"; ;
            return RedirectToAction("Index", "Home");
        }
    }
}

```