
UNIVERSIDAD NACIONAL AUTONOMA DE HONDURAS

Facultad de Ingeniería
Departamento de Ingeniería en Sistemas

Documentación del Sistema de reportes MP (Base de Datos)

Asignatura: Ingeniería de Software

Catedrático: Ing. Miguel Saucedo

Presentado por:

Nombre	Número de Cuenta
Edson Froylan Bonilla	20081003488
Erick Alexander Zelaya	20092500066
Emil Daniel Ordoñez	20091001347
Sindy Danely García	20091004217
Osly Salinas	20081000979

Tegucigalpa M.D.C, 25 de Agosto del 2014

Tabla de contenido

Introducción	3
Scripts de procedimientos almacenados.....	4
Diccionario de Datos.....	14
Base de Datos: mpsiafireportes.....	14
Nombre Tabla: tbl_rol	14
Nombre Tabla: tbl_rol_usuario	14
Nombre Tabla: tbl_usuario	14
Nombre Tabla: tbl_apchck.....	14
Nombre Tabla: tbl_accchk.....	15
Diagrama Relacional: “mpsiafireportes”	15
Base de datos: mpsiafi	16
Nombre Tabla: bgadjm	16
Nombre Tabla: bgtrfm	16
Nombre Tabla: prempg	17
Nombre Tabla: HRJobs	17
Nombre Tabla: apchck	17
Nombre Tabla: bgbudp	18
Nombre Tabla: bgbudt.....	18
Nombre Tabla: prmisd	18
Nombre Tabla: prmisc.....	19
Nombre Tabla: prdept.....	20
Nombre Tabla: prempy	20
Diagrama Relacional: “mpsiafi”	22

Introducción

A continuación se presenta todo lo referente a la estructura de la base de datos utilizada en el proyecto y también de algunas de las tablas de la base de datos original manejada actualmente en el Ministerio Público las cuales fueron consideradas necesarias para el logro del propósito del proyecto.

De igual manera se presentan los scripts de los procedimientos almacenados utilizados para la obtención de los datos que se deseaban mostrar y almacenar.

Se muestran los diccionarios de datos de las diferentes tablas que fueron manejadas. Todo lo anterior se entrega para facilitar una posible implementación y mantenimiento futuro además para permitir fácil entendimiento de quienes hagan uso técnico del mismo.

Scripts de procedimientos almacenados

Procedimientos Almacenados para la administración y control de cuentas de Cheques

```

CREATE PROCEDURE [sp_chk_actualizar]
AS
BEGIN

    SET NOCOUNT ON;

    INSERT INTO mpsiafireportes.dbo.tbl_apchck(cvendno, cpayto, cchktype,
        cbankno, cchkno, ctogl, dcheck, dcreate, lcancel, lhold,
        nchkamt, nfchkamt, nxchgrate, nbchkamt)
        (SELECT mpchk.cvendno, mpchk.cpayto, mpchk.cchktype, mpchk.cbankno,
            mpchk.cchkno, mpchk.ctogl, mpchk.dcheck, mpchk.dcreate,
            mpchk.lcancel, mpchk.lhold, mpchk.nchkamt, mpchk.nfchkamt,
            mpchk.nxchgrate, mpchk.nbchkamt
            FROM mpsiafi.dbo.apchck AS mpchk
            WHERE mpchk.cchkno NOT IN
                (SELECT mprchk.cchkno
                 FROM mpsiafireportes.dbo.tbl_apchck AS mprchk))

    DECLARE cheques CURSOR FOR
        SELECT tchk.cchkno
            FROM mpsiafireportes.dbo.tbl_apchck AS tchk
            WHERE entregado=0;

    -- Apertura del cursor

    OPEN cheques

    -- Lectura de la primera fila del cursor
    DECLARE @num_chk VARCHAR(10);
    DECLARE @nuevo_lcancel SMALLINT;
    DECLARE @nuevo_lhold SMALLINT;

    FETCH cheques INTO @num_chk;

    WHILE (@@FETCH_STATUS = 0 ) BEGIN
        --SE ASIGNAN LOS VALORES ACTUALIZADOS DE lhold y lcancel
        SELECT @nuevo_lcancel=mpchk.lcancel,@nuevo_lhold=mpchk.lhold
            FROM mpsiafi.dbo.apchck AS mpchk
            WHERE cchkno = @num_chk;

        --SE ACTUALIZA LA TABLA tbl_apchck con los valores mas actuales
        UPDATE mpsiafireportes.dbo.tbl_apchck
            SET lcancel = @nuevo_lcancel,
                lhold = @nuevo_lhold
            WHERE mpsiafireportes.dbo.tbl_apchck.cchkno = @num_chk;

        FETCH cheques INTO @num_chk;
    END

    CLOSE cheques;
    DEALLOCATE cheques;

