Rino

ALTER PROCEDURE [dbo].[App\_Notification\_ByUserId\_GetList]

@UserId bigint,

@Keyword nvarchar(255),

@PageIndex int,

@PageSize int,

@TotalRecord bigint out,

@Sort nvarchar(100),

@Direction nvarchar(100)

as

begin try

begin

select

[Title] = nt.[Title],

[Description] = nt.[Description],

[Content] = nt.[Content],

[CreateTime] = nt.[CreateTime],

[IsDelete] = nt.[IsDelete],

[CreatedUserId] = nt.[CreatedUserId],

[Type] = nt.[Type],

[ObjectId] = nt.[ObjectId],

[Id] = nt.[Id],

IsRead = xf.IsRead,

Icon = nt.Icon

from [dbo].[nt\_Notification\_AccountGuest\_Xref] xf (nolock)

inner join nt\_Notification nt on xf.NotificationId = nt.Id and xf.ObjectId = @UserId

left join cat\_Product p on p.ProductId = nt.ObjectId and nt.Type = 'PRODUCT'-- and isnull(p.IsDelete, 0) = 0

--left join cat\_Content\_New n on n.[NewId] = nt.ObjectId and (nt.Type = 'EVENT' or nt.Type = 'NEW')-- and isnull(n.IsDelete, 0) = 0

--left join cat\_Service s on s.[ServiceId] = nt.ObjectId and nt.Type = 'SERVICE'-- and isnull(s.IsDelete, 0) = 0

where

isnull(nt.IsDelete, 0) = 0 AND

( nt.[Title] like Concat('%',@Keyword,'%') or

isnull(@Keyword,'') = ''

)

and ((nt.Type = 'PRODUCT' and isnull(p.IsDelete, 0) = 0) or ((nt.Type = 'EVENT' or nt.Type = 'NEW')) or (nt.Type = 'SERVICE' and isnull(nt.IsDelete,0) = 0))

order by nt.[CreateTime] desc, xf.IsRead asc

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

select

@TotalRecord = count(xf.Id)

from [dbo].[nt\_Notification\_AccountGuest\_Xref] xf (nolock)

inner join nt\_Notification nt on xf.NotificationId = nt.Id and xf.ObjectId = @UserId

left join cat\_Product p on p.ProductId = nt.ObjectId and nt.Type = 'PRODUCT'-- and isnull(p.IsDelete, 0) = 0

--left join cat\_Content\_New n on n.[NewId] = nt.ObjectId and (nt.Type = 'EVENT' or nt.Type = 'NEW')-- and isnull(n.IsDelete, 0) = 0

--left join cat\_Service s on s.[ServiceId] = nt.ObjectId and nt.Type = 'SERVICE'-- and isnull(s.IsDelete, 0) = 0

where

isnull(nt.IsDelete, 0) = 0 AND

( nt.[Title] like Concat('%',@Keyword,'%') or

isnull(@Keyword,'') = ''

)

and ((nt.Type = 'PRODUCT' and isnull(p.IsDelete, 0) = 0) or ((nt.Type = 'EVENT' or nt.Type = 'NEW')) or (nt.Type = 'SERVICE' and isnull(nt.IsDelete,0) = 0))

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'App\_Notification\_ByUserId\_GetList: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@UserId=' + convert(nvarchar, @UserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'nt\_Notification', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[App\_Notification\_ChangeStatus\_Read]

@Id int,

@UserId bigint,

@StatusID int out

as

begin try

set @Id = isnull(@Id, 0)

set @StatusID = 0

if exists ( select 1

from [nt\_Notification\_AccountGuest\_Xref] cct (nolock)

where cct.NotificationId = @Id and cct.ObjectId = @UserId

)

begin

Update [nt\_Notification\_AccountGuest\_Xref]

Set IsRead =1

where NotificationId = @Id and ObjectId = @UserId

set @StatusID = 1

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500),

@SessionID int

set @ErrorNum = error\_number()

set @ErrorMsg = 'App\_Notification\_ChangeStatus\_Read: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@NotificationId=' + convert(varchar,@Id)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'nt\_Notification\_AccountGuest\_Xref', 'DEL', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[App\_User\_Device\_InsertUpdate]

@UserName nvarchar(255),

@XMLID int

AS

BEGIN TRY

DECLARE @UserId int = 0

SELECT @UserId = u.UserId

FROM sec\_User u

WHERE u.UserName=@UserName

IF NOT EXISTS(SELECT UserId FROM [sec\_User\_Device] WHERE UserID=@UserId and (DeviceType = 1 or DeviceType = 2 or DeviceType = 3)) and @UserId > 0

BEGIN

--Insert mode

INSERT INTO [sec\_User\_Device]

(

[UserId],

[DeviceToken],

[DeviceType],

Lang)

SELECT

[UserId] =@UserId,

[DeviceToken] =x.[DeviceToken],

[DeviceType] =x.[DeviceType],

[Lang] =x.[Lang]

FROM OPENXML(@XMLID, '/UserDeviceModel', 2)

WITH (

[DeviceToken] varchar(255),

[DeviceType] smallint,

Lang nvarchar(10)) x

END

ELSE

BEGIN

--Update mode

UPDATE uud

SET

[DeviceToken] =x.[DeviceToken],

[DeviceType] =x.[DeviceType],

[Lang] =x.[Lang]

FROM OPENXML(@XMLID, '/UserDeviceModel', 2)

WITH (

[DeviceToken] varchar(255),

[DeviceType] smallint,

Lang nvarchar(10)) x

INNER JOIN [sec\_User\_Device] uud (nolock)

on uud.UserID = @UserID

WHERE

ISNULL(uud.[DeviceToken],'') <>ISNULL(x.[DeviceToken],'')

or ISNULL(uud.[DeviceType], 0) <>ISNULL(x.[DeviceType], 0)

or ISNULL(uud.[Lang], '') <>ISNULL(x.[Lang], '')

END

END TRY

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'sec\_User\_Device\_InsertUpdate: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@UserName=' + convert(nvarchar, @UserName)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, '[sec\_User\_Device]', 'UPD', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[App\_User\_Device\_Save]

@UserName NVARCHAR(255),

@XML NVARCHAR(max)

AS

BEGIN TRY

DECLARE @XMLID INT

SET @UserName = ISNULL(@UserName,'')

SET @XML = dbo.ufn\_Replace\_XmlChars(@XML)

EXECUTE sp\_xml\_preparedocument @XMLID OUT, @XML

EXECUTE App\_User\_Device\_InsertUpdate @UserName, @XMLID

EXECUTE sp\_xml\_removedocument @XMLID

END TRY

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'sec\_User\_Device\_Save: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@UserName=' + convert(nvarchar, @UserName)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, '[sec\_User\_Device\_Save]', 'UPD', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Banner\_Get]

@BannerId int,

@UserId int

as

begin try

select

[BannerId] = r.[BannerId],

[Name] =Isnull(r.[Name],'{"en":"","vi":""}'),

[Content] =Isnull(r.[Content],'{"en":"","vi":""}'),

[Description] = Isnull(r.[Description],'{"en":"","vi":""}'),

[IsDelete] = r.[IsDelete],

[CreateBy] = r.[CreateBy],

[CreateAt] = r.[CreateAt],

[UpdateBy] = r.[UpdateBy],

[UpdateAt] = r.[UpdateAt],

[Avatar] = r.Avatar

from [dbo].[cat\_Banner] r (nolock)

where

isnull(r.IsDelete,0) = 0

and r.BannerId = @BannerId

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Banner\_Get: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Banner=' + convert(nvarchar, @BannerId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Banner\_Get', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Banner\_GetList]

@UserId bigint,

@Keyword nvarchar(255),

@PageIndex int,

@PageSize int,

@TotalRecord bigint out,

@Sort nvarchar(100),

@Direction nvarchar(100)

as

begin try

begin

select

[BannerId] = ctgr.BannerId,

[Name] =Isnull(ctgr.[Name],'{"en":"","vi":""}'),

[Content] =Isnull(ctgr.[Content],'{"en":"","vi":""}'),

[Description] = Isnull(ctgr.[Description],'{"en":"","vi":""}'),

[Avatar] = ctgr.[Avatar],

[CreateBy] = ctgr.[CreateBy],

[CreateAt] = ctgr.[CreateAt],

[UpdateBy] = ctgr.[UpdateBy],

[UpdateAt] = ctgr.[UpdateAt],

[IsDelete] = ctgr.[IsDelete]

from [dbo].[cat\_Banner] ctgr (nolock)

where

isnull(ctgr.IsDelete,0) = 0 AND

( ctgr.[Name] like Concat('%',@Keyword,'%') or

isnull(@Keyword,'') = ''

)

ORDER BY ctgr.BannerId DESC

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

select

@TotalRecord = count(ctgr.BannerId)

from [dbo].[cat\_Banner] ctgr (nolock)

where

isnull(ctgr.IsDelete,0) = 0 AND

( ctgr.[Name] like Concat('%',@Keyword,'%') or

isnull(@Keyword,'') = ''

)

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Banner\_GetList: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@UserId=' + convert(nvarchar, @UserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Banner\_GetList', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_CancelReason\_Get]

@ReasonId int,

@UserId bigint

as

begin try

select

[ReasonId] = cr.[ReasonId],

[Description] = Isnull(cr.[Description],'{"en":"","vi":""}')

from [dbo].[cat\_Reason] cr (nolock)

where (@ReasonId = 0 or isnull(cr.ReasonId,0)=@ReasonId )

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_CancelReason\_Get: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@ReasonId=' + convert(nvarchar, @ReasonId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Category', 'GET', null, @AddlInfo

end catch

ALTER procedure [dbo].[cat\_Cart\_Check\_List\_Product]

@UserId bigint,

@XML nvarchar(max)

as

begin try

declare @XMLID int

set @XML = dbo.ufn\_Replace\_XmlChars(@XML)

exec sp\_xml\_preparedocument @XMLID out, @XML

exec cat\_Cart\_Check\_List\_Product\_Status @UserId, @XMLID

exec sp\_xml\_removedocument @XMLID

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Cart\_Check\_List\_Product: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = ''

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Cart\_Check\_List\_Product', 'UPD', @SessionID, @AddlInfo

end catch

ALTER procedure [dbo].[cat\_Cart\_Check\_List\_Product\_Status]

@UserId bigint,

@XMLID int

as

begin try

--thêm sp vào giỏ hàng

select

[ProductId] =x.[ProductId],

[Quantity] =cp.Quantity, -- số lượng có sẵn trong clinic

Total = x.[Total], -- số lượng ng dùng muốn mua

[Status] =IIF(cp.Quantity>x.[Total], 1, 0)

from openxml(@XMLID, '/CheckTotalProduct/ListCheckProduct/CheckTotalProduct', 2)

with (

[ProductId] int,

[Total] int

) x

left join [dbo].[cat\_Product] cp on cp.ProductId = x.[ProductId]

order by IIF(cp.Quantity>x.[Total], 1, 0) ASC

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Cart\_Check\_List\_Product\_Status: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = ''

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Cart\_Check\_List\_Product\_Status', 'UPD', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Cart\_Number\_Product]

@UserId bigint,

@Status int output

as

begin try

set @Status =

(SELECT COUNT(ProductId)

FROM cat\_Cart\_User

where UserId = @UserId and Isnull(IsDelete,0)=0)

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Cart\_Number\_Product: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@UserId=' + convert(nvarchar,@UserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Cart\_Number\_Product', 'GET', null, @AddlInfo

end catch

ALTER procedure [dbo].[cat\_Cart\_User\_Delete]

@ListId nvarchar(max),

@StatusID int out,

@UserId bigint

as

begin try

update [dbo].[cat\_Cart\_User]

set IsDelete =1

where CartUserId

IN (SELECT id FROM convert\_string\_to\_table (@ListId, ',') )

set @StatusID =1

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Cart\_User\_Delete: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@ListId=' + convert(nvarchar, @ListId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Cart\_User\_Delete', 'GET', null, @AddlInfo

end catch

ALTER procedure [dbo].[cat\_Cart\_User\_Get]

@CartUserId bigint,

@UserId bigint

as

begin try

SELECT

[CartUserId] =cod.[CartUserId]

,[UserId] =cod.[UserId]

,[ProductId] =cod.[ProductId]

,[Name] =Isnull(cp.Name,'"en":"","vi":""}')

,[ClinicId] =cod.[ClinicId]

,[ClinicName] =Isnull(cc.Name,'"en":"","vi":""}'),

[Number] = cod.Number,

ClinicBranchId = cod.ClinicBranchId,

ClinicBranchName =Isnull(br.Name,'"en":"","vi":""}'),

Avatar =prm.[Path] ,

[Price] =cp.[Price]

--PercentSale =hd.[PercentageDiscount]

,[CreateAt] =cod.[CreateAt]

,[UpdateAt] =cod.[UpdateAt]

,[Address] = Isnull(br.[Address],'{"en":"","vi":""}')

,[City] =prv.Name

,[District] =dt.Name

,[Ward] =cw.Name

FROM [cat\_Cart\_User] cod

left join cat\_clinic cc on cc.[ClinicId]=cod.[ClinicId]

left join cat\_clinic br on br.[ClinicId]=cod.ClinicBranchId

left join cat\_Product cp on cod.ProductId=cp.ProductId

--left join cat\_Hot\_Deals hd on hd.HotDealsId = cp.[HotDealsId]

left join cat\_Ward cw on cw.WardId =br.Ward

left join cat\_District dt on dt.DistrictId =br.District

left join cat\_Province prv on prv.ProvinceId=br.City

left join cat\_Product\_Image prm on prm.ProductId = cp.ProductId and Isnull(prm.IsDelete,0)=0 and prm.IsActive =1

where CartUserId= @CartUserId and Isnull(cod.IsDelete,0)=0

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Cart\_User\_Get: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@CartUserId=' + convert(nvarchar, @CartUserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Cart\_User\_Get', 'GET', null, @AddlInfo

end catch

ALTER procedure [dbo].[cat\_Cart\_User\_GetList]

@UserId bigint

as

begin try

-- danh sach cac chi nhanh user da dat hang

select

distinct ClinicBranchId = ClinicBranchId,

ClinicBranchName =Isnull(ccb.Name,'"en":"","vi":""}'),

ClinicBranchAddress =ccb.Address,

ClinicId = ccu.ClinicId,

ClinicName = Isnull(cc.Name,'"en":"","vi":""}'),

ClinicAddress = cc.Address,

[Transport] = ccb.Transport

from cat\_Cart\_User ccu

left join cat\_Clinic cc on ccu.ClinicId=cc.ClinicId

left join cat\_Clinic ccb on ccb.ClinicId = ccu.ClinicBranchId

where ccu.userId=@UserId and ISNULL(ccu.IsDelete,0)=0

-- danh sách các sản phẩm dc sắp xếp theo thời gian

SELECT

[CartUserId] =cod.[CartUserId]

,[UserId] =cod.[UserId]

,[ProductId] =cod.[ProductId]

,[Name] =Isnull(cp.Name,'"en":"","vi":""}')

, [Number] = cod.Number,

Avatar =prm.[Path] ,

[Price] =iif(Isnull(cp.[IsGeneral],0)=0,cp.[Price],cod.[PriceGeneralProduct]),

[NoteGeneralProduct] = cod.[NoteGeneralProduct],

[PercentSale] = hd.Discount,

TypeReduction = hd.TypeReduction,

IsGeneral = Isnull(cp.IsGeneral,0),

[CreateAt] =cod.[CreateAt]

,[UpdateAt] =cod.[UpdateAt]

,[Address] = Isnull(br.[Address],'{"en":"","vi":""}')

,[City] =prv.Name

,[District] =dt.Name

,[Ward] =cw.Name,

[ClinicBranchId] = cod.[ClinicBranchId],

[TotalProduct] = Isnull(cp.Quantity,0),

[StatusSale] = Isnull(iif((GETDATE() >= (TRY\_PARSE(hd.[StartDate] AS DATETIME USING 'en-gb')) and (getdate() < DATEADD(day, 1, TRY\_PARSE(hd.[EndDate] AS DATETIME USING 'en-gb')))),1,0),0)

FROM [cat\_Cart\_User] cod

left join cat\_clinic cc on cc.[ClinicId]=cod.[ClinicId]

left join cat\_clinic br on br.[ClinicId]=cod.ClinicBranchId

left join cat\_Product cp on cod.ProductId=cp.ProductId

left join cat\_Product\_HotDeal hd on hd.ProductId = cp.ProductId

left join cat\_Ward cw on cw.WardId =br.Ward

left join cat\_District dt on dt.DistrictId =br.District

left join cat\_Province prv on prv.ProvinceId=br.City

left join cat\_Product\_Image prm on prm.ProductId = cp.ProductId and Isnull(prm.IsDelete,0)=0 and prm.IsActive =1

where cod.[UserId] = @UserId and Isnull(cod.IsDelete,0)=0

order by cod.CreateAt DESC

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Cart\_User\_GetList: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@UserId=' + convert(nvarchar, @UserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Cart\_User\_GetList', 'GET', null, @AddlInfo

end catch

ALTER procedure [dbo].[cat\_Cart\_User\_Save]

@CartUserId int output,

@UserId bigint,

@XML nvarchar(max)

as

begin try

declare @XMLID int

set @CartUserId = isnull(@CartUserId,0)

set @XML = dbo.ufn\_Replace\_XmlChars(@XML)

exec sp\_xml\_preparedocument @XMLID out, @XML

exec cat\_Cat\_User\_InsertUpdate @CartUserId output,@UserId, @XMLID

exec sp\_xml\_removedocument @XMLID

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Cart\_User\_Save: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@@CartUserId=' + convert(nvarchar, @CartUserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Cart\_User\_Save', 'UPD', @SessionID, @AddlInfo

end catch

ALTER procedure [dbo].[cat\_Cat\_User\_InsertUpdate]

@CartUserId int out,

@UserId bigint,

@XMLID int

as

begin try

if @CartUserId <= 0

begin

--thêm sp vào giỏ hàng

insert into [cat\_Cart\_User]

(

[UserId]

,[ProductId]

,[ClinicId]

,[Number]

,[CreateAt]

,[ClinicBranchId]

,[PriceGeneralProduct]

,[NoteGeneralProduct]

)

select

[UserId] =@UserId,

[ProductId] =x.[ProductId],

[ClinicId] =x.[ClinicId] ,

[Number] =x.[Number] ,

[CreateAt] = GetDAte() ,

[ClinicBranchId] =x.[ClinicBranchId],

[PriceGeneralProduct] =x.[PriceGeneralProduct],

[NoteGeneralProduct] =x.[NoteGeneralProduct]

from openxml(@XMLID, '/CreateCart', 2)

with (

[UserId] bigint,

[ProductId] int,

[ClinicId] int,

[Number] int,

[CreateAt] datetime ,

[ClinicBranchId] int,

[PriceGeneralProduct] int,

[NoteGeneralProduct] nvarchar(max)

) x

set @CartUserId = scope\_identity()

-- check đã tồn tại sản phẩm đó trong giỏ hàng hay chưa

declare @check1 int

set @check1 = ISnull((select A.CartUserId from cat\_Cart\_User A , cat\_Cart\_User B

where A.ClinicBranchId = B.ClinicBranchId and A.ProductId = B.ProductId

and B.CartUserId = @CartUserId and A.UserId =B.UserId

and Isnull(A.IsDelete,0)=0 and Isnull(B.IsDelete,0)=0 and A.CartUserId <> B.CartUserId ),0)

-- nếu có thì cộng đồn số lượng từ số lần đã thêm trước đó vào lần đặt mới

If(@check1 >0)

begin

declare @number1 int

-- lấy số lượng lần đặt trước đó

set @number1 = (select Number from cat\_Cart\_User where CartUserId = @check1 and Isnull(IsDelete,0)=0)

-- cộng vào lần đặt mới

update cat\_Cart\_User set Number = ISNull(Number,0)+ Isnull(@number1,0) where CartUserId = @CartUserId

-- xóa đi lần đặt trước đó

update cat\_Cart\_User set IsDelete =1 where CartUserId = @check1

end

end

else

begin

--Update mode

update ccl

set

[Number] =x.[Number] ,

[ClinicBranchId] =x.[ClinicBranchId] ,

[CreateAt] = GetDAte() ,

[UpdateAt] = getdate() ,

[PriceGeneralProduct] =x.[PriceGeneralProduct],

[NoteGeneralProduct] =x.[NoteGeneralProduct]

from openxml(@XMLID, '/CreateCart', 2)

with (

[Number] int,

[ClinicBranchId] int,

[CreateAt] datetime ,

[UpdateAt] datetime ,

[PriceGeneralProduct] int,

[NoteGeneralProduct] nvarchar(max)

)x

inner join [cat\_Cart\_User] ccl (nolock)

on ccl.CartUserId = @CartUserId

where

isnull(ccl.[Number],0) <>isnull(x.[Number],0)

or isnull(ccl.[ClinicBranchId],0) <>isnull(x.[ClinicBranchId],0)

or isnull(ccl.[PriceGeneralProduct],0) <>isnull(x.[PriceGeneralProduct],0)

or isnull(ccl.[NoteGeneralProduct],'') <>isnull(x.[NoteGeneralProduct],'')

--declare @check int

--set @check = ISnull((select A.CartUserId from cat\_Cart\_User A , cat\_Cart\_User B

-- where A.ClinicBranchId = B.ClinicBranchId and A.ProductId = B.ProductId

-- and B.CartUserId = @CartUserId and A.UserId =B.UserId

-- and Isnull(A.IsDelete,0)=0 and Isnull(B.IsDelete,0)=0 and A.CartUserId <> B.CartUserId),0)

--If(@check >0)

--begin

-- declare @number int

-- set @number = (select Number from cat\_Cart\_User where CartUserId = @check and Isnull(IsDelete,0)=0)

-- update cat\_Cart\_User set Number =ISNull(Number,0)+ IsNull(@number,0) where CartUserId = @CartUserId

-- update cat\_Cart\_User set IsDelete =1 where CartUserId = @check

--end

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Cat\_User\_InsertUpdate: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@CartUserId=' + convert(nvarchar, @CartUserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Cat\_User\_InsertUpdate', 'UPD', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Category\_ChangeStatus]

