

2018

SISTEMA DE APOYO AL CONTROL DE ASISTENCIA, RESERVA DE SALAS Y CONTROL DE HORAS EXTRAS.

VALDERRAMA ENCELEGRE, CONSTANZA DANIELA

<http://hdl.handle.net/11673/40180>

Repositorio Digital USM, UNIVERSIDAD TECNICA FEDERICO SANTA MARIA

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
SEDE VIÑA DEL MAR – JOSÉ MIGUEL CARRERA

**SISTEMA DE APOYO AL CONTROL DE ASISTENCIA, RESERVA DE SALAS Y
CONTROL DE HORAS EXTRAS**

Trabajo de Titulación para optar al
Título de Técnico Universitario en
INFORMÁTICA

Alumna:

Constanza Daniela Valderrama
Encelegre.

Profesora Guía:

Sra. Catherine Gómez B.

RESUMEN

KEYWORDS: control de asistencias, reserva de sala, horas extras.

En éste texto, se presenta el sistema informático denominado "Sistema de Apoyo al control de Asistencias, reserva de salas y control de horas extras", basado en las necesidades que presentaba una de las oficinas la empresa DIMENSION S.A ubicada en Santiago, específicamente en Av. Libertador Bernardo O'Higgins 1146, 5° piso. El sistema, automatiza el ingreso del personal a la oficina por medio de un microcontrolador arduino UNO con un lector RFIDy permite tener un control con respecto a la reserva de la sala de reuniones utilizando la interfaz proporsionada para la secretaria, también genera automáticamente las horas extras del personal de dicha oficina.

Para el desarrollo de este sistema, se establecieron tres capítulos, en el primer capítulo se define la descripción de la organización, la situación actual, los problemas detectados y el modelo de datos. Para proceder a establecer las bases del sistema propuesto y las entidades que existen en él.

En el segundo capítulo se describen las características del software realizado y del hardware donde va a ser implementado el sistema, además se describen todas las tablas expuestas en el modelo de datos y sus campos.

En el tercer capítulo, se especifica el diseño del sistema, la interfaz que tendrá el usuario, el diagrama modular y la especificación de programas.

Finalmente, se presentan las conclusiones del trabajo, la bibliografía y los Anexos en donde se expone parte del código fuente del sistema.