```

```
END
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

=====
CREATE PROCEDURE [sp_chk_entregar]
    @num VARCHAR(10),@entr VARCHAR(1)=1
AS
BEGIN

    SET NOCOUNT ON;

    UPDATE mpsiafireportes.dbo.tbl_apchck
        SET centregado = @entr
        WHERE cchkno = @num

END
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER OFF
GO

=====
CREATE PROCEDURE [sp_chk_mostrarMonto]
AS
BEGIN

    SET NOCOUNT ON;

    SELECT TOP 1 mpracchk.namt AS MONTO,mpracchk.dfechamod AS FECHA,
        mpracchk.cdasc AS 'DESC'
    FROM mpsiafireportes.dbo.tbl_accchk AS mpracchk
    ORDER BY FECHA DESC

END
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [sp_chk_montoCuenta]
    @monto NUMERIC(16,2),@desc VARCHAR(200)
AS
BEGIN

    SET NOCOUNT ON;

    INSERT INTO tbl_accchk (namt,dfechamod,cdasc)
        VALUES (@monto,GETDATE(),@desc)

END
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
```

```

=====
CREATE PROCEDURE [sp_chk_calculaDisponible]
AS
BEGIN

    SET NOCOUNT ON;

    EXEC sp_chk_actualizar;

    SELECT (SELECT TOP 1 namt AS MONTO
            FROM mpsiafireportes.dbo.tbl_accchk AS accchk
            ORDER BY accchk.dfechamod DESC) - ISNULL(SUM(mprchk.nchkamt),0)
AS DISPONIBLE
    FROM mpsiafireportes.dbo.tbl_apchck AS mprchk
    WHERE mprchk.centregado <> '0' AND mprchk.lcancel='0'

END
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

=====
CREATE PROCEDURE [sp_chk_mostrar]
    @chknun VARCHAR(10) = '', @proveedor VARCHAR(40) = '', @banco
VARCHAR(10) = '', @conciliado SMALLINT = 0,
    @retenido SMALLINT = 0, @entregado VARCHAR(1) = ''
AS
BEGIN

    SET NOCOUNT ON;

    EXEC sp_chk_actualizar;

    SELECT mprchk.cchkno AS '# CHEQUE' ,mprchk.cvendno AS 'COD. PROV.',
mprchk.cpayto AS 'DESC',
    mprchk.cbankno AS 'CTA. BANC.', mprchk.dcheck AS 'F.
CHEQUE',mprchk.dcreate AS 'F. CREACION',
    (CASE mprchk.lcancel WHEN 0 THEN 'NO' ELSE 'SI' END) AS 'CONCILIADO',
    (CASE mprchk.lhold WHEN 0 THEN 'NO' ELSE 'SI' END) AS 'RETENIDO',
mprchk.nchkamt AS CANTIDAD,
    (CASE mprchk.centregado WHEN 0 THEN 'NO' ELSE 'SI' END) AS 'ENTREGADO'
FROM mpsiafireportes.dbo.tbl_apchck AS mprchk
WHERE mprchk.cchkno LIKE '%' + @chknun + '%'
    AND mprchk.cpayto LIKE '%' + @proveedor + '%'
    AND mprchk.cbankno LIKE '%' + @banco + '%'
    AND mprchk.lcancel = @conciliado
    AND mprchk.lhold = @retenido
    AND mprchk.centregado LIKE '%' + @entregado + '%'

END
GO
ALTER TABLE [tbl_apchck] ADD DEFAULT ('0') FOR [centregado]
GO
ALTER TABLE [tbl_rol_usuario] WITH CHECK ADD CONSTRAINT
[tbl_rol_usuario_tbl_rol_FK] FOREIGN KEY([id_rol])
REFERENCES [tbl_rol] ([id_rol])
GO
ALTER TABLE [tbl_rol_usuario] CHECK CONSTRAINT [tbl_rol_usuario_tbl_rol_FK]

```

```

GO
ALTER TABLE [tbl_rol_usuario] WITH NOCHECK ADD CONSTRAINT
[tbl_rol_usuario_tbl_usuarios_FK] FOREIGN KEY([id_usuario])
REFERENCES [tbl_usuarios] ([id_usuario])
GO
ALTER TABLE [tbl_rol_usuario] CHECK CONSTRAINT [tbl_rol_usuario_tbl_usuarios_FK]
GO

```

Procedimiento Almacenado para generación de “Reporte de Empleados con cuenta presupuestaria individual”

```

CREATE PROCEDURE [dbo].[sp_reporte_empl_cuentas_presupuestarias]
    @cargo VARCHAR(40)='', @depto VARCHAR(30)='', @codpres VARCHAR(30)='',
    @est VARCHAR(2)='', @fcontrIn DATETIME=NULL, @fcontrFn DATETIME=NULL,
    @facrdIn DATETIME=NULL, @facrdFn DATETIME=NULL
AS
BEGIN
    SET NOCOUNT ON;
    IF @facrdIn='' BEGIN
        SET @facrdIn = '19000101'
    END
    IF @facrdFn='' BEGIN
        SET @facrdFn = '29000101'
    END
    IF @fcontrIn='' BEGIN
        SET @fcontrIn = '19000101'
    END
    IF @fcontrFn='' BEGIN
        SET @fcontrFn = '29000101'
    END

    SELECT empy.cempno AS CODIGO, empy.cfedid AS ID,
        empy.cfname AS NOMBRES,
        empy.clname AS APELLIDOS, empy.csex AS SEXO,
        empy.dbirth AS NACIMIENTO, empy.cdeptno AS 'COD. DEPTO',
        dept.cdeptname AS 'DEPTO. NOMBRE', jobs.cDesc AS CARGO,
        empy.nmonthpay AS SALARIO, empy.dcntrct AS 'F. CONTRATO',
        empy.dhire AS 'F. ACUERDO',
        empy.ctaxstate, empy.cpaytype AS 'T. PLANILLA',
        empy.cstate AS ESTADO, empg.cwageacc AS 'COD. PRES'
    FROM mpsiafi.dbo.prempy AS empy, mpsiafi.dbo.HRJobs AS jobs,
        mpsiafi.dbo.prdept AS dept, mpsiafi.dbo.prempg AS empg
    WHERE empy.cdeptno = dept.cdeptno
        AND empy.cjobtitle = jobs.cJobTitlNO
        AND empy.cempno = empg.cempno
        AND dhire BETWEEN ISNULL(@facrdIn, '19000101')
            AND ISNULL(@facrdFn, '29001231')
        AND dcntrct BETWEEN ISNULL(@fcontrIn, '19000101')
            AND ISNULL(@fcontrFn, '29001231')
        AND jobs.cDesc LIKE '%'+ ISNULL(@cargo, '') + '%'
        AND dept.cdeptname LIKE '%'+ ISNULL(@depto, '') + '%'
        AND empg.cwageacc LIKE '%'+ ISNULL(@codpres, '') + '%'
        AND empy.cstate LIKE '%'+ ISNULL(@est, '') + '%'

END
GO