@OrderId int,

@UserId int,

@StatusId int out

as

begin try

set @OrderId = isnull(@OrderId, 0)

declare @CheckOrderCashBack int -- kiem tra Order co ton tai va duoc thanh toan chua

set @CheckOrderCashBack =(select count(OrderId) from cat\_Order where CreateBy =@UserId and StatusId =1 and UseCashBack = 0 and TotalUseCashBack <= 0 and OrderId = @OrderId)

declare @CheckCashBack int -- khai bao so tien thanh toan khi tra toan bo don hang

set @CheckCashBack = (Select Total from cat\_Order where OrderId =@OrderId and CreateBy = @UserId and StatusId = 1)

declare @Bonus int -- tong so tien duoc hoan lai cua 1 nguoi

set @Bonus =(select sum(TotalUseCashBack) from cat\_Order where CreateBy =@UserId and StatusId =1 and TotalUseCashBack > 0 and isnull(IsDelete,0)=0)

if(@CheckOrderCashBack = 0 )

set @StatusId =0

else

-- tong so tien thanh toan nam trong khoang tu 3 den 6 trieu

if( 3000000 <= @CheckCashBack and @CheckCashBack <= 6000000)

begin

Update r Set r.UseCashBack =1,r.TotalUseCashBack = r.Total /100 \*3

from cat\_Order r (nolock)

where r.OrderId= @OrderId and r.StatusId = 1 and r.CreateBy =@UserId and ISNULL(r.IsDelete,0)=0

-- Update tong so tien duoc hoan lai trong bang cat\_Moey\_User\_Xref

Update mux Set mux.Bonus =@Bonus from cat\_Moey\_User\_Xref mux where mux.UserId =@UserId

set @StatusId = 1

end

-- tong so tien thanh toan luôn nam trong khoang tu 6 trieu den 10 trieu

else if(6001000 <= @CheckCashBack and @CheckCashBack <= 10000000)

begin

Update r

Set r.UseCashBack =1, r.TotalUseCashBack = r.Total /100 \*4

from cat\_Order r (nolock)

where r.OrderId= @OrderId and r.StatusId = 1 and r.CreateBy =@UserId and ISNULL(r.IsDelete,0)=0

-- Update tong so tien duoc hoan lai trong bang cat\_Moey\_User\_Xref

Update mux Set mux.Bonus =@Bonus from cat\_Moey\_User\_Xref mux where mux.UserId =@UserId

set @StatusId = 2

end ;

-- tong so tien thanh toan luôn nam trong khoang tu 10 trieu den 1tỉ

else if(10001000 <= @CheckCashBack and @CheckCashBack <= 10000000000)

begin

Update r

Set r.UseCashBack =1,r.TotalUseCashBack = r.Total /100 \*5

from cat\_Order r (nolock)

where r.OrderId= @OrderId and r.StatusId = 1 and r.CreateBy =@UserId and ISNULL(r.IsDelete,0)=0

set @StatusId = 3

-- Update tong so tien duoc hoan lai trong bang cat\_Moey\_User\_Xref

Update mux Set mux.Bonus =@Bonus from cat\_Moey\_User\_Xref mux where mux.UserId =@UserId

end ;

else -- tu 3 trieu do xuong hoac tren 1 tỉ

begin

Update r

Set r.UseCashBack =1,r.TotalUseCashBack = r.Total /100 \*2

from cat\_Order r (nolock)

where r.OrderId= @OrderId and r.CreateBy =@UserId and r.StatusId = 1 and isnull(r.IsDelete,0)=0

-- Update tong so tien duoc hoan lai trong bang cat\_Moey\_User\_Xref

Update mux Set mux.Bonus =@Bonus from cat\_Moey\_User\_Xref mux where mux.UserId =@UserId

set @StatusId = 4

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500),

@SessionID int

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Order\_ChangeStatus: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@OrderId=' + convert(varchar,@OrderId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Order\_ChangeStatus', 'DEL', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Category\_FindByName]

@Name nvarchar(255),

@UserId bigint,

@StatusID int out

as

begin try

set @StatusID = 0

if exists ( select 1

from [cat\_Category] cct (nolock)

where cct.Name = @Name

and isnull(IsDelete, 0) = 0

)

begin

set @StatusID = 1

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500),

@SessionID int

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Category\_FindByName: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Name=' + convert(varchar,@Name)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Category\_FindByName', '', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Category\_GetList]

@UserId bigint,

@Keyword nvarchar(255),

@PageIndex int,

@PageSize int,

@TotalRecord bigint out,

@Sort nvarchar(100),

@Direction nvarchar(100)

as

begin try

begin

select

[CategoryId] = ctgr.[CategoryId],

[Name] =Isnull(ctgr.[Name],''),

[Description] =Isnull(ctgr.[Description],''),

[Avatar] = ctgr.[Avatar],

[CreateBy] = ctgr.[CreateBy],

[CreateAt] = ctgr.[CreateAt],

[UpdateBy] = ctgr.[UpdateBy],

[UpdateAt] = ctgr.[UpdateAt],

[IsDelete] = ctgr.[IsDelete]

from [dbo].[cat\_Category] ctgr (nolock)

where

isnull(ctgr.IsDelete,0) = 0 AND

( ctgr.[Name] like Concat('%',@Keyword,'%') or

isnull(@Keyword,'') = ''

)

ORDER BY ctgr.CategoryId DESC

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

select

@TotalRecord = count(ctgr.CategoryId)

from [dbo].[cat\_Category] ctgr (nolock)

where

isnull(ctgr.IsDelete,0) = 0 AND

( ctgr.[Name] like Concat('%',@Keyword,'%') or

isnull(@Keyword,'') = ''

)

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Category\_GetList: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@UserId=' + convert(nvarchar, @UserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Category', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Category\_InsertUpdate]

@CategoryId int out,

@UserId bigint,

@XMLID int,

@XML nvarchar(max)

as

begin try

if @CategoryId <= 0

begin

--Insert mode

insert into [cat\_Category]

(

[Name],

[Status],

[Avatar],

[Code],

[ParentId],

[CreateBy],

[CreateTime],

[IsDelete]

)

select

[Name] = x.[NameObj],

[Status] = x.[Status],

[Avatar] = x.[Avatar],

[Code] = x.[Code],

[ParentId] = x.[ParentId],

[CreateBy] = @UserId,

[CreateTime] = Getdate(),

[IsDelete] = x.[IsDelete]

from openxml(@XMLID, '/CategoryRequestModel', 2)

with (

[NameObj] nvarchar(255),

[Status] int,

[Code] nvarchar(256),

[ParentId] int,

[Avatar] nvarchar(max),

[IsDelete] bit

) x

set @CategoryId = scope\_identity()

--insert action log

exec utl\_Insert\_ActionLog @tableName ='cat\_Category', @fieldPKName ='CategoryId', @fieldPKValue =@CategoryId, @actionName = 'insert', @XML =@XML, @createBy = @UserId

end

else

begin

--Update mode

update cct

set

[Name] = x.[NameObj],

[Status] = x.[Status],

[Code] =x.[Code],

[ParentId] =x.[ParentId],

[Avatar] = x.[Avatar],

[UpdateBy] = @UserId,

[UpdateTime] = Getdate()

from openxml(@XMLID, '/CategoryRequestModel', 2)

with (

[NameObj] nvarchar(255),

[Status] int,

[Code] nvarchar(256),

[ParentId] int,

[Avatar] nvarchar(max),

[UpdateBy] int,

[UpdateTime] datetime

) x

inner join [cat\_Category] cct (nolock)

on cct.CategoryId = @CategoryId

where isnull(cct.[Name],'') <>isnull(x.[NameObj],'')

or isnull(cct.[Status],'') <>isnull(x.[Status],'')

or isnull(cct.[Code],'') <>isnull(x.[Code],'')

or isnull(cct.[ParentId],0) <>isnull(x.[ParentId],0)

or isnull(cct.[Avatar],'') <>isnull(x.[Avatar],'')

--insert action log

exec utl\_Insert\_ActionLog @tableName ='cat\_Category', @fieldPKName ='CategoryId', @fieldPKValue =@CategoryId, @actionName = 'update', @XML =@XML, @createBy = @UserId

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Category\_InsertUpdate: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@CategoryId=' + convert(nvarchar, @CategoryId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Category', 'UPD', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Category\_Save]

@CategoryId int output,

@UserId bigint,

@XML nvarchar(max)

as

begin try

declare @XMLID int

set @CategoryId = isnull(@CategoryId,0)

set @XML = dbo.ufn\_Replace\_XmlChars(@XML)

exec sp\_xml\_preparedocument @XMLID out, @XML

exec cat\_Category\_InsertUpdate @CategoryId output, @UserId, @XMLID, @XML

exec sp\_xml\_removedocument @XMLID

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Category\_Save: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@CategoryId=' + convert(nvarchar, @CategoryId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Category\_Save', 'UPD', @SessionID, @AddlInfo

end catch

ALTER procedure [dbo].[cat\_Change\_Status\_Order]

@StatusID int out

as

begin try

update [dbo].[cat\_Order]

set IsDelete=1

where DATEDIFF(minute,[CreateAt],getdate()) >15 and StatusId =0

set @StatusID = 1

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Change\_Status\_Order: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = ''

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Change\_Status\_Order', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_ChatGroup\_CheckExist]

-- Add the parameters for the stored procedure here

@Id int,

@IsExists smallint out

AS

begin try

set @IsExists = 0

if(@Id != 0)

begin

if exists ( select 1

from [cat\_ChatGroup] u

where u.ChatGroupId=@Id

)

begin

set @IsExists = 1

end

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_ChatGroup\_CheckExist: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Id=' + convert(nvarchar, @Id)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, '[cat\_ChatGroup]', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_ChatGroup\_GetList]

@UserId bigint,

@Keyword nvarchar(255),

@PageIndex int,

@PageSize int,

@TotalRecord bigint out,

@Sort nvarchar(100),

@Direction nvarchar(100),

@Type nvarchar(100),

@IsUnRead bit

as

begin try

begin

declare @UserType int = 1, @ClinicId int = 0

select @UserType = u.UserType from sec\_User u where u.UserId = @UserId

select @ClinicId = cl.ClinicId from cat\_Clinic cl where cl.UserId = @UserId

;with ChatMessage\_CTE(ChatGroupId,LastMessage, LastMessageTime,[UnreadCount])

as

(

select

ChatGroupId = cct.ChatGroupId,

LastMessage = (select top 1 iif(cmm.IsLink = 1, N'Tin nhắn hình ảnh',cmm.[Message]) from cat\_ChatMessage cmm where cct.ChatGroupId = cmm.ChatGroupId order by cmm.CreateTime desc),

LastMessageTime = (select top 1 cmm.CreateTime from cat\_ChatMessage cmm where cct.ChatGroupId = cmm.ChatGroupId order by cmm.CreateTime desc),

[UnreadCount] = (select count(cm.ChatMessageId) from cat\_ChatMessage cm where cct.ChatGroupId = cm.ChatGroupId and isnull(cm.IsRead, 0) = 0 and isnull(cm.IsDelete, 0) = 0 and isnull(cm.TypeQA, 0) = 0)

from [dbo].[cat\_ChatGroup] cct (nolock)

)

select

ChatGroupId = cct.ChatGroupId,

[Title] = cct.[Title],

[AccountGuestId] = cct.[AccountGuestId],

[Avatar] = cct.[Avatar],

[CreateTime] = cct.[CreateTime],

[IsDelete] = cct.[IsDelete],

[ToClinicId] = cct.[ToClinicId],

[ToClinicName] = isnull((select top 1 c.[Name] from cat\_Clinic c where c.ClinicId = cct.[ToClinicId]), ''),

[UnreadCount] = cte.UnreadCount,

LastMessage = cte.LastMessage,

LastMessageTime = cte.LastMessageTime,

AccountGuestName = ac.FullName,

CreateName = u.FullName, --Chỉ khi list danh sach cua dieu phoi vs phong kham

TitleRefName = ct.Title

from [dbo].[cat\_ChatGroup] cct (nolock)

left join sec\_User u on u.UserId = cct.UserId

left join cat\_ChatGroup ct on cct.ChatGroupRefId = ct.ChatGroupId

left join sec\_AccountGuest ac on ac.AccountGuestId = ct.AccountGuestId

left join ChatMessage\_CTE cte on cte.ChatGroupId = cct.ChatGroupId

where

--cct.[AccountGuestId] = @UserId

--and

isnull(cct.IsDelete, 0) = 0

AND

(cct.[Title] like Concat('%',@Keyword,'%') or

isnull(@Keyword,'') = ''

)

and (isnull(@UserType, 1) = 1 or ( @UserType = 0 and cct.ToClinicId = @ClinicId))

and (isnull(@Type, '') = '' or (@Type = 'PK' and isnull(cct.ToClinicId, 0) > 0) or (@Type = 'KH' and isnull(cct.ToClinicId, 0) = 0))

and (isnull(@IsUnRead,0) = 0 or (@IsUnRead = 1 and cte.UnreadCount > 0))

order by cte.LastMessageTime desc

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

select

@TotalRecord = count(cc.ChatGroupId)

from

[cat\_ChatGroup] cc (nolock)

where

isnull(cc.IsDelete, 0) = 0

AND

(cc.[Title] like Concat('%',@Keyword,'%') or

isnull(@Keyword,'') = ''

)

and (isnull(@UserType, 1) = 1 or ( @UserType = 0 and cc.ToClinicId = @ClinicId))

and (isnull(@Type, '') = '' or (@Type = 'PK' and isnull(cc.ToClinicId, 0) > 0) or (@Type = 'KH' and isnull(cc.ToClinicId, 0) = 0))

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_ChatGroup\_GetList: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@UserId=' + convert(nvarchar, @UserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, '[cat\_ChatGroup]', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_ChatGroup\_GetList\_ChatGroupId]

@UserId bigint,

@Keyword nvarchar(255),

@PageIndex int,

@PageSize int,

@TotalRecord bigint out,

@Sort nvarchar(100),

@Direction nvarchar(100),

@ChatGroupId int

as

begin try

begin

select

ChatGroupId = cct.ChatGroupId,

[Title] = cct.[Title],

[AccountGuestId] = cct.[AccountGuestId],

[Avatar] = cct.[Avatar],

[CreateTime] = cct.[CreateTime],

[IsDelete] = cct.[IsDelete],

[Status] = cct.[Status],

[ToClinicId] = cct.[ToClinicId],

[ToClinicName] = isnull((select top 1 c.[Name] from cat\_Clinic c where c.ClinicId = cct.[ToClinicId]), ''),

[UnreadCount] = (select count(cm.ChatMessageId) from cat\_ChatMessage cm where cct.ChatGroupId = cm.ChatGroupId and isnull(cm.IsRead, 0) = 0),

LastMessage = (select top 1 cmm.[Message] from cat\_ChatMessage cmm where cct.ChatGroupId = cmm.ChatGroupId order by cmm.CreateTime desc)

from [dbo].[cat\_ChatGroup] cct (nolock)

where

--cct.[AccountGuestId] = @UserId

--and

isnull(cct.IsDelete, 0) = 0

AND

(cct.[Title] like Concat('%',@Keyword,'%') or

isnull(@Keyword,'') = ''

)

and cct.ChatGroupRefId = @ChatGroupId

order by CreateTime desc

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

select

@TotalRecord = count(cc.ChatGroupId)

from

[cat\_ChatGroup] cc (nolock)

where

isnull(cc.IsDelete, 0) = 0

AND

(cc.[Title] like Concat('%',@Keyword,'%') or

isnull(@Keyword,'') = ''

)

and cc.ChatGroupRefId = @ChatGroupId

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_ChatGroup\_GetList: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@UserId=' + convert(nvarchar, @UserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, '[cat\_ChatGroup]', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_ChatGroup\_InsertUpdate]

@ChatGroupId int out,

@UserId bigint,

@XMLID int

as

begin try

if @ChatGroupId <= 0

begin

--Insert mode

insert into [cat\_ChatGroup]

(

[Title],

UserId,

[Avatar],

[Status],

[ToClinicId],

[ChatGroupRefId],

[CreateTime],

[IsDelete]

)

select

[Title] = x.[Title],

UserId = @UserId,

[Avatar] = x.[Avatar],

[Status] = 0,

[ToClinicId] = x.[ToClinicId],

[ChatGroupRefId] = x.[ChatGroupRefId],

[CreateTime] = Getdate(),

[IsDelete] = 0

from openxml(@XMLID, '/ChatGroup', 2)

with (

[Title] nvarchar(256),

[Avatar] nvarchar(max),

[ToClinicId] int,

[ChatGroupRefId] int

) x

set @ChatGroupId = scope\_identity()

end

else

begin

--Update mode

update cct

set [Title] = x.[Title],

UserId = @UserId,

[Avatar] = x.[Avatar],

[Status] = 0,

[ToClinicId] = x.[ToClinicId],

[ChatGroupRefId] = x.[ChatGroupRefId],

[UpdateTime] = Getdate(),

[IsDelete] = 0

from openxml(@XMLID, '/ChatGroup', 2)

with (

[Title] nvarchar(256),

[Avatar] nvarchar(max),

[ToClinicId] int,

[ChatGroupRefId] int

) x

inner join [cat\_ChatGroup] cct (nolock)

on cct.ChatGroupId = @ChatGroupId

where

isnull(cct.[Title],'') <>isnull(x.[Title],'')or

isnull(cct.[Avatar],'') <>isnull(x.[Avatar],'') or

isnull(cct.[ToClinicId],0) <>isnull(x.[ToClinicId],0) or

isnull(cct.[ChatGroupRefId],0) <>isnull(x.[ChatGroupRefId],0)

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_ChatGroup\_InsertUpdate: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@ChatGroupId=' + convert(nvarchar, @ChatGroupId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_ChatGroup', 'UPD', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_ChatGroup\_Save]

@ChatGroupId int output,

@UserId bigint,

@XML nvarchar(max)

as

begin try

declare @XMLID int

set @ChatGroupId = isnull(@ChatGroupId,0)

set @XML = dbo.ufn\_Replace\_XmlChars(@XML)

exec sp\_xml\_preparedocument @XMLID out, @XML

exec cat\_ChatGroup\_InsertUpdate @ChatGroupId output, @UserId, @XMLID

exec sp\_xml\_removedocument @XMLID

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'App\_ChatGroup\_Save: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@ChatGroupId=' + convert(nvarchar, @ChatGroupId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'App\_ChatGroup\_Save', 'UPD', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_ChatMessage\_GetList]

@UserId bigint,

@ChatGroupId int,

@Keyword nvarchar(255),

@PageIndex int,

@PageSize int,

@TotalRecord bigint out,

@Sort nvarchar(100),

@Direction nvarchar(100)

as

begin try

begin

update cm

set IsRead = 1

from cat\_ChatMessage cm

where cm.ChatGroupId = @ChatGroupId and isnull(cm.IsDelete , 0) = 0 and isnull(cm.IsRead, 0) = 0 and isnull(cm.TypeQA, 0) = 0

select

[ChatMessageId] = cct.[ChatMessageId]

,[ChatGroupId] = cct.[ChatGroupId]

,[Message] = cct.[Message]

,[CreateTime] = cct.[CreateTime]

,[IsDelete] = cct.[IsDelete]

,[IsLink] = cct.[IsLink]

,[TypeQA] = cct.[TypeQA]

,[IsRead] = cct.[IsRead]

,[UserId] = cct.[UserId]

,[AccountGuestId] = cct.[AccountGuestId]

from [dbo].[cat\_ChatMessage] cct (nolock)

where

cct.ChatGroupId = @ChatGroupId

and

isnull(cct.IsDelete, 0) = 0

AND

(cct.[Message] like Concat('%',@Keyword,'%') or

isnull(@Keyword,'') = ''

)

order by CreateTime desc

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

select

@TotalRecord = count(cc.[ChatMessageId])

from

[cat\_ChatMessage] cc (nolock)

where

cc.ChatGroupId = @ChatGroupId

and isnull(cc.IsDelete, 0) = 0

AND

( cc.[Message] like Concat('%',@Keyword,'%') or

isnull(@Keyword,'') = ''

)

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_ChatMessage\_GetList: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@ChatGroupId=' + convert(nvarchar, @ChatGroupId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_ChatMessage', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_ChatMessage\_InsertUpdate]

@ChatMessageId int out,

@UserId bigint,

@XMLID int

as

begin try

if @ChatMessageId <= 0

begin

--Insert mode

insert into [cat\_ChatMessage]

(

ChatGroupId,

[UserId],

[Message],

[IsLink],

[CreateTime],

[IsDelete],

TypeQA

)

select

ChatGroupId = x.ChatGroupId,

[UserId] = @UserId,

[Message] = [Message],

[IsLink] = [IsLink],

[CreateTime] = Getdate(),

[IsDelete] = 0,

TypeQA = 0

from openxml(@XMLID, '/ChatMessageRequestModel', 2)

with (

ChatGroupId int,

[Message] nvarchar(max),

[IsLink] bit

) x

set @ChatMessageId = scope\_identity()

end

else

begin

--Update mode

update cct

set [Message] = x.[Message],

[IsLink] = x.[IsLink]

from openxml(@XMLID, '/ChatMessageRequestModel', 2)

with (

[Message] nvarchar(max),

[IsLink] bit

) x

inner join [cat\_ChatMessage] cct (nolock)

on cct.ChatMessageId = @ChatMessageId

where

isnull(cct.[Message],'') <>isnull(x.[Message],'')or

isnull(cct.[IsLink],0) <>isnull(x.[IsLink],0)

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'App\_ChatMessage\_InsertUpdate: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@ChatMessageId=' + convert(nvarchar, @ChatMessageId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_ChatMessage', 'UPD', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_ChatMessage\_Save]

@ChatMessageId int output,

@UserId bigint,

@XML nvarchar(max)

as

begin try

declare @XMLID int

set @ChatMessageId = isnull(@ChatMessageId,0)

set @XML = dbo.ufn\_Replace\_XmlChars(@XML)

exec sp\_xml\_preparedocument @XMLID out, @XML

exec cat\_ChatMessage\_InsertUpdate @ChatMessageId output, @UserId, @XMLID

exec sp\_xml\_removedocument @XMLID

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'App\_ChatMessage\_Save: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@ChatMessageId=' + convert(nvarchar, @ChatMessageId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'App\_ChatMessage\_Save', 'UPD', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Clinic\_ChangeStatus]

@ClinicId int,

@UserId int,

@StatusID int out

as

begin try

set @ClinicId = isnull(@ClinicId, 0)

--set @StatusID = 0

if exists ( select 1

from [cat\_Clinic] r (nolock)

where r.ClinicId = @ClinicId and r.IsDelete = 0

)

begin

declare @a int

set @a = (Select u.[Status] from cat\_Clinic u where u.ClinicId = @ClinicId)

if(@a = 0)

begin

Update r

Set r.Status = 1 ,r.UpdateBy = @UserId,r.UpdateAt = GETDATE()

from cat\_Clinic r (nolock)

where r.ClinicId = @ClinicId

--Select Code = 0, [Message] = 'Status success'