ÍNDICE

RESUMEN

INTRODUCCIÓN 1

CAPÍTULO 1: ASPECTOS RELEVANTES DEL DISEÑO LÓGICO. 3

1.ASPECTOS RELEVANTES DEL DISEÑO LÓGICO. 5

1.1. DESCRIPCIÓN DE LA ORGANIZACIÓN 5

1.2 DESCRIPCIÓN DE LA SITUACION ACTUAL. 6

1.3 PROBLEMAS DETECTADOS. 8

1.4 DESCRIPCIÓN DEL SISTEMA PROPUESTO. 9

1.4.1 Objetivo Principal..... 9

1.4.2 Objetivos específicos 9

1.4.3 Beneficios del sistema. 10

1.4.4 Descripción general de la solución propuesta..... 10

1.4.5 Diagrama de flujo administrativo 11

1.4.6 Estructura funcional del sistema 11

1.4.7 Descripción de fórmulas y cálculos empleados 13

1.4.8 Información que se manejará 14

1.4.9. Estructura de códigos 16

1.4.10. Condicionantes de diseño..... 17

CAPÍTULO 2:MEDIO AMBIENTE COMPUTACIONAL Y DESCRIPCIÓN DE ARCHIVO.....19

2. MEDIO AMBIENTE COMPUTACIONAL Y DESCRIPCIÓN DE ARCHIVOS. 21

2.1 DESCRIPCIÓN DE RECURSO COMPUTACIONAL. 21

2.2 DESCRIPCIÓN DE ARCHIVOS. 22

CAPÍTULO 3: Descripción de programas..... 31

3. Descripción de programas. 33

3.1 Diagrama Modular..... 33

3.2 Diagrama de menús. 34

3.2.1 Administrador 34

3.3 Programas del Sistema 35

3.4 Descripción de Programas 36

3.4.1 Menú Principal.(Codigo fuente Pag. 53 en Anexos) 36

3.4.2 Inicio Sesión. (Codigo fuente Pag. 57 en Anexos) 37

3.4.3 Marca Ingreso/Salida. (Codigo fuente Pag. 59 en Anexos) 38

3.4.4 Ingreso Personal. (Codigo fuente Pag. 62 en Anexos) 40

3.4.5 Consultar Personal. (Codigo fuente Pag. 72 en Anexos) 42

3.4.6 Ingreso Reserva de Sala. (Codigo fuente Pag. 74 en Anexos) 43

3.4.7 Ingreso Justificación. (Codigo fuente Pag. 78 en Anexos) 44

3.4.8 Consultar Tarjetas. (Codigo fuente Pag. 82 en Anexos) 46

CONCLUSIÓN..... 49

BIBLIOGRAFÍA 51

ANEXOS – PROGRAMAS FUENTE..... 53

ÍNDICE DE DIAGRAMAS

Diagrama 3-1. Diagrama Modular. 33

Diagrama 3-2. Diagrama menú para Usuario Administrador. 34

Diagrama 3-3. Diagrama Menú Principal..... 36

Diagrama 3-4. Diagrama de inicio sesión..... 37

Diagrama 3-5. Diagrama de marca entrada/salida. 39

Diagrama 3-6. Diagrama de ingreso personal..... 40

Diagrama 3-7. Diagrama consultar personal. 42

Diagrama 3-8. Diagrama ingresar reserva de sala 43

Diagrama 3-9. Diagrama ingreso justificación. 45

Diagrama 3-10. Diagrama consultar tarjetas..... 46

ÍNDICE DE TABLAS

Tabla 2-1 Descripción de campos tabla Personal..... 22

Tabla 2-2 Descripción de campos tabla tarjetas. 23

Tabla 2-3 Descripción de campos tabla Marca entrada salida..... 24

Tabla 2-4 Descripción de campos tabla oficina..... 24

Tabla 2-5 Descripción de campos tabla cargos. 25

Tabla 2-6 Descripción de campos tabla Días..... 26

Tabla 2-7 Descripción de campos tabla Usuarios..... 26

Tabla 2-8 Descripción de campos tabla horas trabajadas.	27
Tabla 2-9 Descripción de campos tabla Horario.....	28
Tabla 2-10 Descripción de campos tabla Reserva de sala.	29
Tabla 2-11 Descripción de campos tabla tarjetas personal.	29
Tabla 2-12 Descripción de campos tabla Justificación.	30
Tabla 3-1 Descripción de Programas del Sistema.	35

ÍNDICE DE FIGURAS

Figura 1-1 Libro de asistencia, oficina área desarrollo y QA para Perú y Colombia ...	7
Figura 1-2 Plantilla Excel, cálculo horas extras.	8
Figura 1-3 Diagrama de flujo administrativo – Control de asistencia.....	11
Figura 1-4 Modelo Relacional de Datos.	16
Figura 3-1 Menú Principal	37
Figura 3-2 Pantalla Ingreso.	38
Figura 3-3 Pantalla Ingreso.	38
Figura 3-4 Pantalla Inicio Menú Principal.....	39
Figura 3-5 Pantalla agregar personal-Recuperar datos.....	41
Figura 3-6 Pantalla agregar personal.....	41
Figura 3-7 Pantalla consultar personal.....	43
Figura 3-8 Pantalla Reserva Salas.....	44
Figura 3-9 Pantalla Reserva Salas.....	44
Figura 3-10 Pantalla justificar inasistencia.....	45
Figura 3-11 Pantalla justificar inasistencia.....	46
Figura 3-12 Pantalla Consultar Tarjetas.	47

INTRODUCCIÓN

La empresa DIMENSION S.A, ubicada en Santiago de Chile, fue fundada el año 2010 por Adio Stefony y Francisco Stefony junto a sus accionistas, teniendo como misión dar soluciones informáticas a la medida de necesidades reales de las empresas quiénes solicitasen sus servicios.

Esta empresa consta con diversas oficinas ubicadas en el centro de Santiago. Al hacer ingreso a una de éstas, específicamente la oficina ubicada en Av. Libertador Bernardo O'Higgins 1146, sector A - 5° piso, presenta una notoria falta de un sistema informático que apoye el trabajo de la secretaria, ya que, para registrar el ingreso de una persona usan un libro de asistencia y algunos de los procesos que realizan son bastante informales, como por ejemplo, para utilizar la sala donde realizan reuniones, se preguntan entre ellos si ésta está ocupada, y cada uno de los desarrolladores escribe sus horas extras del mes en una plantilla Excel que se les hace entrega.

