

Technologies used

The API is developed using .Net Framework and C#. The database of choice is Microsoft SQL Server. The demonstration website is developed using bootstrap and ASP.NET.

The Working Narrative

The working assumption is that the HR Manager can login to the system and generate Temporary Ids that are given to new staff. The New Staff can then access the website and register. (A new staff member does not need to login to register).

Temporary Ids have an expiry date within which they can be used. The HR sets these expiry dates when creating them. The HR can create multiple temporary Ids at the same time, but those will have the same expiration date.

Deployment Guide

1. Install and configure the Microsoft SQL Server, create a database name "StaffDB" and copy your connection string.
2. Attached is an SQL query that will create the necessary tables and stored procedures, kindly execute it.
3. Install .Net framework and open the solution using Visual Studio.
4. Restore the Project Libraries used.
5. Change the connection string in the Web.config to your copied database connection string.
6. Right click on the HRManagementAPI.asmx and set it as the Start Page
7. At this point you should be able to run the API and access the Various methods.
8. Next we run the website to further demonstrate the API.
9. Open the Webapp solution in Visual Studio.
10. Ensure the API is running, then click on "service description", copy the API link - it must be in this format "<https://localhost:44357/HRManagementAPI.asmx?WSDL>"
11. Confirm it's the same as the one for the reference in the web app, if it's different, replace it with the one you have copied to update the reference.
12. Now you can run the Webapp to test the API.
13. Sign in credentials are username: edgar and password: admin

```

USE [MyStaffDB]
GO
/***** Object: Table [dbo].[RequestLogs]      Script Date: 07/10/2024
16:40:46 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[RequestLogs](
    [RecordId] [int] IDENTITY(1,1) NOT NULL,
    [API Method] [nvarchar](20) NULL,
    [Request] [nvarchar](max) NULL,
    [IpAddress] [nvarchar](10) NULL,
    [OccuranceTime] [datetime] NULL,
    [RecordDate] [datetime] NULL
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
/***** Object: Table [dbo].[ResponseLogs]      Script Date: 07/10/2024
16:40:46 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[ResponseLogs](
    [RecordId] [int] IDENTITY(1,1) NOT NULL,
    [API Method] [nvarchar](20) NULL,
    [Response] [varchar](max) NULL,
    [OccuranceTime] [datetime] NULL,
    [RecordDate] [datetime] NULL
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
/***** Object: Table [dbo].[Staff]      Script Date: 07/10/2024 16:40:46
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Staff](
    [RecordId] [int] IDENTITY(1,1) NOT NULL,
    [StaffId] [nvarchar](20) NULL,
    [SurName] [nvarchar](50) NULL,
    [OtherNames] [nvarchar](50) NULL,
    [DOB] [datetime] NULL,
    [IdPhoto] [nvarchar](max) NULL,
    [TempIdUsed] [nchar](10) NULL,
    [RecordDate] [datetime] NULL

```

```

) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
/***** Object: Table [dbo].[SystemUsers]      Script Date: 07/10/2024
16:40:46 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[SystemUsers](
    [RecordId] [int] IDENTITY(1,1) NOT NULL,
    [UserName] [varchar](50) NULL,
    [Password] [varchar](50) NULL,
    [CreationDate] [datetime] NULL,
    [UserRole] [varchar](20) NULL,
    [SessionId] [varchar](20) NULL
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[TemporaryIds]      Script Date: 07/10/2024
16:40:46 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[TemporaryIds](
    [RecordId] [int] IDENTITY(1,1) NOT NULL,
    [CreationId] [varchar](20) NULL,
    [TemporaryId] [nvarchar](10) NULL,
    [CreationDate] [datetime] NULL,
    [ExpiryDate] [datetime] NULL
) ON [PRIMARY]
GO
/***** Object: StoredProcedure [dbo].[CreateTemporaryIds]      Script
Date: 07/10/2024 16:40:46 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =====
-- Author:      <Author,,Name>
-- Create date: <Create Date,,>
-- Description:  <Description,,>
-- =====
CREATE PROCEDURE [dbo].[CreateTemporaryIds]
    -- Add the parameters for the stored procedure here
    @count int , @durationinseconds int
AS