--

end

else

begin

Update r

Set r.Status = 0,r.UpdateBy = @UserId,

r.UpdateAt = GETDATE()

from cat\_Clinic r (nolock)

where r.ClinicId = @ClinicId

--Select Code = 0, [Message] = 'Status success'

end

set @StatusID = 1

end

else

begin

--Select Code = 404, [Message] = 'Recored not found'

set @StatusID = -1

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500),

@SessionID int

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Clinic\_ChangeStatus: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@ClinicId=' + convert(varchar,@ClinicId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Clinic', 'DEL', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Clinic\_CheckExist\_Email]

--Created By: dangnv

--Created Time: 2021/02/24

--Updated By:

--Updated Time:

@Email nvarchar(100),

@Id int,

@IsExists smallint out

AS

begin try

set @IsExists = 0

if(@Id != 0)

begin

if exists ( select 1

from [cat\_Clinic] u

where UPPER(u.Email)=UPPER(@Email) and u.ClinicId != @Id

)

begin

set @IsExists = 1

end

end

else

begin

if exists ( select 1

from [cat\_Clinic] u

where UPPER(u.Email)=UPPER(@Email)

)

begin

set @IsExists = 1

end

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Clinic\_CheckExist\_Email: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Id=' + convert(nvarchar, @Id)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Clinic', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Clinic\_CheckExist\_Name]

--Created By: dangnv

--Created Time: 2021/02/24

--Updated By:

--Updated Time:

@Name nvarchar(100),

@Id int,

@IsExists smallint out

AS

begin try

set @IsExists = 0

if(@Id != 0)

begin

if exists ( select 1

from [cat\_Clinic] u

where UPPER(u.Name)=UPPER(@Name) and u.ClinicId != @Id

)

begin

set @IsExists = 1

end

end

else

begin

if exists ( select 1

from [cat\_Clinic] u

where UPPER(u.Name)=UPPER(@Name)

)

begin

set @IsExists = 1

end

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Clinic\_CheckExist\_Name: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Id=' + convert(nvarchar, @Id)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Clinic', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Clinic\_Delete]

@ClinicId int,

@UserId int,

@StatusID int out

as

begin try

set @ClinicId = isnull(@ClinicId, 0)

--set @StatusID = 0

if exists ( select 1

from [cat\_Clinic] r (nolock)

where r.ClinicId = @ClinicId

)

begin

Update r

Set r.IsDelete = 1 , r.UpdateBy = @UserId , r.UpdateTime = getDate()

from cat\_Clinic r (nolock)

where r.ClinicId = @ClinicId

Select Code = 0, [Message] = 'Delete success'

set @StatusID = 1

end

else

begin

Select Code = 404, [Message] = 'Recored not found'

set @StatusID = -1

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500),

@SessionID int

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Clinic\_Delete: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@ClinicId=' + convert(varchar,@ClinicId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Clinic\_Delete', 'DEL', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Clinic\_Get]

@ClinicId int,

@UserId int

as

begin try

declare @ParentId int =0

set @ParentId = (select Isnull(ParentId,0) from [dbo].[cat\_Clinic] where ClinicId = @ClinicId and Isnull(IsDelete,0)=0)

if(@ParentId=0)

begin

-- liên quan đến view

declare @CheckClinicView int =0

set @CheckClinicView = Isnull((select [ClinicViewId] from [dbo].[cat\_Clinic\_View] ccl (nolock) where [ClinicId] = @ClinicId and [CreateAt] = FORMAT (cast( GETDATE() as datetime) , 'MM/yyyy')),0)

if (@CheckClinicView =0)

begin

INSERT INTO [dbo].[cat\_Clinic\_View] ([ClinicId],[CountView],[CreateAt]) VALUES (@ClinicId, 1,FORMAT (cast( GETDATE() as datetime) , 'MM/yyyy'));

end

if (@CheckClinicView >0)

begin

update [dbo].[cat\_Clinic\_View] set [CountView] =[CountView] +1 where [ClinicViewId] = @CheckClinicView

end

end

-- chi tiết clinic

select

[ClinicId] = r.[ClinicId],

[Name] =Isnull(r.[Name],'{"en":"","vi":""}'),

[Description] = Isnull(r.[Description],'{"en":"","vi":""}'),

[Email] = r.[Email],

[Phone] = r.[Phone],

[OpenTime] = r.[OpenTime],

[CloseTime] = r.[CloseTime],

[Address] = r.[Address],

[IsDelete] = r.[IsDelete],

[CreateBy] = r.[CreateBy],

[CreateAt] = r.[CreateAt],

[UpdateBy] = r.[UpdateBy],

[UpdateAt] = r.[UpdateAt],

[Status] = r.[Status],

[City] = p.[Name],

[District] = d.[Name],

[Avatar] = cm.[Path],

Representative = ccr.[Representative],

Representative2 = ccr.[Representative2],

PhoneUserContact = ccr.[Phone],

PhoneUserContact2 = ccr.[Phone2]

from [dbo].[cat\_Clinic] r (nolock)

left join cat\_Clinic\_Image cm on cm.ClinicId = r.ClinicId and cm.IsActive =1

left join [cat\_Clinic\_Representative] ccr on ccr.ClinicId = r.ClinicId

left join cat\_Province p on r.City =p.ProvinceId

left join cat\_District d on r.District =d.DistrictId

where

isnull(r.IsDelete,0) = 0

and r.ClinicId = @ClinicId

-- danh sách file

select

[ClinicFileId] = r.[ClinicFileId],

[Path] = r.[Path],

[IsDelete] = r.[IsDelete],

[ClinicId] = r.[ClinicId],

[IsActive] = r.[IsActive]

from [dbo].[cat\_Clinic\_Image] r (nolock)

where

isnull(r.IsDelete,0) = 0

and r.ClinicId = @ClinicId

-- danh sach chi nhanh neu cos

select

[ClinicId] = r.[ClinicId],

[Name] =Isnull(r.[Name],'{"en":"","vi":""}'),

[Description] = Isnull(r.[Description],'{"en":"","vi":""}'),

[Email] = r.[Email],

[Phone] = r.[Phone],

[OpenTime] = r.[OpenTime],

[CloseTime] = r.[CloseTime],

[Address] = r.[Address],

[IsDelete] = r.[IsDelete],

[CreateBy] = r.[CreateBy],

[CreateAt] = r.[CreateAt],

[UpdateBy] = r.[UpdateBy],

[UpdateAt] = r.[UpdateAt],

[Status] = r.[Status],

[City] = p.[Name],

[District] = d.[Name],

[Avatar] = cm.[Path]

from [dbo].[cat\_Clinic] r (nolock)

left join cat\_Clinic\_Image cm on cm.ClinicId = r.ClinicId and cm.IsActive =1

left join cat\_Province p on r.City =p.ProvinceId

left join cat\_District d on r.District =d.DistrictId

where

isnull(r.IsDelete,0) = 0

and r.ParentId = @ClinicId

-------------------danh sach san pham khuyen mai -------

select top 5

[ClinicId] = pa.[ClinicId],

[ProductId] = pa.[ProductId],

[HotDealId] = phd.[HotDealId],

[Path] = phi.[Path]

from cat\_Product\_HotDeal phd

left join cat\_Product p on phd.ProductId = p.ProductId

left join cat\_Product\_HotDeal\_Image phi on phi.HotDealId = phd.HotDealId

left join cat\_Product\_Address pa on pa.ProductId = phd.ProductId

where

isnull(phd.IsDelete,0) = 0 and isnull(p.IsDelete,0) = 0 and isnull(pa.IsDelete,0) = 0 and isnull(phi.IsDelete,0) = 0

and phd.IsActive = 1 and p.IsActive=1 and pa.IsActiveSell =1

and pa.ClinicId = @ClinicId

Order by HotDealId asc

ALTER PROCEDURE [dbo].[cat\_Clinic\_GetList\_By\_Search]

@UserId bigint,

@Latitude decimal(18, 15),

@Longitude decimal(18, 15),

@Keyword nvarchar(255),

@PageIndex int,

@PageSize int,

@TotalRecord bigint out,

@Sort nvarchar(100),

@Direction nvarchar(100)

as

begin try

begin

select

[ClinicId] = r.[ClinicId],

[Name] =Isnull(r.[Name],'{"en":"","vi":""}'),

[Length] = (geography::Point(Isnull(@Latitude,0),Isnull(@Longitude,0), 4326).STDistance(geography::Point(Isnull(r.Latitude,0),Isnull(r.Longitude,0), 4326)))

from [dbo].[cat\_Clinic] r (nolock)

left join cat\_Clinic\_Image cm on cm.ClinicId = r.ClinicId and cm.IsActive =1

left join cat\_Province p on r.City =p.ProvinceId

left join cat\_District d on r.District =d.DistrictId

where

isnull(r.IsDelete,0) = 0 AND

( ( dbo.ufn\_removeMark(r.[Name]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

or ( dbo.ufn\_removeMark(r.[ClinicId]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

)

order by

CASE WHEN @Sort = '' AND @Direction = 'asc' THEN r.ClinicId END ASC,

CASE WHEN @Sort = '' AND @Direction = 'desc' THEN r.ClinicId END DESC,

CASE WHEN @Sort = '' AND @Direction = '' THEN r.ClinicId END ASC

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

select

@TotalRecord = count(r.ClinicId)

from [dbo].[cat\_Clinic] r (nolock)

left join cat\_Clinic\_Image cm on cm.ClinicId = r.ClinicId and cm.IsActive =1

left join cat\_Province p on r.City =p.ProvinceId

left join cat\_District d on r.District =d.DistrictId

where

isnull(r.IsDelete,0) = 0 AND

( ( dbo.ufn\_removeMark(r.[Name]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

or ( dbo.ufn\_removeMark(r.[ClinicId]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

)

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Clinic\_GetList\_By\_Search: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Keyword=' + convert(nvarchar, @Keyword) + '@PageIndex=' + convert(nvarchar, @PageIndex) + '@PageSize=' + convert(nvarchar, @PageSize) + '@TotalRecord=' + convert(nvarchar, @TotalRecord)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Clinic\_GetList\_By\_Search', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Clinic\_GetListChild]

@UserId bigint,

@ClinicId int,

@Keyword nvarchar(255),

@PageIndex int,

@PageSize int,

@TotalRecord bigint out,

@Sort nvarchar(100),

@Direction nvarchar(100)

as

begin try

if exists(select 1 from [cat\_Clinic] r

where r.ParentId = @ClinicId and isnull(r.IsDelete, 0) = 0)

begin

select

[ClinicId] = r.ClinicId,

[Name] = Isnull(r.[Name],'{"en":"","vi":""}'),

[Description] = Isnull(r.[Description],'{"en":"","vi":""}'),

[Email] = r.[Email],

[Phone] = r.[Phone],

[OpenTime] = r.[OpenTime],

[CloseTime] = r.[CloseTime],

[Address] = r.[Address],

[IsDelete] = r.[IsDelete],

[CreateBy] = r.[CreateBy],

[CreateAt] = r.[CreateAt],

[UpdateBy] = r.[UpdateBy],

UpdateAt = r.UpdateAt,

[Status] = r.[Status],

[City] = r.[City],

[District] = r.[District],

[ParentId] = r.[ParentId],

[CityName] =p.[Name],

[DistrictName] =d.[Name],

[Avatar] = cm.[Path]

from [dbo].[cat\_Clinic] r (nolock)

left join cat\_Clinic\_Image cm on cm.ClinicId = r.ClinicId and cm.IsActive =1

left join cat\_Province p on r.City =p.ProvinceId

left join cat\_District d on r.District =d.DistrictId

where

(r.[ParentId] = @ClinicId) AND

isnull(r.IsDelete,0) = 0 AND

(

( dbo.ufn\_removeMark(r.[Name]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

)

order by

CASE WHEN @Sort = '' AND @Direction = 'asc' THEN r.ClinicId END ASC,

CASE WHEN @Sort = '' AND @Direction = 'desc' THEN r.ClinicId END DESC,

CASE WHEN @Sort = '' AND @Direction = '' THEN r.ClinicId END ASC

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

select

@TotalRecord = count(r.ClinicId)

from [dbo].[cat\_Clinic] r (nolock)

left join cat\_Clinic\_Image cm on cm.ClinicId = r.ClinicId and cm.IsActive =1

left join cat\_Province p on r.City =p.ProvinceId

left join cat\_District d on r.District =d.DistrictId

where

(r.[ParentId] = @ClinicId) AND

isnull(r.IsDelete,0) = 0 AND

(

( dbo.ufn\_removeMark(r.[Name]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

)

end

else

begin

select

[ClinicId] = r.ClinicId,

[Name] = Isnull(r.[Name],'{"en":"","vi":""}'),

[Description] = Isnull(r.[Description],'{"en":"","vi":""}'),

[Email] = r.[Email],

[Phone] = r.[Phone],

[OpenTime] = r.[OpenTime],

[CloseTime] = r.[CloseTime],

[Address] = r.[Address],

[IsDelete] = r.[IsDelete],

[CreateBy] = r.[CreateBy],

[CreateAt] = r.[CreateAt],

[UpdateBy] = r.[UpdateBy],

UpdateAt = r.UpdateAt,

[Status] = r.[Status],

[City] = r.[City],

[District] = r.[District],

[ParentId] = r.[ParentId],

[CityName] =p.[Name],

[DistrictName] =d.[Name],

[Avatar] = cm.[Path]

from [dbo].[cat\_Clinic] r (nolock)

left join cat\_Clinic\_Image cm on cm.ClinicId = r.ClinicId and cm.IsActive =1

left join cat\_Province p on r.City =p.ProvinceId

left join cat\_District d on r.District =d.DistrictId

where

(r.[ClinicId] = @ClinicId) AND

isnull(r.IsDelete,0) = 0 AND

(

( dbo.ufn\_removeMark(r.[Name]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

)

order by

CASE WHEN @Sort = '' AND @Direction = 'asc' THEN r.ClinicId END ASC,

CASE WHEN @Sort = '' AND @Direction = 'desc' THEN r.ClinicId END DESC,

CASE WHEN @Sort = '' AND @Direction = '' THEN r.ClinicId END ASC

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

select

@TotalRecord = count(r.ClinicId)

from [dbo].[cat\_Clinic] r (nolock)

left join cat\_Clinic\_Image cm on cm.ClinicId = r.ClinicId and cm.IsActive =1

left join cat\_Province p on r.City =p.ProvinceId

left join cat\_District d on r.District =d.DistrictId

where

(r.[ClinicId] = @ClinicId) AND

isnull(r.IsDelete,0) = 0 AND

(

( dbo.ufn\_removeMark(r.[Name]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

)

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Clinic\_GetListChild: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Keyword=' + convert(nvarchar, @Keyword) + '@PageIndex=' + convert(nvarchar, @PageIndex) + '@PageSize=' + convert(nvarchar, @PageSize) + '@TotalRecord=' + convert(nvarchar, @TotalRecord)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Clinic\_GetListChild', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_InstallmentPeriod\_GetList]

@UserId bigint,

@Keyword nvarchar(255),

@PageIndex int,

@PageSize int,

@TotalRecord bigint out,

@Sort nvarchar(100),

@Direction nvarchar(100)

as

begin try

begin

select

[InstallmentPeriodId] = r.[InstallmentPeriodId],

[Name] = Isnull(r.[Name],'{"en":"","vi":""}'),

[Description] = Isnull(r.[Description],'{"en":"","vi":""}'),

[OrderNumber] = r.[OrderNumber],

MinPercent = r.MinPercent

from [dbo].[cat\_InstallmentPeriod] r (nolock)

where

isnull(r.IsDelete,0) = 0 AND

(

( dbo.ufn\_removeMark(r.[Name]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

)

order by

CASE WHEN @Sort = '' AND @Direction = 'asc' THEN r.InstallmentPeriodId END ASC,

CASE WHEN @Sort = '' AND @Direction = 'desc' THEN r.InstallmentPeriodId END DESC,

CASE WHEN @Sort = '' AND @Direction = '' THEN r.InstallmentPeriodId END ASC

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

select

@TotalRecord = count(r.InstallmentPeriodId)

from [dbo].[cat\_InstallmentPeriod] r (nolock)

where

isnull(r.IsDelete,0) = 0 AND

( ( dbo.ufn\_removeMark(r.[Name]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

)

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_InstallmentPeriod\_GetList: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Keyword=' + convert(nvarchar, @Keyword) + '@PageIndex=' + convert(nvarchar, @PageIndex) + '@PageSize=' + convert(nvarchar, @PageSize) + '@TotalRecord=' + convert(nvarchar, @TotalRecord)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_InstallmentPeriod\_GetList', 'GET', null, @AddlInfo

end catch

ALTER procedure [dbo].[cat\_Order\_Detail\_By\_User\_GetList]

@UserId bigint,

@Status int,

@HistoryType int,

@Keyword nvarchar(255),

@PageIndex int,

@PageSize int,

@TotalRecord bigint out,

@Sort nvarchar(100),

@Direction nvarchar(100)

as

begin try

--OrderStatusId Name

--1 {"vi":"Chưa xác nhận","en":"Unconfirmed order"}

--2 {"vi":"Đang tiến hành","en":"In-process"}

--3 {"vi":"Hoàn trả","en":"Return order"}

--4 {"vi":"Hoàn thành","en":"Done"}

--7 {"vi":"Từ chối","en":"Reject"}

--8 {"vi":"Hủy đơn","en":"Cancel"}

--9 {"vi":"Chấp nhận","en":"Accept"}

--10 {"vi":"Chờ xử lý","en":"Pending"}

--"- Chia làm 3 tab:

--+ @HistoryType =1 Current: Các đơn chờ thanh toán, chờ duyệt, chờ nhận/giao,đang yêu cầu hoàn trả hoặc

--chấp nhận hoàn trả, hoàn thành mà chưa trả hết tiền(1,2,3,9,10,4)

--+ @HistoryType =2 Return: các đơn hoàn trả thành công

--+ @HistoryType =3 History: Các đơn hoàn thành thanh toán+ nhận hàng/ đơn hủy/bị hủy" (4,7,8)

-- danh sach cac chi nhanh user da dat hang

select

distinct

ClinicBranchId = cc.ClinicId,

OrderId = ccu.OrderId ,

ClinicBranchName =Isnull(cc.Name,'"en":"","vi":""}'),

ClinicBranchAddress =cc.Address,

ClinicId = cc.ParentId,

ClinicName = Isnull(cc1.Name,'"en":"","vi":""}'),

ClinicAddress =cc.Address,

OrderStatusId = ccu.StatusId,

OrderStatusName = iif(@Status=1 and ccu.StatusId=4,'{"vi":"Đang tiến hành","en":"In-process"}',iif(@Status=2,'{"vi":"Hoàn trả","en":"Return order"}', ods.Name)),

IsTransport =ccu.IsTransport,

Transport = ccu.Transport,

Receiver = ccu.Receiver,

DeliveryAddress = ccu.DeliveryAddress,

DeliveryPhone = ccu.DeliveryPhone

from [dbo].[cat\_Order] ccu

left join cat\_Order\_Status ods on ods.OrderStatusId = ccu.StatusId

left join cat\_Order\_Period cop on cop.OrderId=ccu.OrderId

left join cat\_Clinic cc on cc.ClinicId=ccu.ClinicId

left join cat\_Clinic cc1 on cc1.ClinicId=cc.ParentId

where ccu.CreateBy=@UserId and ISNULL(ccu.IsDelete,0)=0 and cc.ClinicId >0

and(

(@Status =1 and (ccu.StatusId in (1,2,3,9,10) or

-- sử dụng và gửi sp hoàn thành mà chưa thanh toán hết

(ccu.StatusId=4 and isnull(ccu.IsSuccess,0)=0 ) or

-- Hoàn trả thành thông mà chuyển khoản lại cho khách hàng

(ccu.StatusId=4 and isnull(ccu.IsSuccess,0)=0 and ISNULL( ccu.IsRefusal,0)=1 )

))

or (@Status =2 and (ccu.StatusId = 4 and ISNULL( ccu.IsRefusal,0)=1) and isnull(ccu.IsSuccess,0)=1)

or(@Status =3 and (

( @HistoryType =1 and (ccu.StatusId in (7,8) or (ccu.StatusId=4 and isnull(ccu.IsSuccess,0)=0 ))) -- hoặc -- và điều kiện đã thanh toán hết)

or (@HistoryType =2 and (ccu.StatusId=4 and isnull(ccu.IsSuccess,0)=0 ))-- và điều kiện đã thanh toán hết))

or (@HistoryType =3 and (ccu.StatusId in (7,8))

))))

order by ccu.OrderId desc

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

-- danh sách các sản phẩm dc sắp xếp theo thời gian

with CTE (OrderId)as(

select

distinct

OrderId = ccu.OrderId

from [dbo].[cat\_Order] ccu

left join cat\_Order\_Status ods on ods.OrderStatusId = ccu.StatusId

left join cat\_Order\_Period cop on cop.OrderId=ccu.OrderId

left join cat\_Clinic cc on cc.ClinicId=ccu.ClinicId

left join cat\_Clinic cc1 on cc1.ClinicId=cc.ParentId

where ccu.CreateBy=@UserId and ISNULL(ccu.IsDelete,0)=0 and cc.ClinicId >0

and(

(@Status =1 and (ccu.StatusId in (1,2,3,9,10) or

-- sử dụng và gửi sp hoàn thành mà chưa thanh toán hết

(ccu.StatusId=4 and isnull(ccu.IsSuccess,0)=0 ) or

-- Hoàn trả thành thông mà chuyển khoản lại cho khách hàng

(ccu.StatusId=4 and isnull(ccu.IsSuccess,0)=0 and ISNULL( ccu.IsRefusal,0)=1 )

))

or (@Status =2 and (ccu.StatusId = 4 and ISNULL( ccu.IsRefusal,0)=1) and isnull(ccu.IsSuccess,0)=1)

or(@Status =3 and (

( @HistoryType =1 and (ccu.StatusId in (7,8) or (ccu.StatusId=4 and isnull(ccu.IsSuccess,0)=0 ))) -- hoặc -- và điều kiện đã thanh toán hết)

or (@HistoryType =2 and (ccu.StatusId=4 and isnull(ccu.IsSuccess,0)=0 ))-- và điều kiện đã thanh toán hết))

or (@HistoryType =3 and (ccu.StatusId in (7,8))

))))

order by ccu.OrderId desc

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY

)

SELECT

[OrderDetailId] =cod.[OrderDetailId]

,[OrderId] =cod.[OrderId]

,[ProductId] =cod.[ProductId]