```

Procedimiento Almacenado para generación de “Reporte de Ingreso de empleados”

```
CREATE PROCEDURE [dbo].[sp_reporte_ingreso_empleados]
AS
BEGIN
SET NOCOUNT ON;

    SELECT E.cempno AS "CODIGO", E.cfedid AS "IDENTIDAD", E.cfname AS "NOMBRES",
    E.clname AS "APELLIDOS", J.cDesc AS "CARGO", E.cdeptno AS "COD. DEPTO.",
    D.cdeptname AS "DEPTO.", M.cref AS "REFERENCIA", M.cbankno AS "BANCO",
    M.cchkno AS "CHEQUE", M.cpayno AS "PLANILLA", M.cpaycode AS "COD. INGR.",
    (M.ngty*M.nrate) AS "VALOR INGR."
    FROM mpsiafi.dbo.prempy AS E, mpsiafi.dbo.prmisc AS M, mpsiafi.dbo.prdept AS
    D, mpsiafi.dbo.HrJobs AS J
    WHERE E.cdeptno = D.cdeptno AND E.cjobtitle = J.cJobTitlNO AND E.cempno =
    M.cempno
END
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
```

Procedimiento Almacenado para generación de “Reporte de deducciones de empleado”

```
CREATE PROCEDURE [dbo].[sp_reporte_deducciones_empleados]
AS
BEGIN

    SET NOCOUNT ON;

    SELECT empl.cempno AS '# EMPLEADO', empl.cfedid AS 'IDENTIDAD',
    empl.cfname AS NOMBRES, empl.clname AS APELLIDOS,
    jobs.cDesc AS CARGO, empl.cdeptno AS 'COD. DEPTO.',
    dept.cdeptname AS 'DEPTO.', misd.cpayno AS PLANILLA,
    misd.cdedno AS 'COD. DED.', misd.cdsc AS DESCRIPCION,
    misd.ndedamt AS 'VALOR DED.'
    FROM mpsiafi.dbo.prempy AS empl
    INNER JOIN mpsiafi.dbo.HRJobs AS jobs
    ON empl.cjobtitle = jobs.cJobTitlNO
    INNER JOIN mpsiafi.dbo.prdept AS dept
    ON empl.cdeptno = dept.cdeptno
    INNER JOIN mpsiafi.dbo.prmisd AS misd
    ON empl.cempno = misd.cempno

END
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
```


Procedimiento Almacenado para generación de “Reporte de presupuesto por programa”

```

CREATE PROCEDURE [dbo].[sp_reporte_presupuesto_programa]
AS
BEGIN

    SET NOCOUNT ON;

    SELECT SUBSTRING(budget.cbudno,5,3) AS PROGRAMA ,budget.cbudid AS
PRESUPUESTO,
        (SUM(budget.namtmod) - SUM(budget.namt)) AS TRANSFERENCIA,
        SUM(budget.namtmod) AS MODIFICADO, SUM(budget.namtused) AS
UTILIZADO,
        SUM(budget.namtmod - budget.namtused) AS DISPONIBLE,
        (SUM(budget.namtused) / SUM(budget.namtmod)) AS '% UTILIZADO',
        ((SUM(budget.namtmod) - SUM(budget.namtused)) /
SUM(budget.namtmod)) AS '% DISPONIBLE'
        FROM mpsiafi.dbo.bgbudt AS budget
        WHERE SUBSTRING(budget.cbudid,5,4) = YEAR(GETDATE())
        GROUP BY SUBSTRING(budget.cbudno,5,3), budget.cbudid

END
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

```

Procedimiento Almacenado para generación de “Reporte de presupuesto por grupo”

```

CREATE PROCEDURE [dbo].[sp_reporte_presupuesto_grupo]
AS
BEGIN

    SET NOCOUNT ON;

    SELECT SUBSTRING(budget.cbudno,17,1) AS GRUPO ,budget.cbudid AS PRESUPUESTO,
        (SUM(budget.namtmod) - SUM(budget.namt)) AS TRANSFERENCIA,
        SUM(budget.namtmod) AS MODIFICADO, SUM(budget.namtused) AS
UTILIZADO,
        SUM(budget.namtmod - budget.namtused) AS DISPONIBLE,
        (SUM(budget.namtused) / SUM(budget.namtmod)) AS '% UTILIZADO',
        ((SUM(budget.namtmod) - SUM(budget.namtused)) /
SUM(budget.namtmod)) AS '% DISPONIBLE'
        FROM mpsiafi.dbo.bgbudt AS budget
        WHERE SUBSTRING(budget.cbudid,5,4) = YEAR(GETDATE())
        GROUP BY SUBSTRING(budget.cbudno,17,1), budget.cbudid

END
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

```

Procedimiento Almacenado para generación de “Reporte de presupuesto por subgrupo”

```
CREATE PROCEDURE [dbo].[sp_reporte_presupuesto_subgrupo]
AS
BEGIN
SELECT SUBSTRING(budget.cbudno,17,2) AS "SUB GRUPO",
      (budget.cbudid) AS "PRESUPUESTO",
      (SUM(budget.namtmod) - SUM(budget.namtused)) AS "TRANSFERENCIA",
      SUM(budget.namtmod) AS "MODIFICADO",
      SUM(budget.namtused) AS "UTILIZADO",
      (SUM(budget.namtmod) - SUM(budget.namtused)) AS "DISPONIBLE",
      (SUM(budget.namtused) / SUM(budget.namtmod)) AS "% UTILIZADO",
      ((SUM(budget.namtmod) - SUM(budget.namtused)) / SUM(budget.namtmod)) AS "%
DISPONIBLE"
FROM mpsiafi.dbo.bgbudt as budget
WHERE SUBSTRING(budget.cbudid,5,4) = YEAR(GETDATE())
GROUP BY SUBSTRING(budget.cbudno,17,2), budget.cbudid;
END
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
```