```

```

BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    declare @datenow datetime
    declare @creationId varchar(20), @tempId varchar(10)
    declare @createdTemps int

    set @creationId = CONVERT(varchar, getdate(),121)
    set @createdTemps = 0

    while @createdTemps < @count
    BEGIN

        set @tempId = 'TEMP' + convert(varchar, FLOOR(RAND() * (9999
- 1000 + 1)) + 1000)

        insert into TemporaryIds values (@creationId, @tempId,
getdate(), DATEADD(ss,@durationinseconds,getdate()))

        set @createdTemps = @createdTemps + 1

    END

    select TemporaryId, CreationDate, ExpiryDate, case when getdate()
> ExpiryDate then 'EXPIRED' else 'ACTIVE'end Status
    from TemporaryIds where CreationId = @creationId
END
GO
/***** Object:  StoredProcedure [dbo].[GetStaffDetails]    Script Date:
07/10/2024 16:40:46 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =====
-- Author:          <Author,,Name>
-- Create date:     <Create Date,,>
-- Description:     <Description,,>
-- =====
CREATE PROCEDURE [dbo].[GetStaffDetails]
    -- Add the parameters for the stored procedure here
    @staffId varchar(20)
AS
BEGIN

```

```

-- SET NOCOUNT ON added to prevent extra result sets from
-- interfering with SELECT statements.
SET NOCOUNT ON;

-- Insert statements for procedure here

select * from Staff where (StaffId = @staffId or @staffId = '')

END
GO
/***** Object:  StoredProcedure [dbo].[GetTemporaryIds]    Script Date:
07/10/2024 16:40:46 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =====
-- Author:      <Author,,Name>
-- Create date: <Create Date,,>
-- Description:  <Description,,>
-- =====
CREATE PROCEDURE [dbo].[GetTemporaryIds]
    -- Add the parameters for the stored procedure here
    @Status varchar(20)
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    -- Insert statements for procedure here
    ;with tablex as (
        select TemporaryId, CreationDate, ExpiryDate,
        case when t.TemporaryId in (select TempIdUsed from Staff) then
'REGISTERED'
        when getdate() > ExpiryDate then 'EXPIRED'
        else 'ACTIVE'end Status
        from TemporaryIds t )
        select * from tablex where (Status = @Status or @Status = '')
        order by ExpiryDate desc
END
GO
/***** Object:  StoredProcedure [dbo].[Login]    Script Date:
07/10/2024 16:40:46 *****/
SET ANSI_NULLS ON
GO

```

```

SET QUOTED_IDENTIFIER ON
GO
-- =====
-- Author:          <Author,,Name>
-- Create date: <Create Date,,>
-- Description:     <Description,,>
-- =====
CREATE PROCEDURE [dbo].[Login]
    -- Add the parameters for the stored procedure here
    @username varchar(50), @password varchar(20)
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    select * from SystemUsers where username = @username and Password =
@password

END
GO
/***** Object:  StoredProcedure [dbo].[LogOut]      Script Date:
07/10/2024 16:40:46 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =====
-- Author:          <Author,,Name>
-- Create date: <Create Date,,>
-- Description:     <Description,,>
-- =====
CREATE PROCEDURE [dbo].[LogOut]
    -- Add the parameters for the stored procedure here
    @SessionId varchar(20)
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    -- Insert statements for procedure here
    update SystemUsers set SessionId = '' where SessionId = @SessionId

END
GO

```

```

/***** Object:  StoredProcedure [dbo].[LogRequest]      Script Date:
07/10/2024 16:40:46 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =====
-- Author:      <Author,,Name>
-- Create date: <Create Date,,>
-- Description:  <Description,,>
-- =====
CREATE PROCEDURE [dbo].[LogRequest]
    -- Add the parameters for the stored procedure here
    @Request varchar(max), @method varchar(20), @requesttime datetime,
    @ipaddress varchar(10)
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    -- Insert statements for procedure here
    insert into RequestLogs values (@method, @Request, @ipaddress,
    @requesttime, getdate())

```

```

END
GO
/***** Object:  StoredProcedure [dbo].[LogResponse]      Script Date:
07/10/2024 16:40:46 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =====
-- Author:      <Author,,Name>
-- Create date: <Create Date,,>
-- Description:  <Description,,>
-- =====
CREATE PROCEDURE [dbo].[LogResponse]
    -- Add the parameters for the stored procedure here

    @Response varchar(max), @method varchar(20), @responsetime
datetime
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from