,[Name] =Isnull(cp.Name,'"en":"","vi":""}'),

[Price] =cod.[Price],

[Quantily] = cod.Quantily,

ImageUrl =prm.[Path] ,

[PercentSale] = hd.Discount,

TypeReduction = hd.TypeReduction,

[CreateAt] =cod.[CreateAt]

,[Address] = Isnull(br.[Address],'{"en":"","vi":""}')

,[City] =prv.Name

,[District] =dt.Name

,[Ward] =cw.Name,

[ClinicId] = cod.ClinicId,

IsGeneral = cp.IsGeneral

FROM [cat\_Order\_Detail] cod

left join cat\_Order od on od.OrderId = cod.OrderId

left join cat\_clinic cc on cc.[ClinicId]=cod.[ClinicId]

left join cat\_clinic br on br.[ClinicId]=cod.ClinicId

left join cat\_Product cp on cod.ProductId=cp.ProductId

left join cat\_Product\_HotDeal hd on hd.HotDealId = cp.[HotDealsId]

left join cat\_Ward cw on cw.WardId =br.Ward

left join cat\_District dt on dt.DistrictId =br.District

left join cat\_Province prv on prv.ProvinceId=br.City

left join cat\_Product\_Image prm on prm.ProductId = cp.ProductId and Isnull(prm.IsDelete,0)=0 and prm.IsActive =1

where od.CreateBy=@UserId and ISNULL(od.IsDelete,0)=0 and cc.ClinicId >0

and cod.OrderId in (select OrderId from CTE )

ALTER procedure [dbo].[cat\_Order\_Detail\_Status\_Update]

@OrderId int,

-- @ClinicId bigint,

@UserId bigint,

@XMLID int,

@ReturnResault int,

@StatusID int out

as

begin try

set @StatusID =0

-- cancel hủy đơn

if (@ReturnResault =1)

begin

set @StatusID =1 --yêu cầu hủy đơn

UPDATE [cat\_Order]

set

[ReasonId] = z.[ReasonId],

[DescriptionReason] = z.[DescriptionReason],

[StatusId] = 8,

[WaitingClinic] = 1

FROM OPENXML(@XMLID, '/Order', 2)

WITH

(

[ReasonId] int,

[DescriptionReason] nvarchar(max)

) z

INNER JOIN [cat\_Order] r (nolock)

ON r.OrderId = @OrderId

where

isnull(r.[ReasonId],0) <>isnull(z.[ReasonId],0)

or isnull(r.[DescriptionReason],'') <>isnull(z.[DescriptionReason],'')

--update cat\_Order

--set [ReasonId] = @ReasonId,

-- [DescriptionReason] = @DescriptionReason,

-- [StatusId] = 8,

-- [WaitingClinic] = 1

--where cat\_Order.OrderId = @OrderId

end

-- return yêu cầu hoàn trả

if (@ReturnResault=2)

begin

set @StatusID =2 --yêu cầu hoàn trả

UPDATE r

set

[ReasonId] = z.[ReasonId],

[DescriptionReason] = z.[DescriptionReason],

[StatusId] = 3,

IsRefusal = 1,

[WaitingClinic] = 1

FROM OPENXML(@XMLID, '/Order', 2)

WITH

(

[ReasonId] int,

[DescriptionReason] nvarchar(max)

) z

INNER JOIN [cat\_Order] r (nolock)

ON r.OrderId = @OrderId

where

isnull(r.[ReasonId],0) <>isnull(z.[ReasonId],0)

or isnull(r.[DescriptionReason],'') <>isnull(z.[DescriptionReason],'')

-- insert hình ảnh lý do trả nếu có

if exists (select 1

from openxml(@XMLID, '/Order/ListImage/string',0)

With (ImageUrl nvarchar(255) '.') z

where isnull(z.ImageUrl, '') <> ''

)

begin

insert into [dbo].[cat\_Order\_Refusal\_Image]

(

[OrderId]

,[ImageUrl]

,[CreateBy]

,[CreateAt]

)

select

[OrderId] =@OrderId,

[ImageUrl] = z.[ImageUrl],

[CreateBy] = @UserId,

[CreateAt] = GETDATE()

from openxml(@XMLID, '/Order/ListImage/string',0)

With ([ImageUrl] nvarchar(255) '.') z

end

--update cat\_Order

-- set [ReasonId] = @ReasonId,

-- [DescriptionReason] = @DescriptionReason,

-- [StatusId] = 3,

-- IsRefusal=1

-- where cat\_Order.OrderId = @OrderId

end

-- recieve hoàn thành

if (@ReturnResault = 3)

begin

set @StatusID =3 --yêu cầu hoàn trả

update cat\_Order

set

[ReasonId] = 0,

[DescriptionReason] = '',

[StatusId] = 4

where cat\_Order.OrderId = @OrderId

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Order\_InsertUpdate: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@OrderId=' + convert(nvarchar, @OrderId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Order', 'UPD', @SessionID, @AddlInfo

end catch

ALTER procedure [dbo].[cat\_Order\_Detail\_UpdateStatus]

@UserId bigint,

@XMLID int,

@StatusID int out

as

begin try

begin

--Update mode

update ccl

set

@StatusID =@StatusID

from openxml(@XMLID, '/Orderes', 2)

with (

OrderDetailId nvarchar(512)

) x

inner join [cat\_Order\_Detail] ccl (nolock)

on ccl.OrderDetailId = x.OrderDetailId

end

end try

begin catch

set @StatusID=-1

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Order\_Detail\_UpdateStatus: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@StatusID=' + convert(nvarchar, @StatusID)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Order\_Detail', 'UPD', @SessionID, @AddlInfo

end catch

ALTER procedure [dbo].[cat\_Order\_Detail\_UpdateStatus\_Save]

@UserId bigint,

@XML nvarchar(max),

@StatusID int out

as

begin try

declare @XMLID int

set @StatusID = isnull(@StatusID,0)

set @XML = dbo.ufn\_Replace\_XmlChars(@XML)

exec sp\_xml\_preparedocument @XMLID out, @XML

exec cat\_Order\_Detail\_UpdateStatus @UserId, @XMLID,@StatusID out

exec sp\_xml\_removedocument @XMLID

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Order\_Detail\_UpdateStatus\_Save: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@StatusID=' + convert(nvarchar, @StatusID)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Order\_Detail', 'UPD', @SessionID, @AddlInfo

end catch

ALTER procedure [dbo].[cat\_Order\_Get\_App]

@OrderId bigint,

@UserId bigint

as

begin try

declare @IsRefusal bit

--Moipt lấy thông tin ngày dự kiến giao để xem có dc yêu cầu hoàn trả k

-- kiểm tra xem trạng thái các sp trong đơn có hoàn thành rồi hay chưa

declare @count int =(select count(\*)from cat\_Order\_Detail where OrderId=@OrderId and StatusId=3)

if @count=0-- dc xem xét hoàn lại hay k

begin

declare @EstimatedDelivery nvarchar(250) =isnull((select top 1 EstimatedDelivery from cat\_Order\_Detail where OrderId=@OrderId and ISNULL(EstimatedDelivery,0)!=0),'')

if @EstimatedDelivery=''-- chưa sử dụng hoặc chưa có cái nào gửi hàng

set @IsRefusal=1

else -- đã có cái gửi hàng cần xem ngày dự kiến giao

begin

-- số ngày dc hoàn trả

declare @Songayduochoantra int=isnull( (select ccp.ExchangeDate from cat\_Clinic cc

left join cat\_Clinic\_Policy ccp on ccp.ClinicId=cc.ClinicId

left join cat\_Order cd on cd.ClinicId=cc.ClinicId

where cd.OrderId=@OrderId),0)

set @EstimatedDelivery=DATEADD(day, @Songayduochoantra, @EstimatedDelivery)

declare @Countdate int= DATEDIFF(day, @EstimatedDelivery, getdate())

if(@Countdate>=0)

set @IsRefusal=1

else

set @IsRefusal=0

end

end

else

--k hiển thị nút hoàn trả

set @IsRefusal=0

select

distinct

IsRefusal=@IsRefusal,

ClinicBranchId = cc.ClinicId,

ClinicBranchName =Isnull(cc.Name,'"en":"","vi":""}'),

ClinicBranchAddress =cc.Address,

ClinicId = cc.ParentId,

ClinicName = Isnull(cc1.Name,'"en":"","vi":""}'),

ClinicAddress =cc.Address,

[OrderId] = od.[OrderId]

,[CreateBy] = od.[CreateBy]

,[CreateAt] = od.[CreateAt]

,[IsPayFull] = od.[IsPayFull]-- true là thanh toán 1 lần all

,[InstallmentPeriodId] = od.[InstallmentPeriodId]-- kỳ thanh toán

,[Description] = Isnull(ci.[Description],'{"en":"","vi":""}')

,[InstallmentPeriodName] = Isnull(ci.[Name],'{"en":"","vi":""}')

,[Total] = od.[Total]

,[TotalUseCashBack] = od.[TotalUseCashBack]

,[UseCashBack] = od.[UseCashBack]

,[StatusId] = od.[StatusId]

,[FirstInstallment] = od.[FirstInstallment],

PaymentConfirmation = Isnull(od.PaymentConfirmation,0),

ReasonForRefusal = od.ReasonForRefusal

,PaymentDebt = sum(op.AmountPaid)

from [dbo].[cat\_Order] od

left join cat\_Clinic cc on cc.ClinicId=od.ClinicId

left join cat\_Clinic cc1 on cc1.ClinicId=cc.ParentId

left join cat\_InstallmentPeriod ci on ci.InstallmentPeriodId=od.InstallmentPeriodId

left join cat\_Order\_Period op on op.OrderId = @OrderId and Isnull(op.StatustId,0)=0

where od.OrderId = @OrderId

group by cc.ClinicId,

cc.Name,

cc.Address,

cc.ParentId,

cc1.Name,

cc.Address,

od.[OrderId],

od.[CreateBy],

od.[CreateAt],

od.[IsPayFull],-- true là thanh toán 1 lần all

od.[InstallmentPeriodId],-- kỳ thanh toán

ci.[Description],

ci.[Name],

od.[Total],

od.[TotalUseCashBack],

od.[UseCashBack],

od.[StatusId],

od.[FirstInstallment],

op.AmountPaid,od.PaymentConfirmation,od.ReasonForRefusal

-- Thông tin sản phẩm của giỏ hàng

SELECT

[OrderDetailId] =cod.[OrderDetailId]

,[OrderId] =cod.[OrderId]

,[ProductId] =cod.[ProductId]

,[Name] =Isnull(cp.[Name],'{"en":"","vi":""}')

,[Quantily] =cod.[Quantily]

,[Price] =cod.[Price]

,[ClinicId] =cod.[ClinicId]

,[ClinicName] =Isnull(cc.[Name],'{"en":"","vi":""}')

,[CreateAt] =cod.[CreateAt]

,[CreateBy] =cod.[CreateBy]

,[Address] =Isnull(cc.[Address],'{"en":"","vi":""}')

,[City] =p.[Name]

,[District] =d.[Name]

,[Ward] =w.[Name],

ClinicParentId = Isnull(cc.ParentId,0),

StatusId =cod.StatusId,

IsGeneral = cp.IsGeneral,

StatusId = Isnull(cod.StatusId,0),

StatusName = Isnull(ost.Name,'{"en":"","vi":""}')

FROM [cat\_Order\_Detail] cod

left join cat\_clinic cc on cc.[ClinicId]=cod.[ClinicId]

left join cat\_Product cp on cod.ProductId=cp.ProductId

left join cat\_Province p on cc.City =p.ProvinceId

left join cat\_District d on cc.District =d.DistrictId

left join cat\_Ward w on cc.Ward =w.WardId

left join cat\_Order\_Status ost on ost.OrderStatusId= cod.StatusId

where cod.OrderId = @OrderId

-- thông tin kỳ thanh toán

SELECT

[OrderPeriodId] =[OrderPeriodId]

,[OrderId] =[OrderId]

,[PeriodDate] =[PeriodDate]

,[AmountPaid] =[AmountPaid]

,[CreateAt] =[CreateAt]

,[CreateBy] =[CreateBy],

StatustId = Isnull(StatustId,0),

PaymentConfirmation = Isnull(PaymentConfirmation,0)

FROM [cat\_Order\_Period] cop where cop.OrderId = @OrderId

ALTER procedure [dbo].[cat\_Order\_Payment]

@OrderId bigint,

@IsOrder bit,

@ImageUrl nvarchar(255),

@StatusID int out,

@UserId bigint

as

begin try

set @StatusID =0

--@IsOrder = 1 tức là thanh toán cho đợt thanh toán đầu tiên

--@IsOrder = 0 tức là thanh toán cho lần thanh toán định kì

if(@IsOrder =1) -- khi rino xác nhận thì update [PaymentConfirmation]=1 và Status =1 xác nhận là thanh toán xong

begin

update [dbo].[cat\_Order] set PaymentAt=getdate(),ImageUrl = @ImageUrl,StatusId=1,[PaymentConfirmation]=0

where OrderId = @OrderId or ParentId = @OrderId

set @StatusID =1

end

if(@IsOrder =0) -- khi rino xác nhận thì update [PaymentConfirmation]=1 và Status =1 xác nhận là thanh toán xong

begin

update [dbo].[cat\_Order\_Period] set PaymentAt=getdate(),ImageUrl = @ImageUrl,[PaymentConfirmation]=0

where OrderPeriodId = @OrderId

set @StatusID =2

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Order\_Payment: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@OrderId=' + convert(nvarchar, @OrderId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Order\_Payment', 'GET', null, @AddlInfo

end catch

ALTER procedure [dbo].[cat\_Order\_ViewOrderByUser]

@Type int,

@UserId bigint,

@Keyword nvarchar(255),

@PageIndex int,

@PageSize int,

@TotalRecord bigint out,

@Sort nvarchar(100),

@Direction nvarchar(100)

as

begin try

--select template ids of cat\_Order

select coo.OrderId

into #id\_orders

from cat\_Order coo

where coo.CreateBy = @UserId and isnull(coo.IsDelete, 0) = 0

and isnull(coo.ParentId, 0) <> 0

and ((coo.StatusId in (1,2,9,10) and @Type = 1) -- current -- status in table cat\_Order\_Status

or (coo.StatusId = 4 and isnull(coo.IsRefusal, 0) = 1 and @Type = 2) -- returned

or (coo.StatusId in (4,7,8) and @Type = 3) -- history

)

select

[ClinicId] = cod.[ClinicId]

from [dbo].[cat\_Order\_Detail] cod (nolock)

right join #id\_orders ids on ids.OrderId = cod.OrderId

--order by

-- CASE WHEN @Sort = 'OrderId' AND @Direction = 'asc' THEN coo.OrderId END ASC,

-- CASE WHEN @Sort = 'OrderId' AND @Direction = 'desc' THEN coo.OrderId END DESC,

-- CASE WHEN @Sort = '' AND @Direction = 'asc' THEN coo.OrderId END ASC,

-- CASE WHEN @Sort = '' AND @Direction = 'desc' THEN coo.OrderId END DESC,

-- CASE WHEN @Sort = '' AND @Direction = '' THEN coo.OrderId END DESC

--offset @PageSize \* (@PageIndex - 1) ROWS

--fetch next @PageSize ROWS ONLY;

ALTER PROCEDURE [dbo].[cat\_Rep\_ChatMessage\_InsertUpdate]

@ChatMessageId int out,

@UserId bigint,

@XMLID int

as

begin try

if @ChatMessageId <= 0

begin

--Insert mode

insert into [cat\_ChatMessage]

(

ChatGroupId,

[UserId],

[Message],

[IsLink],

[CreateTime],

[IsDelete],

[TypeQA]

)

select

ChatGroupId = x.ChatGroupId,

[UserId] = @UserId,

[Message] = [Message],

[IsLink] = [IsLink],

[CreateTime] = Getdate(),

[IsDelete] = 0,

[TypeQA] = 1

from openxml(@XMLID, '/RepChatMessageRequestModel', 2)

with (

ChatGroupId int,

[Message] nvarchar(max),

[IsLink] bit

) x

set @ChatMessageId = scope\_identity()

end

else

begin

--Update mode

update cct

set [Message] = x.[Message],

[IsLink] = x.[IsLink]

from openxml(@XMLID, '/RepChatMessageRequestModel', 2)

with (

[Message] nvarchar(max),

[IsLink] bit

) x

inner join [cat\_ChatMessage] cct (nolock)

on cct.ChatMessageId = @ChatMessageId

where

isnull(cct.[Message],'') <>isnull(x.[Message],'')or

isnull(cct.[IsLink],0) <>isnull(x.[IsLink],0)

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Rep\_ChatMessage\_InsertUpdate: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@ChatMessageId=' + convert(nvarchar, @ChatMessageId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_ChatMessage', 'UPD', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Rep\_ChatMessage\_Save]

@ChatMessageId int output,

@UserId bigint,

@XML nvarchar(max)

as

begin try

declare @XMLID int

set @ChatMessageId = isnull(@ChatMessageId,0)

set @XML = dbo.ufn\_Replace\_XmlChars(@XML)

exec sp\_xml\_preparedocument @XMLID out, @XML

exec cat\_Rep\_ChatMessage\_InsertUpdate @ChatMessageId output, @UserId, @XMLID

exec sp\_xml\_removedocument @XMLID

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'App\_ChatMessage\_Save: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@ChatMessageId=' + convert(nvarchar, @ChatMessageId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'App\_ChatMessage\_Save', 'UPD', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Search\_Delete]

@ClinicId int,

@UserId int,

@StatusID int out

as

begin try

set @ClinicId = isnull(@ClinicId, 0)

--set @StatusID = 0

if exists ( select 1

from [cat\_Clinic] r (nolock)

where r.ClinicId = @ClinicId

)

begin

Update r

Set r.IsDelete = 1 , r.UserId = @UserId

from cat\_Search r (nolock)

where r.ClinicId = @ClinicId and UserId =@UserId

Select Code = 0, [Message] = 'Delete success'

set @StatusID = 1

end

else

begin

Select Code = 404, [Message] = 'Recored not found'

set @StatusID = -1

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500),

@SessionID int

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Search\_Delete: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@ClinicId=' + convert(varchar,@ClinicId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Search\_Delete', 'DEL', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Search\_Detail\_Search]

@Latitude decimal(18, 15),

@Longitude decimal(18, 15),

@ClinicId int

as

begin try

-- chi tiết clinic

select

[ClinicId] = r.[ClinicId],

[Name] =Isnull(r.[Name],'{"en":"","vi":""}'),

[Description] = Isnull(r.[Description],'{"en":"","vi":""}'),

[Address] = r.[Address],

[Latitude] = r.[Latitude],

[Longitude] = r.[Longitude],

[UserId] = r.[UserId],

[CityName] = p.[Name],

[DistrictName] = d.[Name],

[WardName] = w.[Name],

[Avatar] = cm.[Path],

[Representative1] = ccr.[Representative],

[Phone1] = ccr.[Phone],

[Length] =round(((geography::Point(Isnull(@Latitude,0),Isnull(@Longitude,0), 4326).STDistance(geography::Point(Isnull(r.Latitude,0),Isnull(r.Longitude,0), 4326)))/1000),1)

from [dbo].[cat\_Clinic] r (nolock)

left join cat\_Clinic\_Image cm on cm.ClinicId = r.ClinicId and cm.IsActive =1

left join [cat\_Clinic\_Representative] ccr on ccr.ClinicId = r.ClinicId

left join cat\_Province p on r.City =p.ProvinceId

left join cat\_District d on r.District =d.DistrictId

left join cat\_Ward w on r.Ward = w.WardId

where

isnull(r.IsDelete,0) = 0

and r.ClinicId = @ClinicId

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Search\_Detail\_Search: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@ClinicId=' + convert(nvarchar, @ClinicId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Search\_Detail\_Search', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Search\_GetListPlace]

@Latitude decimal(18, 15),

@Longitude decimal(18, 15),

@UserId bigint,

@Keyword nvarchar(255),

@PageIndex int,

@PageSize int,

@TotalRecord bigint out,

@Sort nvarchar(100),

@Direction nvarchar(100)

as

begin try

begin

select

[ClinicId] = r.[ClinicId],

[UserId] = r.[UserId],

[CreateAt] = r.[CreateAt],

[Name] = Isnull(r.[Name],'{"en":"","vi":""}'),

[Address] = r.[Address],

[Latitude] = r.[Latitude],

[Longitude] = r.[Longitude],

[CityName] = pr.[Name],

[DistrictName] = dt.[Name],

[WardName] = wa.[Name],

[Representative]= cr.[Representative],

[Phone] = cr.[Phone],

[Length] = round(((geography::Point(Isnull(@Latitude,0),Isnull(@Longitude,0), 4326).STDistance(geography::Point(Isnull(r.Latitude,0),Isnull(r.Longitude,0), 4326)))/1000),1)

from [dbo].[cat\_Clinic] r (nolock)

left join cat\_Clinic\_Representative cr on cr.ClinicId =r.ClinicId

left join cat\_District dt on dt.DistrictId = r.District

left join cat\_Province pr on pr.ProvinceId = r.City

left join cat\_Ward wa on wa.WardId = r.Ward

where

isnull(r.IsDelete,0) = 0 AND

( ( dbo.ufn\_removeMark(r.[Name]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

or ( dbo.ufn\_removeMark(dt.[Name]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

or ( dbo.ufn\_removeMark(pr.[Name]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

or ( dbo.ufn\_removeMark(wa.[Name]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

or ( dbo.ufn\_removeMark(r.[Address]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

)

order by

CASE WHEN @Sort = '' AND @Direction = 'asc' THEN (geography::Point(@Latitude, @Longitude, 4326).STDistance(geography::Point(r.Latitude, r.Longitude, 4326))) END ASC,

CASE WHEN @Sort = '' AND @Direction = 'desc' THEN (geography::Point(@Latitude, @Longitude, 4326).STDistance(geography::Point(r.Latitude, r.Longitude, 4326))) END DESC,

CASE WHEN @Sort = 'time' AND @Direction = 'asc' THEN r.CreateAt END ASC,

CASE WHEN @Sort = 'time' AND @Direction = 'desc' THEN r.CreateAt END DESC,

CASE WHEN @Sort = '' AND @Direction = '' THEN r.ClinicId END ASC

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

select

@TotalRecord = count(r.ClinicId)

from [dbo].[cat\_Clinic] r (nolock)

left join cat\_Clinic\_Representative cr on cr.ClinicId =r.ClinicId

left join cat\_District dt on dt.DistrictId = r.District

left join cat\_Province pr on pr.ProvinceId = r.City

left join cat\_Ward wa on wa.WardId = r.Ward

where

isnull(r.IsDelete,0) = 0 AND

( ( dbo.ufn\_removeMark(r.[Name]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

or ( dbo.ufn\_removeMark(dt.[Name]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

or ( dbo.ufn\_removeMark(pr.[Name]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

or ( dbo.ufn\_removeMark(wa.[Name]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

or ( dbo.ufn\_removeMark(r.[Address]) like Concat('%',dbo.ufn\_removeMark(@Keyword),'%'))