El siguiente proyecto, propone un sistema informático que tiene como objetivo automatizar el ingreso general del personal de la oficina y dar un mayor control con la reserva de la sala de reuniones, también abarcando el cálculo automático de las horas extras de cada persona.

Para esto se proporcionará un lector de tarjetas y el programa que será administrado por la secretaria, donde tendrá las opciones que pueden ayudar a su trabajo cotidiano y la fluidez general de la oficina.

CAPÍTULO 1: ASPECTOS RELEVANTES DEL DISEÑO LÓGICO.

1. ASPECTOS RELEVANTES DEL DISEÑO LÓGICO.

1.1. DESCRIPCIÓN DE LA ORGANIZACIÓN.

El presente sistema está basado en una de las oficinas de la empresa "DIMENSION S.A", ubicada en Santiago centro, fundada por Adio Stefony y Francisco Stefony junto a sus accionistas David Flores, Mauricio del Río y Juan Carlos Villanueva el año 2010.

Su misión es, participar en la industria de Tecnologías de la Información, con soluciones a propósito y a la medida de las reales necesidades de las empresas grandes, medianas o pequeñas.

El campo de acción de DIMENSION S.A. se circunscribe a todas aquellas materias donde el uso y experiencia del conocimiento aplicado en Tecnologías de la Información, permite entregar soluciones que aporten un valor agregado a determinadas áreas o a solucionar determinados problemas dentro de una organización.

Entregan un servicio integral desde las etapas de levantamiento, diseño, construcción y desarrollo, hasta la implantación y puesta en marcha de una solución. Además pueden aportar el equipamiento y configuraciones tecnológicas (software, hardware y comunicaciones) para el desarrollo del proyecto.

Algunos servicios son:

- Análisis de Requerimientos.
- Desarrollo o Construcción de aplicaciones.
- Capacitación.
- Mantención.
- Elaboración de planes informáticos.
- Definición de Requerimientos para redes locales.
- Desarrollo de sitios web.

Actualmente, la empresa cuenta con un aproximado de noventa empleados y tiene algunos clientes como:

- Banco Itaú Corpbanca.
- Banco BBVA.
- Banco financiero de Perú.

Proyectos desarrollados en el área de informática:

- Manejo de créditos bancarios e inversiones en fondos mutuos.
- Sistema de facturación.
- Carpetas Digitales.
- Sistema de contabilidad.
- Entre otros.

1.2 DESCRIPCIÓN DE LA SITUACION ACTUAL.

Al ingresar a una de las oficinas de la empresa DIMENSION S.A, específicamente del área de desarrollo y QA para algunos proyectos de Perú y Colombia, ubicada en Av. Libertador Bernardo O'Higgins 1146, Sector A - 5° piso, se encuentra inmediatamente el escritorio de la secretaria donde se encuentra el libro de asistencia.

Muchas veces el personal de aseo es quien se encarga de supervisar el libro de asistencia, ya que, la secretaria podría ingresar más tarde.

Durante el día, las reuniones de equipos y conferencias con clientes son bastantes frecuentes, por lo que también es común que alguien pregunte si usarán la sala de reuniones.

Con relación a las horas extras, cada empleado, diferente de un jefe de proyecto, completa una plantilla Excel donde ingresa sus horas trabajadas en el mes.

A continuación, se presentarán los procesos de relevancia para el trabajo:

a) Proceso control de asistencia

Actualmente el ingreso a la oficina ya mencionada, carece de un sistema automatizado, es decir, se basa en un registro manual de asistencia, como se muestra en la Figura 1-1.

Cuando el personal hace ingreso, se debe registrar en el libro, indicando nombre, hora y firma. Este control es administrado por la secretaria o cualquier personal que se encuentre disponible, por ejemplo, personal de aseo, las salidas o ingresos durante el día no se registran, es decir, solo se registra la hora de entrada y su salida de la jornada laboral.

El libro, al término del día, es enviado a RRHH y devuelto para la mañana siguiente.