Procedimiento Almacenado para generación de “Reporte de presupuesto original”

```
CREATE PROCEDURE [dbo].[sp_reporte_presupuesto_compara]
AS
BEGIN

    SET NOCOUNT ON;

    SELECT budget.cbudno AS CUENTA ,budget.cbudid AS PRESUPUESTO,
           (budget.namt) AS ORIGINAL,
           (budget.namtmod) AS MODIFICADO, (budget.namtused) AS UTILIZADO,
           (budget.namtmod - budget.namtused) AS DISPONIBLE,
           ((budget.namtused) / (budget.namtmod)) AS '% UTILIZADO',
           ((budget.namtmod) - (budget.namtused)) / (budget.namtmod)) AS '%
DISPONIBLE'
    FROM mpsiafi.dbo.bgbudt AS budget
    WHERE SUBSTRING(budget.cbudid,5,4) = YEAR(GETDATE())
END
GO
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER OFF
GO
```

Procedimiento Almacenado para generación de “Reporte de transacciones”

```
CREATE PROCEDURE [dbo].[sp_reporte_presupuesto_transacciones]
AS
BEGIN

    SET NOCOUNT ON;

    ((SELECT 'TRANS.' AS TIPO, CONVERT(VARCHAR, trn.cfbudno) AS DESDE,
        trn.ctbudno AS HASTA, trn.ntrfam AS 'CANT.',
        trn.tmodrec AS FECHA
        FROM mpsiafi.dbo.bgtrfm AS trn
        WHERE YEAR(trn.tmodrec) = YEAR(GETDATE()))
    UNION
    (SELECT 'AJUST.' AS TIPO, 'N/A' AS DESDE,
        aju.cbudno AS HASTA, aju.nadjamt AS 'CANT.',
        aju.tmodrec AS FECHA
        FROM mpsiafi.dbo.bgadjm AS aju
        WHERE YEAR(aju.tmodrec) = YEAR(GETDATE()))
    )) ORDER BY FECHA

END
```

Procedimiento Almacenado para obtención de roles de usuario

```
CREATE PROCEDURE [dbo].[sp_getroles]
AS
BEGIN
    SET NOCOUNT ON;

    SELECT r.id_rol AS ROL, r.descripcion AS 'DESC'
        FROM tbl_rol AS r

END
GO
```

Procedimientos Almacenados para administración de usuarios

```
CREATE PROCEDURE [dbo].[sp_nuevousuario]
    @idusuario VARCHAR(45), @correo VARCHAR(100),
    @prnombre VARCHAR(50), @sgnombre VARCHAR(50)=NULL,
    @prapellido VARCHAR(50), @sgapellido VARCHAR(50)=NULL,
    @clave VARCHAR(50)=NULL, @rol TINYINT=3
AS
BEGIN
    -- SET NOCOUNT ON added to prevent extra result sets from
    -- interfering with SELECT statements.
    SET NOCOUNT ON;
    DECLARE @error INT;

    BEGIN TRANSACTION
        INSERT INTO tbl_usuarios (id_usuario, email, prnombre,
            sgnombre, prapellido, sgapellido, clave, fecha_creacion)
            VALUES (@idusuario, @correo, @prnombre,
                @sgnombre, @prapellido, @sgapellido,
                PWDENCRYPT(ISNULL(@clave, '123456')), GETDATE())
    END TRANSACTION
```

```

        INSERT INTO rol_usuarios (id_usuario,id_rol)
            VALUES (@idusuario,@rol)

SET @error = @@ERROR;
IF @error <> 0 BEGIN
    COMMIT
    RETURN 0;
END
ELSE BEGIN
    ROLLBACK
    RETURN @error;
END

END
GO
=====
CREATE PROCEDURE [dbo].[sp_eliminausuario]
    @correo VARCHAR(100)
AS
BEGIN

    SET NOCOUNT ON;

    DELETE tbl_usuarios
        WHERE email = @correo

END
GO

CREATE PROCEDURE [dbo].[sp_getusuarios]
    @id VARCHAR(45)='',@correo VARCHAR(100)=''
AS
BEGIN
    SET NOCOUNT ON;
    SELECT u.email AS CORREO,u.id_usuario AS ID
        FROM tbl_usuarios AS u
        WHERE u.email LIKE '%' + @correo + '%'
            AND u.id_usuario LIKE '%' + @id + '%'

END
GO
=====
CREATE PROCEDURE [dbo].[sp_login]
    @correo VARCHAR(100),@clave VARCHAR(50)
AS
BEGIN

    SET NOCOUNT ON;

    SELECT TOP 1 (usr.pnombre + ' ' + usr.papellido + ' - ' + rol.descripcion)
AS NOMBRE, rol.id_rol AS ROL
        FROM mpsiafireportes.dbo.tbl_usuarios AS usr
            INNER JOIN mpsiafireportes.dbo.tbl_rol_usuario AS rolusr
                ON usr.id_usuario = rolusr.id_usuario
            INNER JOIN mpsiafireportes.dbo.tbl_rol AS rol
                ON rol.id_rol = rolusr.id_rol
        WHERE usr.email = @correo AND PWDCOMPARE(@clave,usr.clave)=1

```

```
UPDATE mpsiafireportes.dbo.tbl_usuarios
    SET fecha_ultimo_ingreso = GETDATE()
    WHERE email = @correo

END
GO

=====
CREATE PROCEDURE [dbo].[sp_reiniciaclave]
    @correo VARCHAR(100),@clave VARCHAR(50)='123456'
AS
BEGIN

    SET NOCOUNT ON;

    UPDATE tbl_usuarios
        SET clave = PWDENCRYPT(@clave)
        WHERE email=@correo;

END
GO
```