)

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Search\_GetListPlace: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Keyword=' + convert(nvarchar, @Keyword) + '@PageIndex=' + convert(nvarchar, @PageIndex) + '@PageSize=' + convert(nvarchar, @PageSize) + '@TotalRecord=' + convert(nvarchar, @TotalRecord)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Clinic\_Search\_GetListPlace', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_Search\_Insert]

@ClinicId int,

@UserId int,

@Status int out

as

begin try

set @Status = 0

--Kiem tra SearchId co ton tai hay chua

set @ClinicId = isnull(@ClinicId, 0)

declare @CheckSearch int

set @CheckSearch = (

select Count(ClinicId)

from [dbo].[cat\_Search] ns

where ns.[ClinicId] = @ClinicId

and ns.UserId = @UserId

)

-- chia truong hop

if(@CheckSearch=0)

begin

insert into [dbo].[cat\_Search]([ClinicId],[UserId],[IsDelete],[CreateAt])

values(@ClinicId,@UserId,0,getdate())

set @Status = 1

end

else

begin

update cat\_Search set CreateAt =getdate()

where UserId =@UserId and ClinicId =@ClinicId

set @Status = 2

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Search\_Insert: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@ClinicId=' + convert(nvarchar, @ClinicId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Search\_Insert', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_User\_Device]

@Lang nvarchar(50),

@ClinicId bigint

as

begin try

begin

SELECT distinct

[UserId] =x.UserID,

[DeviceToken] =x.[DeviceToken],

[DeviceType] =x.[DeviceType],

[Lang] =x.[Lang]

from sec\_User\_Device x

left join cat\_Clinic cl on cl.UserId = x.UserID

where (isnull(@Lang, '') = '' or x.Lang = @Lang)

and (cl.ClinicId =@ClinicId )

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_User\_Device: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Lang=' + convert(nvarchar, @Lang)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'sec\_User\_Device', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_User\_Device\_AccountGuest\_GetAll]

@Lang nvarchar(50),

@ListClinicId nvarchar(max)

as

begin try

begin

SELECT distinct

[UserId] =x.UserID,

[DeviceToken] =x.[DeviceToken],

[DeviceType] =x.[DeviceType],

[Lang] =x.[Lang]

from sec\_User\_Device x

left join cat\_Clinic cl on cl.UserId = x.UserID

where (isnull(@Lang, '') = '' or x.Lang = @Lang)

and cl.ClinicId IN (SELECT id FROM convert\_string\_to\_table (@ListClinicId, ',') )

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_User\_Device\_AccountGuest\_GetAll: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Lang=' + convert(nvarchar, @Lang)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'sec\_User\_Device', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_User\_Device\_Admin\_Rino\_App]

@Lang nvarchar(50)

as

begin try

begin

SELECT distinct

[UserId] =x.UserID,

[DeviceToken] =x.[DeviceToken],

[DeviceType] =x.[DeviceType],

[Lang] =x.[Lang]

from sec\_User\_Device x

left join sec\_User\_Roles ur on ur.UserId = x.UserID

where (isnull(@Lang, '') = '' or x.Lang = @Lang)

and ur.RoleId =3

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_User\_Device\_Admin\_Rino\_App: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Lang=' + convert(nvarchar, @Lang)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_User\_Device\_Admin\_Rino\_App', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_User\_Device\_Order]

@Lang nvarchar(50),

@OrderId bigint

as

begin try

begin

SELECT distinct

[UserId] =x.UserID,

[DeviceToken] =x.[DeviceToken],

[DeviceType] =x.[DeviceType],

[Lang] =x.[Lang]

from sec\_User\_Device x

left join cat\_Clinic cl on cl.UserId = x.UserID

left join cat\_Order o on o.ClinicId = cl.ClinicId

where (isnull(@Lang, '') = '' or x.Lang = @Lang)

and o.OrderId =@OrderId

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_User\_Device\_Order: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Lang=' + convert(nvarchar, @Lang)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'sec\_User\_Device', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_User\_Device\_Update]

@Lang nvarchar(50),

@ClinicId bigint

as

begin try

begin

SELECT distinct

[UserId] =x.UserID,

[DeviceToken] =x.[DeviceToken],

[DeviceType] =x.[DeviceType],

[Lang] =x.[Lang]

from sec\_User\_Device x

left join sec\_User\_Roles ur on ur.UserId = x.UserID

where (isnull(@Lang, '') = '' or x.Lang = @Lang) and ur.RoleId =3 -- RoleId = 3 la Admnin

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_User\_Device\_Update: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Lang=' + convert(nvarchar, @Lang)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_User\_Device\_Update', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cat\_User\_Device\_User]

@Lang nvarchar(50),

@UserId bigint

as

begin try

begin

SELECT distinct

[UserId] =x.UserID,

[DeviceToken] =x.[DeviceToken],

[DeviceType] =x.[DeviceType],

[Lang] =x.[Lang]

from sec\_User\_Device x

left join sec\_User u on u.UserId = x.UserID

where (isnull(@Lang, '') = '' or x.Lang = @Lang)

and (u.UserId =@UserId )

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_User\_Device\_User: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Lang=' + convert(nvarchar, @Lang)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_User\_Device\_User', 'GET', null, @AddlInfo

end catch

ALTER procedure [dbo].[cms\_cat\_Clinic\_Report\_View\_By\_Parent]

@UserId int,

@Year int

as

begin try

declare @ClinicId int

set @ClinicId= (select ClinicId from cat\_Clinic where UserId = @UserId )

SELECT

[ClinicId] = cv.[ClinicId],

[CreateAt] = cv.[CreateAt],

[Name] = cln.Name,

[CountView] = cv.[CountView]

FROM [dev\_Rino].[dbo].[cat\_Clinic\_View] cv

left join [dbo].[cat\_Clinic] cln on cln.ClinicId = cv.ClinicId

where YEAR ('01/'+cv.[CreateAt]) = @Year

and cln.ClinicId = @ClinicId or ParentId = @ClinicId and ISNULL(IsDelete,0)=0

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cms\_cat\_Clinic\_Report\_View\_By\_Parent: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = ''

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cms\_cat\_Clinic\_Report\_View\_By\_Parent', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cms\_cat\_Clinic\_Rino\_Report]

@Type bigint,

@Year int

as

begin try

if @Type=1-- nếu là tháng

begin

;WITH months(MonthNumber) AS

(

SELECT 1

UNION ALL

SELECT MonthNumber+1

FROM months

WHERE MonthNumber < 12

)

select m.MonthNumber, count( distinct cc.UserId) from months m

left join cat\_Clinic cc on MONTH(cc.CreateAt)=m.MonthNumber

left join sec\_User\_Roles sur on sur.UserId=cc.UserId and RoleId=2

where ISNULL(ParentId,0)=0

and isnull(RinoApproval,0)=1

and Year(CreateAt)=@Year

and RoleId=2

group by m.MonthNumber

end

else

begin

;WITH months(MonthNumber,id) AS

(

SELECT 1,1

UNION ALL

SELECT MonthNumber+1 , iif(MonthNumber+1<=3,1 , iif(MonthNumber+ 1 <=6,2,iif(MonthNumber+ 1<=9,3,4)))

FROM months

WHERE MonthNumber < 12

)

select m.id, count(distinct sur.UserId) from months m

left join cat\_Clinic cc on MONTH(cc.CreateAt)=m.MonthNumber

left join sec\_User\_Roles sur on sur.UserId=cc.UserId and RoleId=2

where ISNULL(ParentId,00)=0

and isnull(RinoApproval,0)=1

and Year(CreateAt)=@Year

and RoleId=2

group by m.id

end

ALTER PROCEDURE [dbo].[cms\_cat\_customer\_Rino\_Report]

@Type bigint,

@Year int

as

begin try

if @Type=1-- nếu là tháng

begin

;WITH months(MonthNumber) AS

(

SELECT 1

UNION ALL

SELECT MonthNumber+1

FROM months

WHERE MonthNumber < 12

)

select m.MonthNumber, count( distinct cc.UserId) from months m

left join cat\_Clinic cc on MONTH(cc.CreateAt)=m.MonthNumber

left join sec\_User\_Roles sur on sur.UserId=cc.UserId and RoleId=1

where ISNULL(ParentId,00)=0

--and isnull(RinoApproval,0)=1

and Year(CreateAt)=@Year

and RoleId=1

group by m.MonthNumber

end

else

begin

;WITH months(MonthNumber,id) AS

(

SELECT 1,1

UNION ALL

SELECT MonthNumber+1 , iif(MonthNumber+1<=3,1 , iif(MonthNumber+ 1 <=6,2,iif(MonthNumber+ 1<=9,3,4)))

FROM months

WHERE MonthNumber < 12

)

select m.id, count(distinct sur.UserId) from months m

left join cat\_Clinic cc on MONTH(cc.CreateAt)=m.MonthNumber

left join sec\_User\_Roles sur on sur.UserId=cc.UserId and RoleId=1

where ISNULL(ParentId,00)=0

--and isnull(RinoApproval,0)=1

and Year(CreateAt)=@Year

and RoleId=1

group by m.id

end

ALTER PROCEDURE [dbo].[cms\_cat\_Debt\_Export]

@UserId bigint

as

begin try

declare @ClinicId bigint = isnull((select ClinicId from cat\_Clinic where UserId = @UserId and isnull(IsDelete, 0) = 0), 0);

select

--[DebtId] = d.[DebtId],

--[ClinicId] = d.[ClinicId],

[Date] = d.[Date],

[OldDebt] = d.[OldDebt],

[Revenue] = d.[Revenue],

[Expense] = d.[Expense],

[Paid] = d.[Paid],

[NewDebt] = d.[NewDebt]

from [dbo].[cat\_Debt] d (nolock)

where d.ClinicId = @ClinicId

union

select

--[DebtId] = NULL,

--[ClinicId] = @ClinicId,

[Date] = N'Tổng',

[OldDebt] = (select top (1) OldDebt from cat\_Debt where ClinicId = @ClinicId),

[Revenue] = (select sum(Revenue) from cat\_Debt where ClinicId = @ClinicId),

[Expense] = (select sum(Expense) from cat\_Debt where ClinicId = @ClinicId),

[Paid] = (select sum(Paid) from cat\_Debt where ClinicId = @ClinicId),

[NewDebt] = (select top (1) NewDebt from cat\_Debt where ClinicId = @ClinicId order by DebtId desc)

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cms\_cat\_Debt\_Export: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@UserId=' + convert(nvarchar, @UserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Debt', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cms\_cat\_Debt\_GetList]

@UserId bigint,

@DateFrom nvarchar(255),

@DateTo nvarchar(255),

@Type int,

@Keyword nvarchar(255),

@PageIndex int,

@PageSize int,

@TotalRecord bigint out,

@Sort nvarchar(100),

@Direction nvarchar(100)

as

begin try

declare @ClinicId bigint = isnull((select ClinicId from cat\_Clinic where UserId = @UserId and isnull(IsDelete, 0) = 0), 0);

declare @OldDebt bigint = (select top (1) OldDebt from cat\_Debt where ClinicId = @ClinicId and ((isnull(@DateFrom,'') = '' or isnull(@DateTo,'') = '')

or convert(datetime, [Date], 103) between convert(datetime, @DateFrom, 103) and convert(datetime, @DateTo, 103)));

declare @NewDebt bigint = (select top (1) NewDebt from cat\_Debt where ClinicId = @ClinicId and ((isnull(@DateFrom,'') = '' or isnull(@DateTo,'') = '')

or convert(datetime, [Date], 103) between convert(datetime, @DateFrom, 103) and convert(datetime, @DateTo, 103))

order by DebtId desc);

if(@Type = 1) --day

begin

select

[DebtId] = d.[DebtId],

[ClinicId] = d.[ClinicId],

[Date] = d.[Date],

[OldDebt] = d.[OldDebt],

[Revenue] = d.[Revenue],

[Expense] = d.[Expense],

[Paid] = d.[Paid],

[NewDebt] = d.[NewDebt]

from [dbo].[cat\_Debt] d (nolock)

where d.ClinicId = @ClinicId

and ((isnull(@DateFrom,'') = '' or isnull(@DateTo,'') = '')

or convert(datetime, d.[Date], 103) between convert(datetime, @DateFrom, 103) and convert(datetime, @DateTo, 103))

order by

CASE WHEN @Sort = '' AND @Direction = '' THEN d.DebtId END desc

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

end

if(@Type = 2) --month

begin

select

tab.[Date],

tab.[OldDebt],

tab.[Revenue],

tab.[Expense],

tab.[Paid],

tab.[NewDebt]

from

(select

--[DebtId] = d.[DebtId],

--[ClinicId] = d.[ClinicId],

[Date] = convert(varchar(20), DATEADD(MONTH, DATEDIFF(MONTH, 0, convert(datetime, d.[Date], 103) ), 0), 103),

[OldDebt] = @OldDebt,

[Revenue] = sum(d.[Revenue]),

[Expense] = sum(d.[Expense]),

[Paid] = sum(d.[Paid]),

[NewDebt] = @NewDebt

from [dbo].[cat\_Debt] d (nolock)

where d.ClinicId = @ClinicId

and ((isnull(@DateFrom,'') = '' or isnull(@DateTo,'') = '')

or convert(datetime, d.[Date], 103) between convert(datetime, @DateFrom, 103) and convert(datetime, @DateTo, 103))

group by DATEADD(MONTH, DATEDIFF(MONTH, 0, convert(datetime, d.[Date], 103) ), 0)

) tab

order by

CASE WHEN @Sort = '' AND @Direction = '' THEN tab.[Date] END desc

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

end

if(@Type = 3) --year

begin

select

tab.[Date],

tab.[OldDebt],

tab.[Revenue],

tab.[Expense],

tab.[Paid],

tab.[NewDebt]

from

(select

--[DebtId] = d.[DebtId],

--[ClinicId] = d.[ClinicId],

[Date] = convert(varchar(20), DATEADD(YEAR, DATEDIFF(YEAR, 0, convert(datetime, d.[Date], 103) ), 0), 103),

[OldDebt] = @OldDebt,

[Revenue] = sum(d.[Revenue]),

[Expense] = sum(d.[Expense]),

[Paid] = sum(d.[Paid]),

[NewDebt] = @NewDebt

from [dbo].[cat\_Debt] d (nolock)

where d.ClinicId = @ClinicId

and ((isnull(@DateFrom,'') = '' or isnull(@DateTo,'') = '')

or convert(datetime, d.[Date], 103) between convert(datetime, @DateFrom, 103) and convert(datetime, @DateTo, 103))

group by DATEADD(YEAR, DATEDIFF(YEAR, 0, convert(datetime, d.[Date], 103) ), 0)

) tab

order by

CASE WHEN @Sort = '' AND @Direction = '' THEN tab.[Date] END desc

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

end

--Sum

select

[ClinicId] = @ClinicId,

[OldDebt] = @OldDebt,

[Revenue] = sum(d.Revenue),

[Expense] = sum(d.Expense),

[Paid] = sum(d.Paid),

[NewDebt] = @NewDebt

from [dbo].[cat\_Debt] d (nolock)

where d.ClinicId = @ClinicId

and ((isnull(@DateFrom,'') = '' or isnull(@DateTo,'') = '')

or convert(datetime, d.[Date], 103) between convert(datetime, @DateFrom, 103) and convert(datetime, @DateTo, 103))

select

@TotalRecord = count(d.DebtId)

from [dbo].[cat\_Debt] d (nolock)

where d.ClinicId = @ClinicId

and ((isnull(@DateFrom,'') = '' or isnull(@DateTo,'') = '')

or convert(datetime, d.[Date], 103) between convert(datetime, @DateFrom, 103) and convert(datetime, @DateTo, 103))

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cms\_cat\_Debt\_GetList: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Keyword=' + convert(nvarchar, @Keyword) + '@PageIndex=' + convert(nvarchar, @PageIndex) + '@PageSize=' + convert(nvarchar, @PageSize) + '@TotalRecord=' + convert(nvarchar, @TotalRecord)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Debt', 'GET', null, @AddlInfo

end catch

ALTER procedure [dbo].[cms\_cat\_Order\_CountView\_Branch]

@UserId bigint,

@Year int

as

begin try

declare @ListMonth nvarchar(200) = '1,2,3,4,5,6,7,8,9,10,11,12'

begin

SELECT

[Month] = th.id

FROM convert\_string\_to\_table (@ListMonth, ',') th

SELECT

[Month] = MONTH ( TRY\_PARSE('01/'+ cv.CreateAt AS DATETIME USING 'en-gb')),

[Total] = cv.[CountView]

FROM cat\_Clinic\_View cv

left join cat\_Clinic cl on cl.ClinicId= cv.ClinicId

where cl.UserId= @UserId and YEAR ( TRY\_PARSE('01/'+ cv.CreateAt AS DATETIME USING 'en-gb'))=@Year

--left join (SELECT id FROM convert\_string\_to\_table (@Year, ',')) br on br.id = br.id

--left join cat\_Clinic\_View cv on YEAR ( TRY\_PARSE('01/'+ cv.CreateAt AS DATETIME USING 'en-gb'))=@Year and MONTH ( TRY\_PARSE('01/'+ cv.CreateAt AS DATETIME USING 'en-gb'))=th.id

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cms\_cat\_Order\_CountView\_Branch: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@UserId=' + convert(nvarchar, @UserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cms\_cat\_Order\_CountView\_Branch', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cms\_cat\_Order\_Detail\_Turnover\_By\_Time]

@UserId bigint,

@Type int,

@ClinicId int,

@TimeStart nvarchar(255),

@TimeEnd nvarchar(255),

@Keyword nvarchar(255),

@PageIndex int,

@PageSize int,

@TotalRecord bigint out,

@Sort nvarchar(100),

@Direction nvarchar(100)

as

begin try

begin

declare @ClinicParentId int= 0

set @ClinicParentId = (select ClinicId from [dbo].[cat\_Clinic] where UserId = @UserId and ISnull(IsDelete,0)=0 and Isnull(ParentId,0)=0)

if(@Type=1)

begin

SELECT

[ClinicId] = odt.[ClinicId],

[Name] = ccc.Name,

[CreateAt] = FORMAT (cast(odt.[CreateAt] as datetime) , 'dd/MM/yyyy')

FROM [dev\_Rino].[dbo].[cat\_Order\_Detail] odt

left join cat\_Clinic ccc on ccc.ClinicId=odt.ClinicId

left join cat\_Order co on co.OrderId=odt.OrderId and ISNULL(co.DateUse,0)!=0

where ccc.ParentId=26

and (FORMAT (cast(odt.[CreateAt] as datetime) , 'dd/MM/yyyy') BETWEEN @TimeStart AND @TimeEnd)

and (@ClinicId=0 or Isnull(ccc.ClinicId,0) = @ClinicId)

group by odt.[ClinicId],

ccc.Name,

FORMAT (cast(odt.[CreateAt] as datetime) , 'dd/MM/yyyy')

order by

CASE WHEN @Sort = '' AND @Direction = 'asc' THEN FORMAT (cast(odt.[CreateAt] as datetime) , 'dd/MM/yyyy') END ASC,

CASE WHEN @Sort = '' AND @Direction = 'desc' THEN FORMAT (cast(odt.[CreateAt] as datetime) , 'dd/MM/yyyy') END Desc,

CASE WHEN @Sort = '' AND @Direction = '' THEN FORMAT (cast(odt.[CreateAt] as datetime) , 'dd/MM/yyyy') END Desc

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

select

@TotalRecord = count(odt.[ClinicId])

FROM [dev\_Rino].[dbo].[cat\_Order\_Detail] odt

left join cat\_Clinic ccc on ccc.ClinicId=odt.ClinicId

left join cat\_Product\_Address cd on cd.ProductId=odt.ProductId

left join cat\_Order co on co.OrderId=odt.OrderId and ISNULL(co.DateUse,0)!=0

where ccc.ParentId=@ClinicParentId

and (FORMAT (cast(odt.[CreateAt] as datetime) , 'dd/MM/yyyy') BETWEEN @TimeStart AND @TimeEnd)

and (@ClinicId=0 or Isnull(ccc.ClinicId,0) = @ClinicId)

group by odt.[ClinicId],

ccc.Name,

FORMAT (cast(odt.[CreateAt] as datetime) , 'dd/MM/yyyy')

SELECT

[ClinicId] = odt.[ClinicId],

[CreateAt] = FORMAT (cast(odt.[CreateAt] as datetime) , 'dd/MM/yyyy'),

[Turnover] = Isnull((odt.Price\*Isnull(SUM(Distinct odt.Quantily),0)) - Isnull(sum(odt.[Discount]),0),0)

FROM [dev\_Rino].[dbo].[cat\_Order\_Detail] odt

left join cat\_Clinic ccc on ccc.ClinicId=odt.ClinicId

left join cat\_Product\_Address cd on cd.ProductId=odt.ProductId

left join cat\_Order co on co.OrderId=odt.OrderId and ISNULL(co.DateUse,0)!=0

where ccc.ParentId=@ClinicParentId

and (FORMAT (cast(odt.[CreateAt] as datetime) , 'dd/MM/yyyy') BETWEEN @TimeStart AND @TimeEnd)

group by odt.[ProductId]

,odt.[Quantily]

,odt.[Price]

,odt.[ClinicId]

,FORMAT (cast(odt.[CreateAt] as datetime) , 'dd/MM/yyyy') ,odt.[Discount],odt.[CreateAt]

order by odt.CreateAt ASC

end

if (@Type =2)

begin

SELECT

[ClinicId] = odt.[ClinicId],

[Name] = ccc.Name,

[CreateAt] = FORMAT (cast(odt.[CreateAt] as datetime) , 'MM/yyyy')