```

```

-- interfering with SELECT statements.
SET NOCOUNT ON;

-- Insert statements for procedure here
insert into ResponseLogs values (@method, @Response,
@responsetime, getdate())

END
GO
/***** Object: StoredProcedure [dbo].[RegisterNewStaff]    Script
Date: 07/10/2024 16:40:46 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =====
-- Author:      <Author,,Name>
-- Create date: <Create Date,,>
-- Description:  <Description,,>
-- =====
CREATE PROCEDURE [dbo].[RegisterNewStaff]
-- Add the parameters for the stored procedure here
@TemporaryId varchar(20), @Surname varchar(20), @OtherNames
varchar(20), @DOB varchar(20), @IDPicture varchar(20)

AS
BEGIN
-- SET NOCOUNT ON added to prevent extra result sets from
-- interfering with SELECT statements.
SET NOCOUNT ON;

declare @StaffId varchar(20), @recordId int

insert into Staff values ('', @Surname, @OtherNames, @DOB,
@IDPicture, @TemporaryId, getdate())

select @recordId = recordid from Staff where TempIdUsed =
@TemporaryId

set @StaffId = 'DFCU' + format(@recordId, '000000')

update Staff set StaffId = @StaffId where TempIdUsed =
@TemporaryId

select StaffId from Staff where TempIdUsed = @TemporaryId

```



```

END
GO
/***** Object:  StoredProcedure [dbo].[UpdateSessionId]    Script Date:
07/10/2024 16:40:46 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =====
-- Author:      <Author,,Name>
-- Create date: <Create Date,,>
-- Description:  <Description,,>
-- =====
CREATE PROCEDURE [dbo].[UpdateSessionId]
    -- Add the parameters for the stored procedure here
    @username varchar(50), @password varchar(20), @sessionId int
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    update SystemUsers set sessionId = @sessionId
    where username = @username
    and Password = @password
END
GO
/***** Object:  StoredProcedure [dbo].[UpdateStaffDetails]    Script
Date: 07/10/2024 16:40:46 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =====
-- Author:      <Author,,Name>
-- Create date: <Create Date,,>
-- Description:  <Description,,>
-- =====
CREATE PROCEDURE [dbo].[UpdateStaffDetails]
    -- Add the parameters for the stored procedure here
    @staffId varchar(20), @NewDOB datetime, @NewIDPhoto varchar(max),
@updateby varchar(20)
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.

```

```

        SET NOCOUNT ON;

-- Insert statements for procedure here

        declare @_NewDOB datetime, @_NewIDPhoto varchar(max)

        set @_NewDOB = case when len(@NewDOB) > 0 then @NewDOB else
(select DOB from staff where StaffId = @staffId) end
        set @_NewIDPhoto = case when len(@NewIDPhoto) > 0 then @NewIDPhoto
else (select IdPhoto from staff where StaffId = @staffId) end

        update Staff set DOB = @_NewDOB,
        IdPhoto = @_NewIDPhoto where StaffId = @staffId

END
GO
/***** Object: StoredProcedure [dbo].[ValidateSessionId]      Script
Date: 07/10/2024 16:40:46 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =====
-- Author:          <Author,,Name>
-- Create date: <Create Date,,>
-- Description:     <Description,,>
-- =====
CREATE PROCEDURE [dbo].[ValidateSessionId]
    -- Add the parameters for the stored procedure here
    @sessionId varchar(20)
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    -- Insert statements for procedure here
    select * from SystemUsers where SessionId = @sessionId
END
GO
/***** Object: StoredProcedure [dbo].[ValidateTemporaryId]      Script
Date: 07/10/2024 16:40:46 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

```

```

-- =====
-- Author:      <Author,,Name>
-- Create date: <Create Date,,>
-- Description:  <Description,,>
-- =====
CREATE PROCEDURE [dbo].[ValidateTemporaryId]
    -- Add the parameters for the stored procedure here
    @tempId varchar(10)
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;

    -- Insert statements for procedure here
    select TemporaryId, CreationDate, ExpiryDate,
        case when t.TemporaryId in (select TempIdUsed from Staff) then
'REGISTERED'
        when getdate() > ExpiryDate then 'EXPIRED'
        else 'PENDING' end Status
    from TemporaryIds t where TemporaryId = @tempId
    order by CreationDate desc
END
GO

insert into SystemUsers values ('edgar','admin',getdate(),'HR','')

```