Diccionario de Datos

Base de Datos: mpsiafireportes

Nombre Tabla: tbl_rol

No	Nombre Columna	PK	FK	M	Tipo de Dato
1	id_rol	P		Y	SMALLINT
2	descripcion				VARCHAR (45)

Nombre Tabla: tbl_rol_usuario

No	Nombre Columna	PK	FK	M	Tipo de Dato
1	id_usuario	P	F	Y	VARCHAR (45)
2	id_rol	P	F	Y	SMALLINT

Nombre Tabla: tbl_usuario

No	Nombre Columna	PK	FK	M	Tipo de Dato
1	id_usuario	P		Y	VARCHAR (45)
2	prnombre			Y	VARCHAR (50)
3	sgnombre				VARCHAR (50)
4	prapellido			Y	VARCHAR (50)
5	sgapellido				VARCHAR (50)
6	clave			Y	VARBINARY
7	fecha_creacion				Datetime
8	fecha_ultimo_ingreso				Datetime
9	email				VARCHAR (100)
10	id_departamento			Y	Integer

Nombre Tabla: tbl_apchck

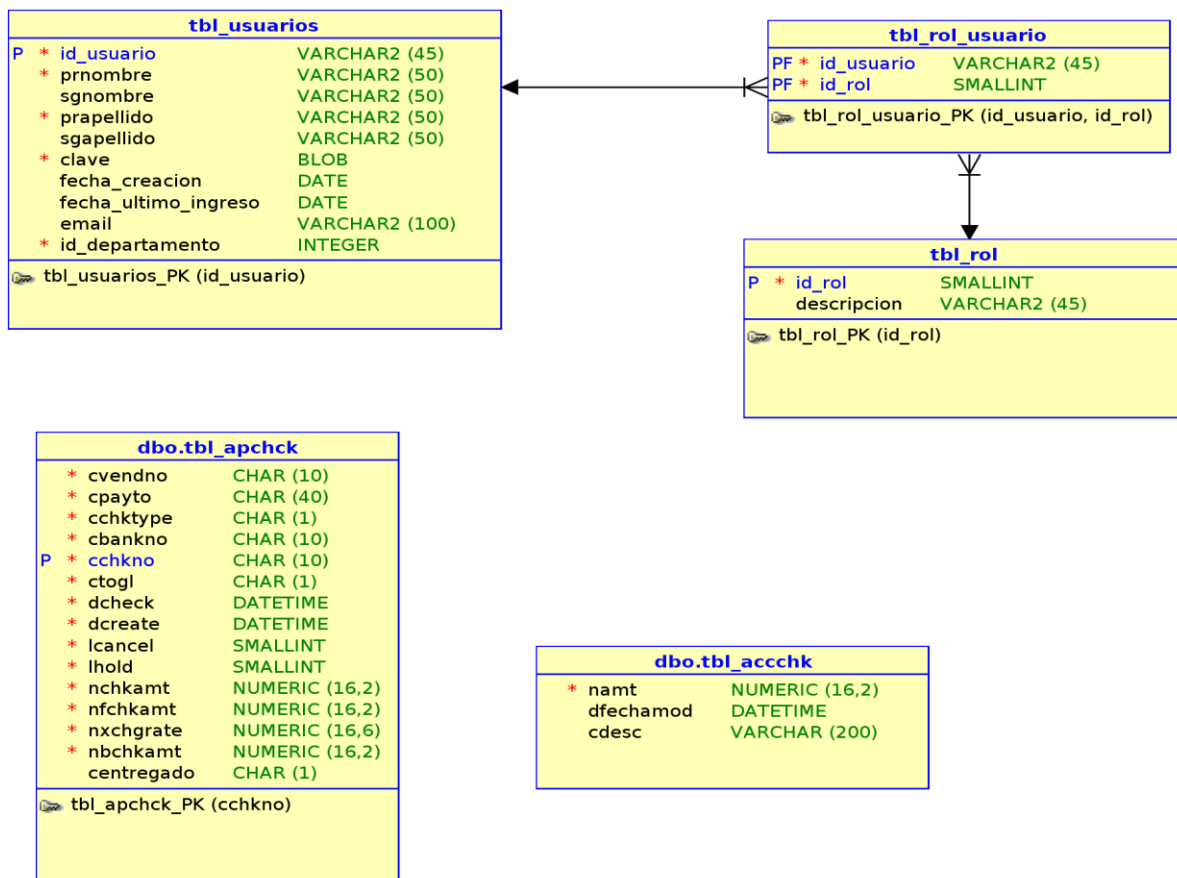
No	Nombre Columna	PK	M	Tipo de Dato
1	cvendno		Y	CHAR(10)
2	cpayto		Y	CHAR(40)
3	cchktype		Y	CHAR(1)
4	cbankno		Y	CHAR(10)
5	cchkno	P	Y	CHAR(10)

No	Nombre Columna	PK	M	Tipo de Dato
6	ctogl		Y	CHAR(1)
7	dcheck		Y	Datetime
8	dcreate		Y	Datetime
9	lcancel		Y	SMALLINT
10	lhold		Y	SMALLINT
11	nchkamt		Y	NUMERIC(16,2)
12	nfchkamt		Y	NUMERIC(16,2)
13	nxchgrate		Y	NUMERIC(16,6)
14	nbchkamt		Y	NUMERIC(16,2)
15	centregado		Y	CHAR(1)

Nombre Tabla: tbl_accchk

No	Nombre Columna	M	Tipo de Dato
1	namt	Y	NUMERIC(16,2)
2	dfechamod	Y	DATETIME
3	cdesc	Y	VARCHAR(200)

Diagrama Relacional: “mpsiafireportes”