FROM [dev\_Rino].[dbo].[cat\_Order\_Detail] odt

left join cat\_Clinic ccc on ccc.ClinicId=odt.ClinicId

left join cat\_Product\_Address cd on cd.ProductId=odt.ProductId

left join cat\_Order co on co.OrderId=odt.OrderId and ISNULL(co.DateUse,0)!=0

where ccc.ParentId=@ClinicParentId

and (FORMAT (cast(odt.[CreateAt] as datetime) , 'MM/yyyy') BETWEEN @TimeStart AND @TimeEnd)

and (@ClinicId=0 or Isnull(ccc.ClinicId,0) = @ClinicId)

group by odt.[ClinicId],

ccc.Name,

FORMAT (cast(odt.[CreateAt] as datetime) , 'MM/yyyy')

order by

CASE WHEN @Sort = '' AND @Direction = 'asc' THEN FORMAT (cast(odt.[CreateAt] as datetime) , 'MM/yyyy') END ASC,

CASE WHEN @Sort = '' AND @Direction = 'desc' THEN FORMAT (cast(odt.[CreateAt] as datetime) , 'MM/yyyy') END Desc,

CASE WHEN @Sort = '' AND @Direction = '' THEN FORMAT (cast(odt.[CreateAt] as datetime) , 'MM/yyyy') END Desc

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

select

@TotalRecord = count(odt.[ClinicId])

FROM [dev\_Rino].[dbo].[cat\_Order\_Detail] odt

left join cat\_Clinic ccc on ccc.ClinicId=odt.ClinicId

left join cat\_Product\_Address cd on cd.ProductId=odt.ProductId

left join cat\_Order co on co.OrderId=odt.OrderId and ISNULL(co.DateUse,0)!=0

where ccc.ParentId=@ClinicParentId

and (FORMAT (cast(odt.[CreateAt] as datetime) , 'MM/yyyy') BETWEEN @TimeStart AND @TimeEnd)

and (@ClinicId=0 or Isnull(ccc.ClinicId,0) = @ClinicId)

group by odt.[ClinicId],

ccc.Name,

FORMAT (cast(odt.[CreateAt] as datetime) , 'MM/yyyy')

SELECT

[ClinicId] = odt.[ClinicId],

[CreateAt] = FORMAT (cast(odt.[CreateAt] as datetime) , 'MM/yyyy'),

[Turnover] = Isnull((odt.Price\*Isnull(SUM(Distinct odt.Quantily),0)) - Isnull(sum(odt.[Discount]),0),0)

FROM [dev\_Rino].[dbo].[cat\_Order\_Detail] odt

left join cat\_Clinic ccc on ccc.ClinicId=odt.ClinicId

left join cat\_Product\_Address cd on cd.ProductId=odt.ProductId

left join cat\_Order co on co.OrderId=odt.OrderId and ISNULL(co.DateUse,0)!=0

where ccc.ParentId=@ClinicParentId

and (FORMAT (cast(odt.[CreateAt] as datetime) , 'MM/yyyy') BETWEEN @TimeStart AND @TimeEnd)

group by odt.[ProductId]

,odt.[Quantily]

,odt.[Price]

,odt.[ClinicId]

,FORMAT (cast(odt.[CreateAt] as datetime) , 'MM/yyyy') ,odt.[Discount],odt.[CreateAt]

order by odt.CreateAt ASC

end

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cms\_cat\_Order\_Detail\_Turnover\_By\_Time: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@UserId=' + convert(nvarchar, @UserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cms\_cat\_Order\_Detail\_Turnover\_By\_Time', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cms\_cat\_Order\_Detail\_Update]

@OrderDetailId bigint ,

@UserId bigint,

@LinkOrClaimId nvarchar(250),

@StatusId int out,

@EstimatedDelivery nvarchar(250)

as

begin try

set @StatusId=2

-- kiểm tra xem nó là sản phẩm hay dịch vụ

-- nếu là sản phẩm nhập link xong đổi trạng thái của sản phẩm là đã gửi hàng

-- nếu là dịch vụ nhập claimid xong là hoàn thành

Declare @ProductId bigint=isnull((select top 1 ProductId from cat\_Order\_Detail where OrderDetailId=@OrderDetailId),0)

Declare @TypeId int=isnull((select typeId from cat\_Product where ProductId=@ProductId),1)

-- type =2 là dịch vụ , 1 là sp

declare @Status int

set @Status=iif(@TypeId=1,4,3)

update cat\_Order\_Detail set LinkOrClaimId=@LinkOrClaimId, StatusId=@Status, EstimatedDelivery=@EstimatedDelivery where OrderDetailId=@OrderDetailId

-- nếu tất cả các sp đã hoàn thành thì chuyển order sang trạng thái mới

declare @OrderId bigint=isnull((select top 1 OrderId from cat\_Order\_Detail where OrderDetailId=@OrderDetailId),0)

declare @Count int=isnull((select count(\*) from cat\_Order\_Detail where OrderId=@OrderId and StatusId!=3),0)

--cập nhật đơn hàng là hoàn thành khi tất cả các sp trong đơn đã hoàn thành

if @Count=0

begin

set @StatusId=1

update cat\_Order set StatusId=4 where OrderId=@OrderId

-- nếu sp vừa cập nhật là haofn thành thì cập nhật ngày hoàn thành của đơn(1 sp trong đơn hoàn thành thì tính all sp trong đơn đó hoàn thành để tính doanh thu)

if(@Status=4)

begin

update cat\_Order set DateUse=GETDATE() where OrderId=@OrderId and ISNULL(DateUse,0)=0

-- cập nhật thông tin xem là đơn ý đã thanh toán hết hết chưa

declare @countPeriod int= (select count(\*) from cat\_Order\_Period where OrderId=@OrderId and isnull( PaymentConfirmation,0)=0)

if @countPeriod=0

update cat\_Order set IsSuccess=1 where OrderId=@OrderId

end

end

end try

begin catch

set @StatusId=-1

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Order\_Receive: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@OrderId=' + convert(nvarchar, @OrderId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Order', 'UPD', @SessionID, @AddlInfo

end catch

ALTER procedure [dbo].[cms\_cat\_Order\_GetDetail\_ByClinicId]

@OrderId bigint,

@UserId bigint--,

--@ClinicId bigint

as

begin try

select

[OrderId] = od.[OrderId]

,[CreateBy] = od.[CreateBy]

,[CreateAt] = od.[CreateAt]

,[FullName] = od.Receiver

,[Address] = od.DeliveryAddress

,[Phone] = od.DeliveryPhone

,IsShowRefuse = iif(DATEDIFF(MINUTE,od.[CreateAt] ,getdate())>1440,0,1)-- hiển thị nút từ chối hay k. Quá 24h không có nút từ chối

,[Total] = od.[Total]

,[StatusId] = isnull(od.StatusId,1),

[StatusName] = iif(isnull(od.StatusId,0)!=0, cs.Name,N'{"vi":"Chưa xác nhận","en":"Unconfimred"}')

from [dbo].[cat\_Order] od (nolock)

left join cat\_Order\_Status cs on cs.OrderStatusId=od.StatusId

left join cat\_Order\_Detail cod on cod.OrderId=od.OrderId

left join cat\_Product cp on cod.ProductId=cp.ProductId

left join cat\_Product\_Address cpa on cpa.ProductId=cp.ProductId and ISNULL(cpa.isdelete,0)=0

where od.OrderId = @OrderId

and Isnull(od.IsDelete,0)=0

and ISNULL(cpa.isdelete,0)=0

-- Thông tin sản phẩm của giỏ hàng

SELECT

[OrderDetailId] =cod.[OrderDetailId]

,[OrderId] =cod.[OrderId]

,[ProductId] =cod.[ProductId]

,[Name] =Isnull(cp.[Name],'{"vi":"","en":""}')

,[Quantily] =cod.[Quantily]

,[Price] =cod.[Price]

,[CreateAt] =cod.[CreateAt]

,[CreateBy] =cod.[CreateBy]

,Discount =iif(isnull(cph.HotDealId,0)=0,0,iif(isnull(cph.typeReduction,0)=1, cph.Discount\*(cod.Price/100),cph.Discount))

-- cod.Discount--typeReduction=true là % giảm, false là số tiền giảm

,[StatusId] =isnull(cod.StatusId,1),

[StatusName] =iif(isnull(cod.StatusId,0)<=1,iif(isnull(cp.TypeId,1)=1,N'{"vi":"Gửi sản phẩm","en":"Send products"}',N'{"vi":"Cung cấp dịch vụ","en":"Provide services"}'),cs.Name)

FROM [cat\_Order\_Detail] cod

left join cat\_Product\_HotDeal cph on cph.ProductId=cod.ProductId

left join cat\_Product\_Status cs on cs.ProductStatusId=cod.StatusId

left join cat\_clinic cc on cc.[ClinicId]=cod.[ClinicId]

left join cat\_Product cp on cod.ProductId=cp.ProductId

left join cat\_Product\_Type cpt on cpt.TypeId=cp.TypeId

left join cat\_Province p on cc.City =p.ProvinceId

left join cat\_District d on cc.District =d.DistrictId

left join cat\_Ward w on cc.Ward =w.WardId

where cod.OrderId = @OrderId

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Order\_Get: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@OrderId=' + convert(nvarchar, @OrderId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Order\_Get', 'GET', null, @AddlInfo

end catch

ALTER procedure [dbo].[cms\_cat\_Order\_GetDetail\_Refusal\_ByClinicId]

@OrderId bigint,

@UserId bigint

as

begin try

select

[OrderId] = od.[OrderId]

,[CreateBy] = od.[CreateBy]

,[CreateAt] = od.[CreateAt]

,[FullName] = od.Receiver

,[Address] = od.DeliveryAddress

,[Phone] = od.DeliveryPhone

--,IsShowRefuse = iif(DATEDIFF(MINUTE,od.[CreateAt] ,getdate())>1440,0,1)-- hiển thị nút từ chối hay k. Quá 24h không có nút từ chối

,[Total] = od.[Total]

,[TotalRefusal] =iif( isnull( od.IsPayFull,0)=0,Sum(cop.AmountPaid),od.Total)

,[StatusId] = isnull(od.StatusId,1),

[StatusName] = isnull(cs.Name,N'{"vi":"Chưa xác nhận","en":"Unconfimred"}'),

[ReasonForRefusal] = od.ReasonForRefusal,

[IsRefusal] = od.IsRefusal,

TotalUseCashBack = od.TotalUseCashBack

from [dbo].[cat\_Order] od (nolock)

left join cat\_Order\_Status cs on cs.OrderStatusId=od.StatusId

left join cat\_Order\_Period cop on cop.OrderId=od.StatusId and ISNULL(cop.StatustId,0)=1

left join cat\_Order\_Detail cod on cod.OrderId=od.OrderId

left join cat\_Product cp on cod.ProductId=cp.ProductId

left join cat\_Product\_Address cpa on cpa.ProductId=cp.ProductId and ISNULL(cpa.isdelete,0)=0

where od.OrderId = @OrderId

and Isnull(od.IsDelete,0)=0

and ISNULL(cpa.isdelete,0)=0

group by

od.[OrderId]

,od.[CreateBy]

,od.[CreateAt]

,od.Receiver

,od.DeliveryAddress

,od.DeliveryPhone

,od.[Total]

,od.IsPayFull

,cop.AmountPaid

,od.StatusId

,od.ReasonForRefusal

,od.IsRefusal

,cs.Name

,od.TotalUseCashBack

-- Thông tin sản phẩm của giỏ hàng

SELECT

[OrderDetailId] =cod.[OrderDetailId]

,[OrderId] =cod.[OrderId]

,[ProductId] =cod.[ProductId]

,[Name] =Isnull(cp.[Name],'{"en":"","vi":""}')

,[Quantily] =cod.[Quantily]

,[Price] =cod.[Price]

,[CreateAt] =cod.[CreateAt]

,[CreateBy] =cod.[CreateBy]

,Discount =cod.Discount

,[StatusId] = isnull(cod.StatusId,1),

[StatusName] = isnull(cs.Name,N'{"vi":"Chưa xác nhận","en":"Unconfimred"}')

FROM [cat\_Order\_Detail] cod

left join cat\_Product\_Status cs on cs.ProductStatusId=cod.StatusId

left join cat\_clinic cc on cc.[ClinicId]=cod.[ClinicId]

left join cat\_Product cp on cod.ProductId=cp.ProductId

left join cat\_Product\_Type cpt on cpt.TypeId=cp.TypeId

left join cat\_Province p on cc.City =p.ProvinceId

left join cat\_District d on cc.District =d.DistrictId

left join cat\_Ward w on cc.Ward =w.WardId

where cod.OrderId = @OrderId

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cat\_Order\_Get: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@OrderId=' + convert(nvarchar, @OrderId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Order\_Get', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cms\_cat\_User\_Device\_By\_List\_String]

@Lang nvarchar(50),

@ListUser nvarchar(max)

as

begin try

begin

SELECT distinct

[UserId] =x.UserID,

[DeviceToken] =x.[DeviceToken],

[DeviceType] =x.[DeviceType],

[Lang] =x.[Lang]

from sec\_User\_Device x

where (isnull(@Lang, '') = '' or x.Lang = @Lang)

and x.UserID IN (SELECT id FROM convert\_string\_to\_table (@ListUser, ',') )

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cms\_cat\_User\_Device\_By\_List\_String: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Lang=' + convert(nvarchar, @Lang)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cms\_cat\_User\_Device\_By\_List\_String', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cms\_cat\_User\_Device\_By\_User]

@CreateBy bigint

as

begin try

begin

SELECT distinct

[UserId] =x.UserID,

[DeviceToken] =x.[DeviceToken],

[DeviceType] =x.[DeviceType],

[Lang] =x.[Lang]

from sec\_User\_Device x

where x.UserID= @CreateBy

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cms\_cat\_User\_Device\_By\_User: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Createby=' + convert(nvarchar, @CreateBy)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cms\_cat\_User\_Device\_By\_User', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cms\_GetListDebt]

@UserId bigint,

@ClinicId bigint,

@DateFrom nvarchar(255),

@DateTo nvarchar(255),

@Keyword nvarchar(255)

as

begin try

begin

select

[DebtId] = d.[DebtId],

[ClinicId] = d.[ClinicId],

[Date] = d.[Date],

[OpeningBalance] = d.[OldDebt],

[Revenue] = d.[Revenue],

[Expense] = d.[Expense],

[RinoPaid] = d.[Paid],

[ClosingBalance] = d.[NewDebt]

from [dbo].[cat\_Debt] d (nolock)

where d.ClinicId = @ClinicId

and ((isnull(@DateFrom,'') = '' or isnull(@DateTo,'') = '')

or convert(datetime, d.[Date], 103) between convert(datetime, @DateFrom, 103) and convert(datetime, @DateTo, 103))

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cms\_GetListDebt: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Keyword=' + convert(nvarchar, @Keyword)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cms\_GetListDebt', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cms\_Rino\_Revenue\_Report]

@Type int,

@Year int

as

begin try

if @Type=1-- nếu là tháng

begin

------lấy ra doanh thu thực tế theo tháng

--select \* from

--(

WITH months(MonthNumber) AS

(

SELECT 0

UNION ALL

SELECT MonthNumber+1

FROM months

WHERE MonthNumber < 12

)

select

[MonthNumber] = isnull(m.MonthNumber,0),

[ActualRevenues] = sum(o.Total) - sum(op.AmountPaid)

from months m

left join cat\_Order o on MONTH(o.CreateAt)=m.MonthNumber

left join cat\_Order\_Period op on month(CONVERT(datetime, PeriodDate, 103))=m.MonthNumber

where Year(o.CreateAt)=@Year

and isnull(o.IsRefusal,0) = 0

and Isnull(op.StatustId,0)=0

group by m.MonthNumber

--) t1

--lấy ra doanh thu chưa nhận theo tháng

;WITH months(MonthNumber) AS

(

SELECT 0

UNION ALL

SELECT MonthNumber+1

FROM months

WHERE MonthNumber < 12

)

select

[MonthNumber] = isnull(m.MonthNumber,0),

[UnearnedRevenue] = sum(op.AmountPaid)

from months m

left join cat\_Order o on MONTH(o.CreateAt)=m.MonthNumber

left join cat\_Order\_Period op on month(CONVERT(datetime, PeriodDate, 103))=m.MonthNumber

where Year(o.CreateAt)=@Year

and isnull(o.IsRefusal,0) = 0

and Isnull(op.StatustId,0)=0

group by m.MonthNumber

end

--theo quý

else

begin

-- doanh thu thực nhận

;WITH months(MonthNumber,id) AS

(

SELECT 1,1

UNION ALL

SELECT MonthNumber+1 , iif(MonthNumber+1<=3,1 , iif(MonthNumber+ 1 <=6,2,iif(MonthNumber+ 1<=9,3,4)))

FROM months

WHERE MonthNumber < 12

)

select

m.id,

[ActualRevenues] = sum(o.Total) - sum(op.AmountPaid)

from months m

left join cat\_Order o on MONTH(o.CreateAt)=m.MonthNumber

left join cat\_Order\_Period op on month(CONVERT(datetime, PeriodDate, 103))=m.MonthNumber

where Year(o.CreateAt)=@Year

and isnull(o.IsRefusal,0) = 0

and Isnull(op.StatustId,0)=0

group by m.id

--doanh thu chưa nhận

;WITH months(MonthNumber,id) AS

(

SELECT 1,1

UNION ALL

SELECT MonthNumber+1 , iif(MonthNumber+1<=3,1 , iif(MonthNumber+ 1 <=6,2,iif(MonthNumber+ 1<=9,3,4)))

FROM months

WHERE MonthNumber < 12

)

select

m.id,

[UnearnedRevenue] = sum(op.AmountPaid)

from months m

left join cat\_Order o on MONTH(o.CreateAt)=m.MonthNumber

left join cat\_Order\_Period op on month(CONVERT(datetime, PeriodDate, 103))=m.MonthNumber

where Year(o.CreateAt)=@Year

and isnull(o.IsRefusal,0) = 0

and Isnull(op.StatustId,0)=0

group by m.id

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cms\_Rino\_Revenue\_Report: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = ''

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cms\_Rino\_Revenue\_Report', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[sec\_User\_GetFireBaseToken\_ByChatGroupId]

@ChatGroupId int

as

begin try

select

DeviceToken = cct.DeviceToken

from [dbo].[sec\_User\_Device] cct (nolock)

where cct.DeviceType = 3

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'sec\_User\_GetFireBaseToken\_ByChatGroupId: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@ChatGroupId=' + convert(nvarchar, @ChatGroupId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'sec\_User\_Device', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[utl\_Insert\_ErrorLog]

@ErrorCd int,

@ErrorMsg varchar(max),

@ProcName varchar(50),

@TableName varchar(50),

@ActionType varchar(3),

@SessionID bigint,

@AddlInfo varchar(max)

as

insert into utl\_Error\_Log

(ErrorNum,

ErrorMsg,

ProcName,

TableName,

ActionType,

SessionID,

AddlInfo,

CreatedDate)

select ErrorNum = @ErrorCd,

ErrorMsg = @ErrorMsg,

ProcName = @ProcName,

TableName = @TableName,

ActionType = @ActionType,

SessionID = @SessionID,

AddlInfo = @AddlInfo,

CreatedDate = getdate()

ALTER PROCEDURE [dbo].[utl\_Insert\_ActionLog]

@tableName nvarchar(255),

@fieldPKName nvarchar(255),

@fieldPKValue nvarchar(50),

@actionName nvarchar(255),

@XML nvarchar(max),

@createBy bigint

as

begin

begin try

declare @valueBefore nvarchar(max)

declare @valueAfter nvarchar(max)

declare @SQL nvarchar(max)

set @SQL = 'SET @Out = (select \* from ' + @tableName + ' where ' + @fieldPKName + '=' + @fieldPKValue + ' for XML AUTO, ELEMENTS)'

declare @xmlBefore XML

EXEC sp\_executesql @SQL, N'@Out XML OUTPUT', @Out = @xmlBefore OUTPUT;

if isnull(@actionName,'update')='update'

begin

select @valueBefore=dbo.ufn\_XmlToJson((@xmlBefore))

end

else

begin

set @valueBefore=''

end

select @valueAfter=dbo.ufn\_XmlToJson((@XML))

insert into utl\_Activity\_Log(

TableName,

FieldPkID,

[Action],

ValueBefore,

ValueAfter,

CreateBy,

CreateTime)

values(

@tableName,

@fieldPKValue,

@actionName,

@valueBefore,

@valueAfter,

@createBy,

getdate()

)

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'utl\_Insert\_ActionLog: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@tableName=' + @tableName

+ ', @fieldPKName=' + @fieldPKName

+ ', @fieldPKValue=' + @fieldPKValue

+ ', @actionName=' + @actionName

+ ', @createBy=' + @createBy

+ ', @XML=' + @XML

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'utl\_Insert\_ActionLog', 'UID', null, @AddlInfo

end catch

end

ALTER PROCEDURE [dbo].[utl\_Insert\_SmsLog]

@Status bit,

@Phone nvarchar(50),

@Content nvarchar(max),

@ResponseMessage nvarchar(max)

as

begin try

insert into dbo.utl\_Sms\_Log

([Status],Phone,Content,ResponseMessage,CreatedDate)

Values

(@Status,@Phone,@Content,@ResponseMessage,GETDATE())

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'utl\_Insert\_SmsLog: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@Status =' + Convert(nvarchar,@Status) + '@Phone =' + Convert(nvarchar,@Phone) + '@Content =' + Convert(nvarchar,@Content) + '@ResponseMessage =' + Convert(nvarchar,@ResponseMessage)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'utl\_Insert\_SmsLog', 'UPD', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[utl\_Notification\_CreateRequestApprove]

@BookId bigint,

@UserId bigint,

@StatusId int,

@RoleIds nvarchar(max)

as

begin try

declare @userName nvarchar(255)

declare @bookTitle nvarchar(255)

set @userName = (select top 1 userName

from sec\_User

where UserId=@UserId)

set @bookTitle = (select top 1 [Name]

from cat\_Book

where BookId=@BookId)

declare @message nvarchar(255)

if (@StatusId = 1)

begin

set @message= N'Nhập liệu viên ' + @userName + N' đã hoàn thành nhập sách ' + @bookTitle

end

if (@StatusId = 2)

begin

set @message= N'Biên tập viên ' + @userName + N' đã duyệt sách ' + @bookTitle

end

declare @Separated char = ','

insert into [utl\_Notification]

(

[UserId],

[Message],

[IsSend],

[IsRead],

[CreateAt])

(select distinct ur.UserId, @message,0,0,getdate()

from sec\_User\_Roles ur

left join sec\_Role r on r.RoleId=ur.RoleId

left join sec\_User u on u.UserId = ur.UserId

where ur.RoleId in (select \* from [dbo].[udf\_Split](@RoleIds,@Separated))

)

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'utl\_Notification\_CreateRequestApprove: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@BookId=' + convert(nvarchar, @BookId) +',@UserId=' + convert(nvarchar, @UserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'utl\_Notification\_CreateRequestApprove', 'UPD', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[utl\_Notification\_CreateRequestApprove\_Question]

@RefId bigint,

@UserId bigint,

@StatusId int,

@RoleIds nvarchar(max)

as

begin try

declare @userName nvarchar(255)

declare @bookTitle nvarchar(255)

declare @QuestionTitle nvarchar(255)

set @userName = (select top 1 userName

from sec\_User

where UserId=@UserId)

set @QuestionTitle = (select top 1 [Title]

from cat\_Question

where QuestionId = @RefId)

set @bookTitle = (select top 1 bbk.[Name]

from cat\_Question qs

left join cat\_Book\_Index bkk

on qs.BookIndexId = bkk.BookIndexId

left join cat\_Book bbk

on bkk.BookId = bbk.BookId

where qs.QuestionId = @RefId)

declare @message nvarchar(255)

if (@StatusId = 1)

begin

set @message= N'Nhập liệu viên ' + @userName + N' hoàn thành câu ' + @QuestionTitle + ' trong sách ' + @bookTitle

end

if (@StatusId = 2)

begin

set @message= N'Biên tập viên ' + @userName + N' đã duyệt câu ' +@QuestionTitle + ' trong sách ' + @bookTitle

end

declare @Separated char = ','

insert into [utl\_Notification]

(

[UserId],

[Message],

[IsSend],

[IsRead],

[CreateAt])

(select distinct ur.UserId, @message,0,0,getdate()

from sec\_User\_Roles ur

left join sec\_Role r on r.RoleId=ur.RoleId

left join sec\_User u on u.UserId = ur.UserId

where (ur.RoleId in (select \* from [dbo].[udf\_Split](@RoleIds,@Separated))

and ur.UserId in (select urp.UserId from sec\_User\_Permission urp where urp.RefId = (select top 1 bkk.[BookId] from cat\_Question qs left join cat\_Book\_Index bkk on qs.BookIndexId = bkk.BookIndexId where qs.QuestionId = @RefId)))

)

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'utl\_Notification\_CreateRequestApprove\_Question: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@QuestionId=' + convert(nvarchar, @RefId) +',@UserId=' + convert(nvarchar, @UserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'utl\_Notification\_CreateRequestApprove\_Question', 'UPD', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[utl\_Notification\_CreateRequestReject]

@BookId bigint,

@UserId bigint

as

begin try

declare @createBy bigint

declare @bookTitle nvarchar(255)

select @bookTitle=[Name],@createBy=[CreateBy]

from cat\_Book

where BookId=@BookId

declare @message nvarchar(255)

set @message= N'Sách ' + convert(nvarchar,@bookTitle) + N' của bạn chưa được phê duyệt'

insert into [utl\_Notification]

(

[UserId],

[Message],

[IsSend],

[IsRead],

[CreateAt])

values (

@createBy,

@message,

0,

0,

getdate()

)

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'utl\_Notification\_CreateRequestApprove: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@BookId=' + convert(nvarchar, @BookId) +',@UserId=' + convert(nvarchar, @UserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'utl\_Notification\_CreateRequestApprove', 'UPD', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[utl\_Notification\_CreateRequestReject\_Question]

@QuestionId bigint,

@UserId bigint

as

begin try

declare @createBy bigint

declare @bookTitle nvarchar(255)

declare @QuestionTitle nvarchar(255)

select @QuestionTitle=[Title] from cat\_Question qs where qs.QuestionId = @QuestionId

select top 1 @bookTitle=bbk.[Name],@createBy=bbk.[CreateBy]

from cat\_Question qs

left join cat\_Book\_Index bkk

on qs.BookIndexId = bkk.BookIndexId

left join cat\_Book bbk

on bkk.BookId = bbk.BookId

where qs.QuestionId = @QuestionId

declare @message nvarchar(255)

set @message= N'Câu hỏi ' + convert(nvarchar,@QuestionTitle) + N' trong sách '+convert(nvarchar,@bookTitle) +N' của bạn chưa được phê duyệt'

insert into [utl\_Notification]

(

[UserId],

[Message],

[IsSend],

[IsRead],

[CreateAt])

values (

@createBy,

@message,

0,

0,

getdate()

)

end try

begin catch

declare @ErrorNum int,

@ErrorMsg nvarchar(200),

@ErrorProc varchar(50),

@SessionID int,

@AddlInfo varchar(max)

set @ErrorNum = error\_number()

set @ErrorMsg = 'utl\_Notification\_CreateRequestReject\_Question: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@QuestionId=' + convert(nvarchar, @QuestionId) +',@UserId=' + convert(nvarchar, @UserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'utl\_Notification\_CreateRequestReject\_Question', 'UPD', @SessionID, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[cms\_Rino\_Revenue\_Report]

@Type int,

@Year int,

@ListYear nvarchar(max)

as

begin try

declare @ListMonth nvarchar(200) = '1,2,3,4,5,6,7,8,9,10,11,12'

declare @ListQuarter nvarchar(200) = '1,2,3,4'

if @Type=1-- nếu là tháng

begin

SELECT

[Month] = th.id

FROM convert\_string\_to\_table (@ListMonth, ',') th

------lấy ra doanh thu thực tế theo tháng

SELECT

[Month] = th.id,

[ActualRevenues] = sum(o.Total) - sum(op.AmountPaid)

FROM convert\_string\_to\_table (@ListMonth, ',') th

left join cat\_Order o on DATEPART(YEAR,o.CreateAt)=@Year and DATEPART(MONTH,o.CreateAt)=th.id

left join cat\_Order\_Period op on DATEPART(YEAR,CONVERT(datetime, op.PeriodDate, 103))=@Year

and DATEPART(MONTH,CONVERT(datetime, op.PeriodDate, 103))=th.id

where Year(o.CreateAt)=@Year

and isnull(o.IsRefusal,0) = 0

and Isnull(op.StatustId,0)=0

group by th.id

--lấy ra doanh thu chưa nhận theo tháng

SELECT

[Month] = th.id,

[UnearnedRevenue] = sum(op.AmountPaid)

FROM convert\_string\_to\_table (@ListMonth, ',') th

left join cat\_Order o on DATEPART(YEAR,o.CreateAt)=@Year and DATEPART(MONTH,o.CreateAt)=th.id

left join cat\_Order\_Period op on DATEPART(YEAR,CONVERT(datetime, op.PeriodDate, 103))=@Year

and DATEPART(MONTH,CONVERT(datetime, op.PeriodDate, 103))=th.id

where Year(o.CreateAt)=@Year

and isnull(o.IsRefusal,0) = 0

and Isnull(op.StatustId,0)=0

group by th.id

end

--theo quý

if(@Type =2) -- tính tổng doanh thu theo quý trong năm nhập vào

begin

SELECT

[Quarter] = th.id

FROM convert\_string\_to\_table (@ListQuarter, ',') th

-- doanh thu thực nhận

SELECT

[Quarter] = th.id,

[ActualRevenues] = sum(o.Total) - sum(op.AmountPaid)

FROM convert\_string\_to\_table (@ListQuarter, ',') th

left join cat\_Order o on DATEPART(YEAR,o.CreateAt)=@Year and DATEPART(QUARTER,o.CreateAt)=th.id

left join cat\_Order\_Period op on DATEPART(YEAR,CONVERT(datetime, op.PeriodDate, 103))=@Year

and DATEPART(QUARTER,CONVERT(datetime, op.PeriodDate, 103))=th.id

where Year(o.CreateAt)=@Year

and isnull(o.IsRefusal,0) = 0

and Isnull(op.StatustId,0)=0

group by th.id

--doanh thu chưa nhận

SELECT

[Quarter] = th.id,

[UnearnedRevenue] = sum(op.AmountPaid)

FROM convert\_string\_to\_table (@ListQuarter, ',') th

left join cat\_Order o on DATEPART(YEAR,o.CreateAt)=@Year and DATEPART(QUARTER,o.CreateAt)=th.id

left join cat\_Order\_Period op on DATEPART(YEAR,CONVERT(datetime, op.PeriodDate, 103))=@Year

and DATEPART(QUARTER,CONVERT(datetime, op.PeriodDate, 103))=th.id

where Year(o.CreateAt)=@Year

and isnull(o.IsRefusal,0) = 0

and Isnull(op.StatustId,0)=0

group by th.id

end

--theo năm

if(@Type =3)

begin

SELECT

[Year] = th.id

FROM convert\_string\_to\_table (@ListYear, ',') th

--Doanh thu thực nhận

SELECT

[Year] = th.id,

[ActualRevenues] = sum(o.Total) - sum(op.AmountPaid)

FROM convert\_string\_to\_table (@ListYear, ',') th

left join cat\_Order o on DATEPART(YEAR,o.CreateAt)=@Year

left join cat\_Order\_Period op on DATEPART(YEAR,CONVERT(datetime, op.PeriodDate, 103))=@Year

where Year(o.CreateAt)=th.id

and isnull(o.IsRefusal,0) = 0

and Isnull(op.StatustId,0)=0

group by th.id

--Doanh thu chưa nhận

SELECT

[Year] = th.id,

[UnearnedRevenue] = sum(op.AmountPaid)

FROM convert\_string\_to\_table (@ListYear, ',') th

left join cat\_Order o on DATEPART(YEAR,o.CreateAt)=@Year

left join cat\_Order\_Period op on DATEPART(YEAR,CONVERT(datetime, op.PeriodDate, 103))=@Year

where Year(o.CreateAt)=th.id

and isnull(o.IsRefusal,0) = 0

and Isnull(op.StatustId,0)=0

and Year(CONVERT(datetime, PeriodDate, 103))=th.id

group by th.id

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cms\_Rino\_Revenue\_Report: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = ''

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cms\_Rino\_Revenue\_Report', 'GET', null, @AddlInfo

end catch

USE [dev\_Rino]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[cms\_Clinic\_Information\_Update] Script Date: 27/5/2022 8:15:18 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

/\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- Author: Phucvv

-- Description: insert, update branch of enterprise

-- Parent: Internal

-- Date PIC Update record

-- 2022/04/06 Phucvv InsertUpdate

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*/

ALTER PROCEDURE [dbo].[cms\_Clinic\_Information\_Update]

@ClinicId bigint out,

@UserId bigint,

@XMLID int,

@XML nvarchar(max)

as

BEGIN

BEGIN TRY

--- clinic khi update vào các bảng tạm là clinic cũ , không phải clinic mới sinh ra từ bảng tạm

update cat\_Clinic set UpdateApproval =1 where ClinicId =@ClinicId

declare @ClinicidNew bigint=isnull((select top 1 ClinicId from cat\_Clinic\_User where OldId=@Clinicid) ,0)

if @ClinicidNew>0

begin

delete from cat\_Clinic\_User where OldId=@ClinicId

delete from cat\_Clinic\_Image\_User where ClinicId=@ClinicidNew

delete from cat\_Clinic\_Category\_Xref\_User where ClinicId=@ClinicidNew

delete from cat\_Clinic\_Representative\_User where ClinicId=@ClinicidNew

end

BEGIN

-- Update mode

-- Kiem tra bang Clincic User có Clinic hau chưa

declare @CheckClincId bigint = Isnull((select Count([OldId]) from [dbo].[cat\_Clinic\_User] cu left join cat\_Clinic c on c.ClinicId = cu.OldId where cu.OldId = @ClinicId),0)

--- kiem tra bang nguoi dai dien

declare @checkRepresentative int = Isnull((select Count(cru.OldId) from [dbo].[cat\_Clinic\_Representative\_User] cru left join cat\_Clinic\_Representative cr on cr.ClinicId = cru.OldId where cru.OldId = @ClinicId ),0)

----Kiem tra chinh sach

declare @checkPolicy int = Isnull((select Count(cru.OldId) from [dbo].[cat\_Clinic\_Policy\_User] cru left join cat\_Clinic\_Policy cr on cr.ClinicId = cru.OldId where cru.OldId = @ClinicId ),0)

----Kiem tra anh

declare @checkImage int = Isnull((select Count(cru.OldId) from [dbo].[cat\_Clinic\_Image\_User] cru left join cat\_Clinic\_Image cr on cr.ClinicId = cru.OldId where cru.OldId = @ClinicId ),0)

----Kiem tra linh vuc

declare @checkCategory int = Isnull((select Count(cru.OldId) from [dbo].[cat\_Clinic\_Category\_Xref\_User] cru left join cat\_Clinic\_Category\_Xref cr on cr.ClinicId = cru.OldId where cru.OldId = @ClinicId ),0)

--declare @OldIdRepresentative int = (select cru.OldId from [dbo].[cat\_Clinic\_Representative\_User] cru left join cat\_Clinic\_Representative cr on cr.ClinicId = cru.OldId where cru.OldId = @ClinicId)

--declare @OldCategory int = (select cru.OldId from [dbo].[cat\_Clinic\_Category\_Xref\_User] cru left join cat\_Clinic\_Category\_Xref cr on cr.ClinicId = cru.OldId where cru.OldId = @ClinicId)

--declare @OldRepresentative int = (select cru.OldId from [dbo].[cat\_Clinic\_Representative\_User] cru left join cat\_Clinic\_Representative cr on cr.ClinicId = cru.OldId where cru.OldId = @ClinicId)

--declare @OldPolyci int = (select cru.OldId from [dbo].[cat\_Clinic\_Policy\_User] cru left join cat\_Clinic\_Policy cr on cr.ClinicId = cru.OldId where cru.OldId = @ClinicId)

--declare @OldImage int = (select cru.OldId from [dbo].[cat\_Clinic\_Image\_User] cru left join cat\_Clinic\_Image cr on cr.ClinicId = cru.OldId where cru.OldId = @ClinicId)

--declare @UserClincId bigint = (select c.UserId from [dbo].[cat\_Clinic\_User] cu left join cat\_Clinic c on c.ClinicId = cu.OldId where cu.OldId = @ClinicId)

if @CheckClincId =0

begin

insert into cat\_Clinic\_User

(

[Name] ,

[Email] ,

[Phone] ,

[OpenTime],

[CloseTime],

[Ward] ,

[District] ,

[City] ,

[Address] ,

[CreateBy] ,

[CreateAt] ,

[IsDelete] ,

[Status] ,

[Description],

[ParentId] ,

[RegisteredBusinessNumber],

[BusinessLicense],

[TaxCode],

[BankInformation],

[ExpenseRino],

[FileUrl],

[Latitude],

[Longitude],

[StatustId],

[LinkUrl],

[RinoApproval],

[OldId],

[UserId],

Contract

)

select

[Name] = x.[Name] ,

[Email] = x.[Email] ,

[Phone] = x.[Phone] ,

[OpenTime] = x.[OpenTime] ,

[CloseTime] = x.[CloseTime] ,

[Ward] = x.[Ward] ,

[District] = x.[District] ,

[City] = x.[City] ,

[Address] = x.[Address] ,

[CreateBy] = @UserId ,

[CreateAt] = getdate() ,

[IsDelete] = 0 ,

[Status] = 0 ,

[Description] = x.[Description] ,

[ParentId] = null,

[RegisteredBusinessNumber] = x.[RegisteredBusinessNumber],

[BusinessLicense] = x.[BusinessLicense],

[TaxCode] = x.[TaxCode],

[BankInformation] = x.[BankInformation],

[ExpenseRino] = x.[ExpenseRino],

[FileUrl] = x.[FileUrl],

[Latitude] = x.[Latitude],

[Longitude] = x.[Longitude],

[StatustId] = x.[StatustId],

[LinkUrl] = x.[LinkUrl],

[RinoApproval] = 1,

[OldId] = @ClinicId,

[UserId] = @UserId,

Contract =x.Contract

from openxml(@XMLID, '/ClinicInformation', 2)

with (

[Name] nvarchar(255),

[Email] nvarchar(255),

[Phone] nvarchar(255),

[OpenTime] nvarchar(255),

[CloseTime] nvarchar(255),

[Ward] int,

[District] int,

[City] int,

[Address] nvarchar(255),

[RegisteredBusinessNumber] nvarchar(50),

[BusinessLicense] nvarchar(50),

[TaxCode] nvarchar(50),

[BankInformation] nvarchar(250),

[ExpenseRino] float,

[FileUrl] nvarchar(250),

[LinkUrl] nvarchar(250),

[CreateBy] int,

[CreateAt] datetime,

[IsDelete] bit,

[Status] int,

[Description] nvarchar(max),

[ParentId] bigint,

[Latitude] decimal(18,15),

[Longitude] decimal(18,15),

[RinoApproval] bit,

[StatustId] int,

[OldId] bigint,

[UserId] bigint ,

Contract nvarchar(Max)

) x

set @ClinicId = scope\_identity()

----------------------them nguoi dai dien--------------------------------------

insert into cat\_Clinic\_Representative\_User

(

[Representative],

[Phone],

[Representative2],

[Phone2],

[ClinicId]

)

select

[Representative] = z.[Representative],

[Phone] = z.[Phone],

[Representative2] = z.[Representative2],

[Phone2] = z.[Phone2],

[ClinicId] = @ClinicId

from openxml(@XMLID, '/ClinicInformation/ListRepresentative', 2)

with (

[RepresentativeId] bigint,

[Representative] nvarchar(255),

[Phone] nvarchar(15),

[Representative2] nvarchar(255),

[Phone2] nvarchar(15),

[OldId] bigint,

[ClinicId] bigint

) z

--WHERE

-- ISNULL( z.[RepresentativeId],0)=0

-------------------them anh -----------------------

insert into [dbo].[cat\_Clinic\_Image\_User] ---them anh trong image----

(

[ClinicId],

[IsActive],

[IsDelete],

[Path]

)

select

[ClinicId] = @ClinicId,

[IsActive] = 1,

[IsDelete] = 0,

[Path] = x.[Path]

from openxml(@XMLID, '/ClinicInformation/ListImages/ClinicFileModel', 2)

with (

[ClinicFileId] bigint,

[ClinicId] bigint,

[OldId] bigint,

[IsActive] bit,

[IsDelete] bit,

[Path] nvarchar(255)

) x

--where ISNULL( x.[ClinicFileId],0)=0

--------------------------Them linh vuc---------------------------------------

---Insert cac linh vuc moi---

insert into [dbo].[cat\_Clinic\_Category\_Xref\_User]

(

[ClinicId],

[CategoryId],

[CreateBy],

[CreateAt]

)

select

[ClinicId] = @ClinicId ,

[CategoryId] = x.[CategoryId],

[CreateBy] = @UserId,

[CreateAt] = getdate()

from openxml(@XMLID, '/ClinicInformation/ListCategory/ClinicCategory', 2)

with(

[ClinicCategoryId] bigint,

[ClinicId] bigint,

[CategoryId] bigint

) x

end

end

END TRY

BEGIN CATCH

DECLARE @ErrorNum INT,

@ErrorMsg NVARCHAR(200),

@ErrorProc VARCHAR(50),

@SessionID INT,

@AddlInfo VARCHAR(max)

SET @ErrorNum = ERROR\_NUMBER()

SET @ErrorMsg = 'cms\_Clinic\_Information\_InsertUpdate: ' + ERROR\_MESSAGE()

SET @ErrorProc = ERROR\_PROCEDURE()

SET @AddlInfo = '@ClinicId=' + convert(NVARCHAR, @ClinicId)

EXEC utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cat\_Clinic', 'UPD', @SessionID, @AddlInfo

END CATCH

END

USE [dev\_Rino]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[cms\_Clinic\_Information\_Update\_Approval] Script Date: 27/5/2022 8:17:04 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- Created By: Phucvv

-- Created Time: 2022/02/09

-- methor : Chap nhan thay doi thong tin phong kham

ALTER PROCEDURE [dbo].[cms\_Clinic\_Information\_Update\_Approval]

@ClinicId bigint,

@Approval bit,

@UserId int,

@StatusID int out

as

begin try

set @StatusID = 0

declare @clinicIdNew bigint=isnull((select ClinicId from cat\_Clinic\_User where OldId = @ClinicId ),0)

--Kiem tra doanh nghiep co muon thay doi thong tin phong kham

declare @CheckApproval int

set @CheckApproval = isnull((select UpdateApproval from cat\_Clinic where ClinicId = @ClinicId ),0)

-- select @CheckApproval

if(@CheckApproval =1 and @Approval = 1 ) -- co yeu cau thay doi thong tin phong kham va rino chap thuan thay doi thong tin(@Approval = 1)

begin

update c

set

[UpdateApproval] = 0,-- UpdateApproval = 0 la trang thai chua updpate , = 1 la trang thai yeu cua update

[Name] = cu.[Name] ,

[Description] = cu.[Description],

[Email] = cu.[Email] ,

[Phone] = cu.[Phone] ,

[Ward] = cu.[Ward] ,

[District] = cu.[District] ,

[City] = cu.[City] ,

[Address] = cu.[Address] ,

[RegisteredBusinessNumber] = cu.[RegisteredBusinessNumber],--[Số đăng ký]

[BusinessLicense] = cu.[BusinessLicense],--[Giấy phép kinh doanh]

[TaxCode] = cu.[TaxCode],--,[Mã số thuế]

[BankInformation] = cu.[BankInformation],--[Thông tin ngân hàng]

[ExpenseRino] = cu.[ExpenseRino],--[chi phi Rino]

[FileUrl] = cu.[FileUrl],--file hop dong

[LinkUrl] = cu.[LinkUrl],--link lien ket

[Contract] = cu.[Contract],

[Latitude] = cu.[Latitude],-- toa do

[Longitude] = cu.[Longitude],-- toa do

[UpdateBy] = @UserId ,

[UpdateAt] = getdate()

FROM cat\_Clinic c (nolock)

INNER JOIN cat\_Clinic\_User cu ON c.ClinicId = cu.OldId

where c.ClinicId =@ClinicId and isnull(c.IsDelete, 0) = 0

-------------Update nguoi dai dien () --------------------

delete cat\_Clinic\_Representative where ClinicId = @ClinicId

insert into cat\_Clinic\_Representative(

[Representative]

,[Phone]

,[Representative2]

,[Phone2]

,[ClinicId]

,[CreateAt]

,[CreateBy]

,[UpdateAt]

,[UpdateBy]

,[IsDelete] ) select

cru.[Representative],

cru.[Phone],

cru.[Representative2],

cru.[Phone2],

@ClinicId,

cru.[CreateAt],

cru.[CreateBy],

cru.[UpdateAt],

cru.[UpdateBy],

cru.[IsDelete]

from cat\_Clinic\_Representative\_User cru

where cru.ClinicId =@clinicIdNew

-------------------Update Policy (Chinh Sach) MÀN HÌNH NAY K CẦN CẬP NHẬT CHÍNH SÁCH------------------------------------------