Base de datos: mpsiafi

Nombre Tabla: bgadjm

No	Nombre Columna	M	Tipo de Dato
1	cadjno	Y	CHAR
2	cbudid	Y	CHAR
3	cbudno	Y	CHAR
4	cappcode	Y	CHAR
5	cdesc	Y	CHAR
6	dtrs		Datetime
7	cuser	Y	CHAR
8	nadjamt	Y	NUMERIC
9	nadjqty	Y	NUMERIC
10	mnotepad	Y	TEXT
11	tmodrec	Y	Datetime
12	lvoid	Y	SMALLINT
13	dvoid		Datetime

Nombre Tabla: bgtrfm

No	Nombre Columna	M	Tipo de Dato
1	ctrfno		CHAR
2	cbudid		CHAR
3	cfbudno		CHAR
4	ctbudno		CHAR
5	cappcode		CHAR
6	cdesc		CHAR
7	dtrs		Datetime
8	cuser		CHAR
9	ntrfamt		NUMERIC
10	ntrfqty		NUMERIC
11	mnotepad		TEXT
12	tmodrec		Datetime
13	lvoid		SMALLINT
14	dvoid		Datetime

Nombre Tabla: prempg

No	Nombre Columna	M	Tipo de Dato
1	cuid	Y	CHAR
2	cempno	Y	CHAR
3	cwageacc	Y	CHAR
4	ctaxacc	Y	CHAR
5	ndistpct	Y	NUMERIC
6	cfwtacc	Y	CHAR
7	cficaacc	Y	CHAR
8	cmediacc	Y	CHAR
9	cfutaacc	Y	CHAR
10	csutaacc	Y	CHAR
11	cficaeracc	Y	CHAR
12	cmedieracc	Y	CHAR
13	cfutaeracc	Y	CHAR
14	csutaeracc	Y	CHAR

Nombre Tabla: HRJobs

No	Nombre Columna	M	Tipo de Dato
1	cJobTitlNO	Y	VARCHAR
2	cDesc	Y	VARCHAR
3	nSalary	Y	DECIMAL
4	nPorPlus	Y	DECIMAL
5	mnnotepad		VARCHAR
6	tmodrec	Y	Datetime

Nombre Tabla: apchck

No	Nombre Columna	M	Tipo de Dato
1	cvendno	Y	CHAR
2	cpayto	Y	CHAR
3	cchktype	Y	CHAR
4	cbankno	Y	CHAR
5	cchkno	Y	CHAR
6	ctogl	Y	CHAR
7	dcheck	Y	Datetime
8	dcreate	Y	Datetime
9	lcancel	Y	SMALLINT
10	lhold	Y	SMALLINT
11	nchkamt	Y	NUMERIC
12	nfchkamt	Y	NUMERIC
13	nxchgrate	Y	NUMERIC
14	nbchkamt	Y	NUMERIC

Nombre Tabla: bgbudp

No	Nombre Columna	M	Tipo de Dato
1	cuid	Y	CHAR
2	cbudid	Y	CHAR
3	cbudno	Y	CHAR
4	cyear	Y	CHAR
5	cpdno	Y	CHAR
6	namt	Y	NUMERIC
7	namtused	Y	NUMERIC
8	nqty	Y	NUMERIC
9	nqtyused	Y	NUMERIC
10	tmodrec	Y	Datetime
11	namtaccr	Y	NUMERIC
12	nqtyaccr	Y	NUMERIC
13	namtmod	Y	NUMERIC
14	nqtypmod	Y	NUMERIC

Nombre Tabla: bgbudt

No	Nombre Columna	M	Tipo de Dato
1	cuid	Y	CHAR
2	cbudid	Y	CHAR
3	cbudno	Y	CHAR
4	cdesc	Y	CHAR
5	cstmacct	Y	CHAR
6	crsvacct	Y	CHAR
7	cclsid	Y	CHAR
8	ctypid	Y	CHAR
9	cdeptno	Y	CHAR
10	namt	Y	NUMERIC
11	namtused	Y	NUMERIC
12	nqty	Y	NUMERIC
13	nqtyused	Y	NUMERIC
14	tmodrec	Y	Datetime
15	namtaccr	Y	NUMERIC
16	nqtyaccr	Y	NUMERIC
17	namtmod	Y	NUMERIC
18	nqtypmod	Y	NUMERIC

Nombre Tabla: prmsid

No	Nombre Columna	M	Tipo de Dato
1	cuid	Y	CHAR
2	cempno	Y	CHAR
3	cdedno	Y	CHAR

No	Nombre Columna	M	Tipo de Dato
4	cdedcode	Y	CHAR
5	cpayno	Y	CHAR
6	ndedamt	Y	NUMERIC
7	tmodrec	Y	Datetime
8	lsystem	Y	SMALLINT
9	cdesc	Y	CHAR
10	ctrsno	Y	CHAR

Nombre Tabla: pmisc

No	Nombre Columna	M	Tipo de Dato
1	cuid	Y	CHAR
2	cempno		CHAR
3	clname		CHAR
4	cfname		CHAR
5	ctaxstate		CHAR
6	cdeptno		CHAR
7	cpaytype		CHAR
8	cref		CHAR
9	ctrsno		CHAR
10	cbankno		CHAR
11	cchkno		CHAR
12	dtrs		Datetime
13	dcheck		Datetime
14	ltaxable		SMALLINT
15	nohtax		NUMERIC
16	nohtntax		NUMERIC
17	csegid1		CHAR
18	csegid2		CHAR
19	csegid3		CHAR
20	csegid4		CHAR
21	csegid5		CHAR
22	csegmid		CHAR
23	ccycle		CHAR
24	cpayno		CHAR
25	cpaycode		CHAR
26	nqty		NUMERIC

No	Nombre Columna	M	Tipo de Dato
27	nrrate		NUMERIC
28	lssystem		SMALLINT
29	nothertax		NUMERIC
30	nBaseGrTax		NUMERIC
31	cdocno		CHAR
32	cdoctyp		CHAR