-------------Update image thay doi-------

delete cat\_Clinic\_Image where ClinicId = @ClinicId -- xoa anh trong clinic image

insert into [dbo].[cat\_Clinic\_Image] ---them anh trong image----

(

[ClinicId],

[IsActive],

[IsDelete],

[Path]

)

select

[ClinicId] =@ClinicId,

[IsActive] = ciu.[IsActive],

[IsDelete] = ciu.[IsDelete],

[Path] = ciu.[Path]

from cat\_Clinic\_Image\_User ciu

where ClinicId =@clinicIdNew

---------------------------update category = thay doi linh vuc--------------------------------------------------

delete cat\_Clinic\_Category\_Xref where ClinicId = @ClinicId -- xoa anh trong clinic image

---Insert cac linh vuc moi---

insert into [dbo].[cat\_Clinic\_Category\_Xref]

(

[ClinicId],

[CategoryId],

[Isdelete],

[CreateBy],

[CreateAt]

)

select

[ClinicId] = @ClinicId,

[CategoryId] = x.[CategoryId],

[IsDelete] = x.[IsDelete],

[CreateBy] = x.[CreateBy],

[CreateAt] = x.[CreateAt]

from cat\_Clinic\_Category\_Xref\_User x

where ClinicId =@clinicIdNew

set @StatusID = 1

delete from cat\_Clinic\_User where OldId=@ClinicId

delete from cat\_Clinic\_Image\_User where ClinicId=@clinicIdNew

delete from cat\_Clinic\_Category\_Xref\_User where ClinicId=@clinicIdNew

delete from cat\_Clinic\_Representative\_User where ClinicId=@clinicIdNew

update cat\_Clinic set UpdateApproval = 0

end

if(@CheckApproval =1 and @Approval = 0 )--co yeu cau thay doi thong tin phong kham nhung rino tu choi thay doi thong tin(@Approval = 1)

begin

update cat\_Clinic set UpdateApproval = 0 where ClinicId =@ClinicId and ISNULL(IsDelete,0)=0-- tro ve trang thai chua yeu cau update

delete from cat\_Clinic\_User where OldId=@ClinicId

delete from cat\_Clinic\_Image\_User where ClinicId=@clinicIdNew

delete from cat\_Clinic\_Category\_Xref\_User where ClinicId=@clinicIdNew

delete from cat\_Clinic\_Representative\_User where ClinicId=@clinicIdNew

set @StatusID =2

end

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cms\_Clinic\_Information\_Update\_Approval: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@ClinicId=' + convert(nvarchar, @ClinicId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cms\_Clinic\_Information\_Update\_Approval', 'UPP', null, @AddlInfo

end catch

USE [dev\_Rino]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[cms\_cat\_Order\_Period\_List\_User] Script Date: 30/5/2022 8:59:40 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

/\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- Author: moipq

-- Description: danh sách user sắp đến kỳ thanh toán, đến ngày thanh toán, chậm thanh toán

-- Parent: Internal

-- Date PIC Update record

-- 2022/04/12 Vuognnx get detail product

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*/

ALTER PROCEDURE [dbo].[cms\_cat\_Order\_Period\_List\_User]

as

begin try

-- danh sách nhưng user có thanh toán định kì trong 3 ngày tới , update những order đó (typeNoti =1) để lần sau k gửi nữa

SELECT

[OrderId] =OrderPeriodId,

UserId = [CreateBy]

FROM [dev\_Rino].[dbo].[cat\_Order\_Period]

where DATEDIFF(DAY,GETDATE(),TRY\_PARSE(PeriodDate AS DATETIME USING 'en-gb')) =3

and Isnull([TypeNoti],0)=0 and ISNULL(StatustId,0)=0

update [dbo].[cat\_Order\_Period] set [TypeNoti] =1 where Isnull([TypeNoti],0)=0 and DATEDIFF(DAY,GETDATE(),TRY\_PARSE(PeriodDate AS DATETIME USING 'en-gb')) =3

-- danh sách nhưng user có thanh toán định kì hôm nay

SELECT

[OrderId] =OrderPeriodId,

UserId = [CreateBy]

FROM [dev\_Rino].[dbo].[cat\_Order\_Period]

where DATEDIFF(DAY,GETDATE(),TRY\_PARSE(PeriodDate AS DATETIME USING 'en-gb')) =0 and [TypeNoti] =1 and Isnull(StatustId,0)=0

update [dbo].[cat\_Order\_Period] set [TypeNoti] =2 where [TypeNoti]=1 and DATEDIFF(DAY,GETDATE(),TRY\_PARSE(PeriodDate AS DATETIME USING 'en-gb')) =0

-- danh sách nhưng user có thanh toán bị muộn

SELECT

[OrderId] =OrderPeriodId,

UserId = [CreateBy]

FROM [dev\_Rino].[dbo].[cat\_Order\_Period]

where DATEDIFF(DAY,GETDATE(),TRY\_PARSE(PeriodDate AS DATETIME USING 'en-gb')) <=-1

and [TypeNoti]=2 and ISNULL(StatustId,0)=0

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cms\_cat\_Order\_Period\_List\_User: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = ''

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cms\_cat\_Order\_Period\_List\_User', 'GET', null, @AddlInfo

end catch

USE [dev\_Rino]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[cms\_Support\_GetList] Script Date: 3/6/2022 8:54:25 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- Created By: halt

ALTER PROCEDURE [dbo].[cms\_Support\_GetList]

@IsClinic bit,

@ListCity nvarchar(max),

@UserId bigint,

@Keyword nvarchar(255),

@PageIndex int,

@PageSize int,

@TotalRecord bigint out,

@Sort nvarchar(100),

@Direction nvarchar(100)

as

begin try

select

[SupportId] = s.[SupportId],

[SupportIssue] = s.[SupportIssue],

[Solution] = s.[Solution],

[CreateBy] = s.[CreateBy],

[CreateAt] = s.[CreateAt],

[Supporter] = s.[Supporter],

[StatusSupport] = s.[StatusSupport],

[Name] = iif(@IsClinic=1,cl.Name,u.FullName),

[DateSupport] = s.[DateSupport],

[IsClinic] =s.[IsClinic],

Email =iif(@IsClinic=1,cl.Email,u.Email),

Phone =iif(@IsClinic=1,cl.Phone,u.PhoneNumber)

from [dbo].[sec\_Support] s (nolock)

left join cat\_Clinic cl on cl.UserId = s.CreateBy and cl.City in (SELECT id FROM convert\_string\_to\_table (@ListCity, ',') )

left join sec\_User u on u.UserId= s.CreateBy and u.ProvinceId in (SELECT id FROM convert\_string\_to\_table (@ListCity, ','))

where

(@IsClinic =2 or Isnull(s.IsClinic,0)=@IsClinic)

and

([SupportIssue] like Concat('%',@Keyword,'%'))

order by

CASE WHEN @Sort = '' AND @Direction = 'asc' THEN [SupportId] END DESC,

CASE WHEN @Sort = '' AND @Direction = 'desc' THEN [SupportId] END DESC,

CASE WHEN @Sort = '' AND @Direction = '' THEN [SupportId] END DESC

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

select

@TotalRecord = count([SupportId])

from [dbo].[sec\_Support] s (nolock)

left join cat\_Clinic cl on cl.UserId = s.CreateBy and cl.City in (SELECT id FROM convert\_string\_to\_table (@ListCity, ',') )

left join sec\_User u on u.UserId= s.CreateBy and u.ProvinceId in (SELECT id FROM convert\_string\_to\_table (@ListCity, ','))

where

(@IsClinic =2 or Isnull(s.IsClinic,0)=@IsClinic)

and

([SupportIssue] like Concat('%',@Keyword,'%'))

select

[SupportId] = s.[SupportId],

[SupportImageId] = spi.[SupportImageId],

[ImageUrl] = spi.[ImageUrl]

from [dbo].[sec\_Support] s (nolock)

left join cat\_Clinic cl on cl.UserId = s.CreateBy and cl.City in (SELECT id FROM convert\_string\_to\_table (@ListCity, ',') )

left join sec\_User u on u.UserId= s.CreateBy and u.ProvinceId in (SELECT id FROM convert\_string\_to\_table (@ListCity, ','))

join cat\_Support\_Image spi on spi.SupportId = s.SupportId and isnull(spi.IsDelete,0)=0

where

(@IsClinic =2 or Isnull(s.IsClinic,0)=@IsClinic)

and

([SupportIssue] like Concat('%',@Keyword,'%'))

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cms\_Support\_GetList: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = ''

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cms\_Support\_GetList', 'GET', null, @AddlInfo

end catch

USE [dev\_Rino]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[cms\_Clinic\_Detail] Script Date: 6/6/2022 4:03:41 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

/\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

-- Author: Vuongnx

-- Description: Clinic Information

-- Parent: Internal

-- Date PIC

-- 2022/04/08 vươnnx

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*/

ALTER PROCEDURE [dbo].[cms\_Clinic\_Detail]

@ClinicId bigint,

@UserId bigint

as

begin try

declare @check int

set @check = isnull((select UpdateApproval from cat\_Clinic where UserId =@UserId),0)

if(@check = 0)-- khi da chap nhan update lay thong tin tu bang cat\_clinic

begin

--Clinic information

select distinct

[ClinicId] = r.[ClinicId],

[Name] = isnull(r.[Name],'{"en":"","vi":""}'),

[Description] = isnull(r.[Description],'{"en":"","vi":""}'),

[Email] = r.[Email],

[Phone] = r.[Phone],

[Ward] = r.[Ward],

[District] = r.[District],

[City] = r.[City],

[Address] = r.[Address],

[CreateBy] = r.[CreateBy],

[CreateAt] = r.[CreateAt],

[UpdateAt] = r.[UpdateAt],

[UpdateBy] = r.[UpdateBy],

[ParentId] = r.[ParentId],

[UserId] = r.[UserId],

[RegisteredBusinessNumber] = r.[RegisteredBusinessNumber],

[BusinessLicense] = r.[BusinessLicense],

[TaxCode] = r.[TaxCode],

[BankInformation] = r.[BankInformation],

[ExpenseRino] = r.[ExpenseRino],

[FileUrl] = r.[FileUrl],

[Latitude] = r.[Latitude],

[Longitude] = r.[Longitude],

[StatustId] = r.[StatustId],

[LinkUrl] = r.[LinkUrl],

[Contract] = r.[Contract],

[CityName] = p.[Name],

[DistrictName] = d.[Name],

[WardName] = w.[Name],

TypeId =iif(ISNULL(r.parentId,0)!=0, iif(count(r1.[ClinicId])>0,2,3) ,3),

Contract =r.Contract,

[PackageManagementName] = cpms.[Name] -- Gói quản lý đã đăng ký

from [dbo].[cat\_Clinic] r (nolock)

left join cat\_Clinic r1 on r.ParentId=r1.ClinicId

left join cat\_Clinic\_Image cm on cm.ClinicId = r.ClinicId

left join cat\_Province p on r.City =p.ProvinceId

left join cat\_District d on r.District =d.DistrictId

left join cat\_Ward w on w.WardId = r.Ward

left join cat\_Clinic\_Package\_Management\_Services cpms on cpms.PackageManagementId = r.PackageManagementServicesId

where

isnull(r.IsDelete,0) = 0 and isnull(cm.IsDelete,0)= 0 and r.ClinicId = @ClinicId

GROUP BY

r.[ClinicId],

isnull(r.[Name],'{"en":"","vi":""}'),

isnull(r.[Description],'{"en":"","vi":""}'),

r.[Email],

r.[Phone],

r.[Ward],

r.[District],

r.[City],

r.[Address],

r.[CreateBy],

r.[CreateAt],

r.[UpdateAt],

r.[UpdateBy],

r.[ParentId],

r.[UserId],

r.[RegisteredBusinessNumber],

r.[BusinessLicense],

r.[TaxCode],

r.[BankInformation],

r.[ExpenseRino],

r.[FileUrl],

r.[Latitude],

r.[Longitude],

r.[StatustId],

r.[LinkUrl],

r.[Contract],

cm.[Path],

p.[Name],

d.[Name],

w.[Name],

r.Contract,

cpms.[Name]

--Category information = lĩnh vực

select

[CategoryId] = r.[CategoryId],

[Name] = isnull(r.[Name],'{"en":"","vi":""}'),

[ClinicCategoryId] = c.[ClinicCategoryId],

[ClinicId] = c.[ClinicId]

from [dbo].[cat\_Category] r (nolock)

left join cat\_Clinic\_Category\_Xref c on c.CategoryId= r.CategoryId

left join cat\_Clinic cc on cc.ClinicId=c.ClinicId and cc.ClinicId =@ClinicId

where

isnull(r.IsDelete,0) = 0 and cc.UserId=@UserId

--Nguoi dai dien

select

[RepresentativeId] = r.[RepresentativeId],

[ClinicId] = r.[ClinicId],

[Representative] = r.[Representative],

[Phone] = r.[Phone],

[Representative2] = r.[Representative2],

[Phone2] = r.[Phone2]

from [dbo].[cat\_Clinic\_Representative] r (nolock)

left join cat\_Clinic cc on cc.ClinicId=r.ClinicId

where

isnull(r.IsDelete,0) = 0 and cc.ClinicId =@ClinicId

end

else

begin

select distinct

[ClinicId] = r.[ClinicId],

[Name] = isnull(r.[Name],'{"en":"","vi":""}'),

[Description] = isnull(r.[Description],'{"en":"","vi":""}'),

[Email] = r.[Email],

[Phone] = r.[Phone],

[Ward] = r.[Ward],

[District] = r.[District],

[City] = r.[City],

[Address] = r.[Address],

[CreateBy] = r.[CreateBy],

[CreateAt] = r.[CreateAt],

[UpdateAt] = r.[UpdateAt],

[UpdateBy] = r.[UpdateBy],

[ParentId] = r.[ParentId],

[UserId] = r.[UserId],

[RegisteredBusinessNumber] = r.[RegisteredBusinessNumber],

[BusinessLicense] = r.[BusinessLicense],

[TaxCode] = r.[TaxCode],

[BankInformation] = r.[BankInformation],

[ExpenseRino] = r.[ExpenseRino],

[FileUrl] = r.[FileUrl],

[Latitude] = r.[Latitude],

[Longitude] = r.[Longitude],

[StatustId] = r.[StatustId],

[LinkUrl] = r.[LinkUrl],

[Contract] = r.[Contract],

[CityName] = p.[Name],

[DistrictName] = d.[Name],

[WardName] = w.[Name],

TypeId =iif(ISNULL(r.parentId,0)!=0, iif(count(r1.[ClinicId])>0,2,3) ,3),

Contract =r.Contract,

[PackageManagementName] = cpms.[Name] -- Gói quản lý đã đăng ký

from [dbo].[cat\_Clinic\_User] r (nolock)

left join cat\_Clinic r1 on r.ParentId=r1.ClinicId

left join cat\_Clinic\_Image cm on cm.ClinicId = r.ClinicId

left join cat\_Province p on r.City =p.ProvinceId

left join cat\_District d on r.District =d.DistrictId

left join cat\_Ward w on w.WardId = r.Ward

left join cat\_Clinic\_Package\_Management\_Services cpms on cpms.PackageManagementId = r.PackageManagementServicesId

where

isnull(r.IsDelete,0) = 0 and isnull(cm.IsDelete,0)= 0 and r.OldId =@ClinicId

GROUP BY

r.[ClinicId],

isnull(r.[Name],'{"en":"","vi":""}'),

isnull(r.[Description],'{"en":"","vi":""}'),

r.[Email],

r.[Phone],

r.[Ward],

r.[District],

r.[City],

r.[Address],

r.[CreateBy],

r.[CreateAt],

r.[UpdateAt],

r.[UpdateBy],

r.[ParentId],

r.[UserId],

r.[RegisteredBusinessNumber],

r.[BusinessLicense],

r.[TaxCode],

r.[BankInformation],

r.[ExpenseRino],

r.[FileUrl],

r.[Latitude],

r.[Longitude],

r.[StatustId],

r.[LinkUrl],

r.[Contract],

cm.[Path],

p.[Name],

d.[Name],

w.[Name],

r.Contract,

cpms.[Name]

--Category information = lĩnh vực

select

[CategoryId] = r.[CategoryId],

[Name] = isnull(r.[Name],'{"en":"","vi":""}'),

[ClinicCategoryId] = c.[ClinicCategoryId],

[ClinicId] = c.[ClinicId]

from [dbo].[cat\_Category] r (nolock)

left join cat\_Clinic\_Category\_Xref\_User c on c.CategoryId= r.CategoryId

left join cat\_Clinic\_User cc on cc.ClinicId=c.ClinicId

where

isnull(r.IsDelete,0) = 0 and cc.ClinicId =@ClinicId

--Nguoi dai dien

select

[RepresentativeId] = r.[RepresentativeId],

[ClinicId] = r.[ClinicId],

[Representative] = r.[Representative],

[Phone] = r.[Phone],

[Representative2] = r.[Representative2],

[Phone2] = r.[Phone2]

from [dbo].[cat\_Clinic\_Representative\_User] r (nolock)

left join cat\_Clinic cc on cc.ClinicId=r.ClinicId

where

isnull(r.IsDelete,0) = 0 and cc.ClinicId =@ClinicId

end

-- công nợ của clinic

declare @OldDebt bigint -- số dư cuối kỳ lần trước

set @OldDebt = Isnull((select top 1 [NewDebt] from cat\_Debt where ClinicId = @ClinicId order by [DebtId] desc),0)

declare @Revenue bigint -- doanh thu gần đây nhất của doanh nghiệp =tổng doanh thu của các chi nhánh hoặc của ncc đó

set @Revenue = Isnull((

select sum(Total)

from cat\_Order od

left join cat\_Clinic cl on cl.ClinicId= od.ClinicId

where (cl.ClinicId = @Clinicid or cl.ParentId = @Clinicid)

and IsSuccess=1 and od.ParentId >=1 and Isnull(od.[IsDebtCalculation],0)=0

),0)

select

[OldDebt] = @OldDebt,

[Revenue] = @Revenue

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cms\_Clinic\_Detail: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = '@UserId=' + convert(nvarchar, @UserId)

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cms\_Clinic\_Detail', 'GET', null, @AddlInfo

end catch

USE [dev\_Rino]

GO

/\*\*\*\*\*\* Object: StoredProcedure [dbo].[cms\_Support\_GetList] Script Date: 6/6/2022 4:37:04 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

-- Created By: halt

ALTER PROCEDURE [dbo].[cms\_Support\_GetList]

@IsClinic bit,

@ListCity nvarchar(max),

@UserId bigint,

@Keyword nvarchar(255),

@PageIndex int,

@PageSize int,

@TotalRecord bigint out,

@Sort nvarchar(100),

@Direction nvarchar(100)

as

begin try

select

[SupportId] = s.[SupportId],

[SupportIssue] = s.[SupportIssue],

[Solution] = s.[Solution],

[CreateBy] = s.[CreateBy],

[CreateAt] = s.[CreateAt],

[Supporter] = s.[Supporter],

[StatusSupport] = s.[StatusSupport],

[Name] = iif(@IsClinic=1,cl.Name,u.FullName),

[DateSupport] = s.[DateSupport],

[IsClinic] =s.[IsClinic],

Email =iif(@IsClinic=1,cl.Email,u.Email),

Phone =iif(@IsClinic=1,cl.Phone,u.PhoneNumber)

from [dbo].[sec\_Support] s (nolock)

left join cat\_Clinic cl on cl.UserId = s.CreateBy and cl.City in (SELECT id FROM convert\_string\_to\_table (@ListCity, ',') )

left join sec\_User u on u.UserId= s.CreateBy and u.ProvinceId in (SELECT id FROM convert\_string\_to\_table (@ListCity, ','))

where

(@IsClinic =2 or Isnull(s.IsClinic,0)=@IsClinic)

and

([SupportIssue] like Concat('%',@Keyword,'%'))

order by

CASE WHEN @Sort = '' AND @Direction = 'asc' THEN [SupportId] END DESC,

CASE WHEN @Sort = '' AND @Direction = 'desc' THEN [SupportId] END DESC,

CASE WHEN @Sort = '' AND @Direction = '' THEN [SupportId] END DESC

offset @PageSize \* (@PageIndex - 1) ROWS

fetch next @PageSize ROWS ONLY;

select

@TotalRecord = count([SupportId])

from [dbo].[sec\_Support] s (nolock)

left join cat\_Clinic cl on cl.UserId = s.CreateBy and cl.City in (SELECT id FROM convert\_string\_to\_table (@ListCity, ',') )

left join sec\_User u on u.UserId= s.CreateBy and u.ProvinceId in (SELECT id FROM convert\_string\_to\_table (@ListCity, ','))

where

(@IsClinic =2 or Isnull(s.IsClinic,0)=@IsClinic)

and

([SupportIssue] like Concat('%',@Keyword,'%'))

select

[SupportId] = s.[SupportId],

[SupportImageId] = spi.[SupportImageId],

[ImageUrl] = spi.[ImageUrl]

from [dbo].[sec\_Support] s (nolock)

left join cat\_Clinic cl on cl.UserId = s.CreateBy and cl.City in (SELECT id FROM convert\_string\_to\_table (@ListCity, ',') )

left join sec\_User u on u.UserId= s.CreateBy and u.ProvinceId in (SELECT id FROM convert\_string\_to\_table (@ListCity, ','))

join cat\_Support\_Image spi on spi.SupportId = s.SupportId and isnull(spi.IsDelete,0)=0

where

(@IsClinic =2 or Isnull(s.IsClinic,0)=@IsClinic)

and

([SupportIssue] like Concat('%',@Keyword,'%'))

end try

begin catch

declare @ErrorNum int,

@ErrorMsg varchar(200),

@ErrorProc varchar(50),

@AddlInfo varchar(500)

set @ErrorNum = error\_number()

set @ErrorMsg = 'cms\_Support\_GetList: ' + error\_message()

set @ErrorProc = error\_procedure()

set @AddlInfo = ''

exec utl\_Insert\_ErrorLog @ErrorNum, @ErrorMsg, @ErrorProc, 'cms\_Support\_GetList', 'GET', null, @AddlInfo

end catch

ALTER PROCEDURE [dbo].[sec\_User\_DeleteListUser]

@ListId nvarchar(max),

@UserId int,

@StatusID int out

as

begin try

begin

Update u

Set u.[IsDelete] = 1

from sec\_user u (nolock)

where u.UserId IN (SELECT id FROM convert\_string\_to\_table (@ListId, ',') )

set @StatusID = 1

end

end try

begin catch