Nombre Tabla: prdept

No	Nombre Columna	M	Tipo de Dato
1	cdeptno	Y	CHAR
2	cdeptname	Y	CHAR
3	cwageacc	Y	CHAR
4	ctaxacc	Y	CHAR
5	cpaytype	Y	CHAR
6	cpayperiod	Y	CHAR
7	lallowot		SMALLINT
8	lepayment		SMALLINT
9	lusetips		SMALLINT
10	nrhr	Y	NUMERIC
11	cficaeracc	Y	CHAR
12	cmedieracc	Y	CHAR
13	cfutaeracc	Y	CHAR
14	csutaeracc	Y	CHAR
15	cficaacc	Y	CHAR
16	cmediacc	Y	CHAR
17	cfutaacc	Y	CHAR
18	csutaacc	Y	CHAR
19	cdeptglseg	Y	CHAR

Nombre Tabla: prempy

No	Nombre Columna	M	Tipo de Datos
1	compno	Y	CHAR
2	cfname	Y	CHAR
3	clname	Y	CHAR
4	cinitial	Y	CHAR
5	cssn		CHAR
6	cdeptno	Y	CHAR
7	cstatus	Y	CHAR
8	cfedid		CHAR
9	caddr1	Y	CHAR

No	Nombre Columna	M	Tipo de Datos
10	caddr2	Y	CHAR
11	ccity	Y	CHAR
12	cstate	Y	CHAR
13	czip	Y	CHAR
14	cphone	Y	CHAR
15	cjobtitle	Y	CHAR
16	csex	Y	CHAR
17	ccitizen	Y	CHAR
18	ceeoid	Y	CHAR
19	cethniccd	Y	CHAR
20	cpaytype	Y	CHAR
21	cpayperiod	Y	CHAR
22	cmarry	Y	CHAR
23	cmttype	Y	CHAR
24	ceicstatus	Y	CHAR
25	ctaxstate	Y	CHAR
26	csutastate	Y	CHAR
27	cltaxstate	Y	CHAR
28	cltaxcd	Y	CHAR
29	capplycd	Y	CHAR
30	cprenote	Y	CHAR
31	cw2year	Y	CHAR
32	dhire		Datetime
33	dterminate		Datetime
34	dbirth		Datetime
35	deff		Datetime
36	dnextpf		Datetime
37	dnextsal		Datetime
38	dlastpaid		Datetime
39	dprenote		Datetime
40	lallowot		SMALLINT
41	lepayment		SMALLINT
42	lusetips		SMALLINT
43	lfica		SMALLINT
44	lmedi		SMALLINT
45	lfuta		SMALLINT
46	lsuta		SMALLINT
47	lsdi		SMALLINT
48	leic		SMALLINT
49	lftax		SMALLINT

No	Nombre Columna	M	Tipo de Datos
50	lstax		SMALLINT
51	lsres		SMALLINT
52	lltax		SMALLINT
53	llres		SMALLINT
54	lapply		SMALLINT
55	nbpayrate	Y	NUMERIC
56	napayrate	Y	NUMERIC
57	nrhr	Y	NUMERIC
58	nafwt	Y	NUMERIC
59	nfallow	Y	NUMERIC
60	naswt	Y	NUMERIC
61	nsallow	Y	NUMERIC
62	nalwt	Y	NUMERIC
63	csrcid	Y	CHAR
64	cplnid	Y	CHAR
65	cshfid	Y	CHAR
66	cmssc1id	Y	CHAR
67	cmssc2id	Y	CHAR
68	cmssc3id	Y	CHAR
69	cmssc4id	Y	CHAR
70	cmssc5id	Y	CHAR
71	csegid1	Y	CHAR
72	csegid2	Y	CHAR
73	csegid3	Y	CHAR
74	csegid4	Y	CHAR
75	csegid5	Y	CHAR
76	lPaySlip	Y	SMALLINT
77	ccustno	Y	CHAR
78	cdayoff	Y	CHAR
79	cbatchid	Y	CHAR
80	nmonthpay	Y	NUMERIC
81	nlstmthpay	Y	NUMERIC
82	ltemporal	Y	SMALLINT
83	nfixover	Y	NUMERIC
84	crtm	Y	CHAR
85	dcntrct		Datetime
86	laddtax1	Y	SMALLINT
87	laddtax2	Y	SMALLINT
88	laddtax3	Y	SMALLINT
89	ltax11mth	Y	SMALLINT

Diagrama Relacional: “mpsiafi”

tbl_prempy	
* cempno	CHAR
* cfname	CHAR
* clname	CHAR
* cinitial	CHAR
* cssn	CHAR
* cdeptno	CHAR
* cstatus	CHAR
* cdedid	CHAR
* caddr1	CHAR
* caddr2	CHAR
* ccity	CHAR
* cstate	CHAR
* czip	CHAR
* cphone	CHAR
* cjobtitle	CHAR
* csex	CHAR
* ccitizen	CHAR
* ceoid	CHAR
* cethniccd	CHAR
* cpaytype	CHAR
* cpayperiod	CHAR
* cmarry	CHAR
* cmtype	CHAR
* ceicstatus	CHAR
* ctaxstate	CHAR
* csutastate	CHAR
* cltaxstate	CHAR
* cltaxcd	CHAR
* capplycd	CHAR
* cprenote	CHAR
* cw2year	CHAR
* dhire	DATE
* dterminate	DATE
* dbirth	DATE
* deff	DATE
* dnextpf	DATE
* dnextsal	DATE
* dlastpaid	DATE
* dprenote	DATE
* lallowot	SMALLINT
* lepayment	SMALLINT
* lusetips	SMALLINT
* lfica	SMALLINT
* lmedi	SMALLINT
* lfuta	SMALLINT
* lsuta	SMALLINT
* lsdi	SMALLINT
* leic	SMALLINT
* lftax	SMALLINT
* lstax	SMALLINT
* lsres	SMALLINT
* lltax	SMALLINT
* llres	SMALLINT
* lapply	SMALLINT
* nbpayrate	NUMBER
* napayrate	NUMBER
* nrhr	NUMBER
* nafvt	NUMBER
* nfallow	NUMBER
* nasvt	NUMBER
* nsallow	NUMBER
* nalwt	NUMBER
* csrctid	CHAR
* cplnid	CHAR
* cshfid	CHAR
* cmisc1id	CHAR
* cmisc2id	CHAR
* cmisc3id	CHAR
* cmisc4id	CHAR
* cmisc5id	CHAR
* csegid1	CHAR
* csegid2	CHAR
* csegid3	CHAR
* csegid4	CHAR
* csegid5	CHAR
* lPaySlip	SMALLINT
* ccustno	CHAR
* cdayoff	CHAR
* cbatchid	CHAR
* nmonthpay	NUMBER
* nlstnmpay	NUMBER
* ltemporal	SMALLINT
* nfixover	NUMBER
* crtn	CHAR
* dcnttrct	DATE
* laddtax1	SMALLINT
* laddtax2	SMALLINT
* laddtax3	SMALLINT
* ltax11mth	SMALLINT

tbl_prmisc	
* cuid	CHAR
* cempno	CHAR
* clname	CHAR
* cfname	CHAR
* ctaxstate	CHAR
* cdeptno	CHAR
* cpaytype	CHAR
* cref	CHAR
* ctrsno	CHAR
* cbankno	CHAR
* cchkno	CHAR
* dtrs	DATE
* dcheck	DATE
* ltaxable	SMALLINT
* nothtax	NUMBER
* nothntax	NUMBER
* csegid1	CHAR
* csegid2	CHAR
* csegid3	CHAR
* csegid4	CHAR
* csegid5	CHAR
* csegmid	CHAR
* ccycle	CHAR
* cpayno	CHAR
* cpaycode	CHAR
* nqty	NUMBER
* nrte	NUMBER
* lsystem	SMALLINT
* nothertax	NUMBER
* nBaseGrTax	NUMBER
* cdocno	CHAR
* cdoctyp	CHAR

tbl_prdept	
* cdeptno	CHAR
* cdeptname	CHAR
* cwageacc	CHAR
* ctaxacc	CHAR
* cpaytype	CHAR
* cpayperiod	CHAR
* lallowot	SMALLINT
* lepayment	SMALLINT
* lusetips	SMALLINT
* nrhr	NUMBER
* cficaeracc	CHAR
* cmedieracc	CHAR
* cfutaeracc	CHAR
* csutaeracc	CHAR
* cficaacc	CHAR
* cmediacc	CHAR
* cfutaacc	CHAR
* csutaacc	CHAR
* cdeptglseg	CHAR

tbl_bgbudt	
* cuid	CHAR
* cbudid	CHAR
* cbudno	CHAR
* cdesc	CHAR
* cstmacct	CHAR
* crsvacct	CHAR
* cclsid	CHAR
* ctypid	CHAR
* cdeptno	CHAR
* namt	NUMBER
* namtused	NUMBER
* nqty	NUMBER
* nqtyused	NUMBER
* tmodrec	DATE
* namtaccr	NUMBER
* nqtyaccr	NUMBER
* namtmod	NUMBER
* nqtypmod	NUMBER

bgadjm	
* cadjno	CHAR
* cbudid	CHAR
* cbudno	CHAR
* cappcode	CHAR
* cdesc	CHAR
* dtrs	DATE
* cuser	CHAR
* nadjamt	NUMBER
* nadjqty	NUMBER
* mnotepad	CLOB
* tmodrec	DATE
* lvoid	SMALLINT
* dvoid	DATE

prempg	
* cuid	CHAR
* cempno	CHAR
* cwageacc	CHAR
* ctaxacc	CHAR
* ndistpct	NUMBER
* cfwacc	CHAR
* cficaacc	CHAR
* cmediacc	CHAR
* cfutaacc	CHAR
* csutaacc	CHAR
* cficaeracc	CHAR
* cmedieracc	CHAR
* cfutaeracc	CHAR
* csutaeracc	CHAR

bgtrfm	
ctrfno	CHAR
cbudid	CHAR
cfbudno	CHAR
ctbudno	CHAR
cappcode	CHAR
cdesc	CHAR
dtrs	DATE
cuser	CHAR
ntrfamt	NUMBER
ntrfqty	NUMBER
mnotepad	CLOB
tmodrec	DATE
lvoid	SMALLINT
dvoid	DATE

tbl_apchck	
* cvendno	CHAR
* cpayto	CHAR
* cchktype	CHAR
* cbankno	CHAR
* cchkno	CHAR
* ctogl	CHAR
* dcheck	DATE
* dcreate	DATE
* lcancel	SMALLINT
* lhold	SMALLINT
* nchkamt	NUMBER
* nfchkamt	NUMBER
* nxchgrate	NUMBER
* nbchkamt	NUMBER

tbl_bgbudp	
* cuid	CHAR
* cbudid	CHAR
* cbudno	CHAR
* cyear	CHAR
* cpdno	CHAR
* namt	NUMBER
* namtused	NUMBER
* nqty	NUMBER
* nqtyused	NUMBER
* tmodrec	DATE
* namtaccr	NUMBER
* nqtyaccr	NUMBER
* namtmod	NUMBER
* nqtypmod	NUMBER

tbl_HRJobs	
* cJobTitINO	VARCHAR2
* cDesc	VARCHAR2
* nSalary	NUMBER
* nPorPlus	NUMBER
* mnotepad	VARCHAR2
* tmodrec	DATE

tbl_prmisd	
* cuid	CHAR
* cempno	CHAR
* cdedno	CHAR
* cdedcode	CHAR
* cpayno	CHAR
* ndedamt	NUMBER
* tmodrec	DATE
* lsystem	SMALLINT
* cdesc	CHAR
* ctrsno	CHAR