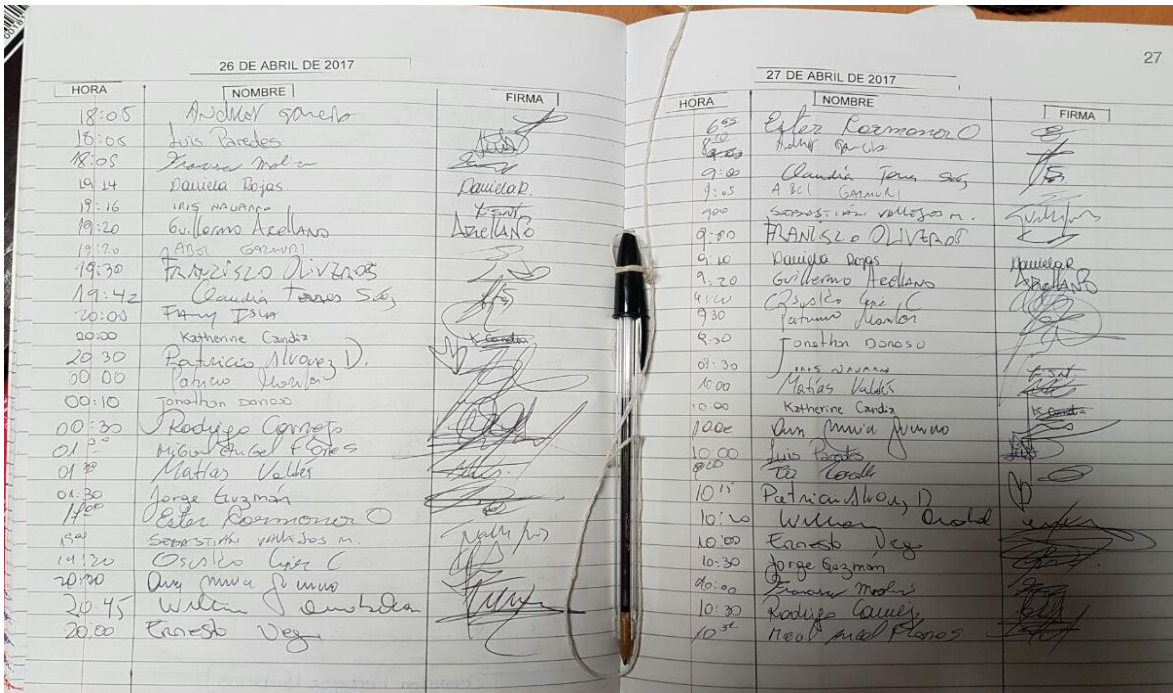


Figura 1-1 Libro de asistencia, oficina área desarrollo y QA para Perú y Colombia.

Fuente: Luis Paredes, miembro del equipo de desarrollo para Colombia.

b) Proceso conteo horas extras

Todo personal bajo el cargo del jefe de proyecto debe entregar una plantilla Excel que se les entrega cada mes, en un día previamente definido, con todas sus horas extras realizadas hasta la fecha especificada.

En esta plantilla, completan las casillas con sus horas de salidas y el motivo de éstas para calcular el valor.

El documento de cada uno es entregado a su jefe de proyecto, para que luego éste lo envíe a RRHH.

A continuación, una imagen de cómo es actualmente:

Horas Extra Periodo:						Fecha:
Nombre:						
Fecha	Hora Entrada/Salida	Desde	Hasta	Cantidad de Horas	Justificación	
03/03/2017	9:20	9:20	10:00	0:40:00	Trabajo diario	
06/03/2017	9:30	10:00	10:00	0:00:00	Trabajo diario	
07/03/2017	8:30	8:00	9:00	1:00:00	Trabajo diario	
08/03/2017	20:00	20:00	20:40	0:40:00	Trabajo diario	
16/03/2017	9:30	10:00	10:00	0:00:00	Trabajo diario	
17/03/2017	8:30	8:00	10:00	2:00:00	Query consulta	
18/03/2017	12:00	12:00	18:30	6:30:00	Soporte a Phoenix / Query consulta	
				10:50:00	Total horas	

Figura 1-2 Plantilla Excel, cálculo horas extras.

Fuente: Luis Paredes, miembro del equipo de desarrollo para Colombia.

c) Solicitud sala de reunión

A la fecha no existe registro del uso de la sala de reunión, por lo cual, el personal se ve obligado a consultar a la secretaria o miembros de otro equipo para saber si la sala se encuentra disponible o no, provocando confusión y desorden dentro de la oficina.

El personal debería estar al tanto de la disponibilidad de la sala.

1.3 PROBLEMAS DETECTADOS.

a) Control de asistencia del personal de desarrollo y QA.

El hecho que se realice de forma manual, permite escribir una hora cualquiera sin necesidad de ser correcta, además, al ser de esta manera, posibilita la tardanza al ingreso, la automatización de éste garantizaría la autenticidad de los datos ingresados, aparte de no generar demora al marcar, por otra parte, existe la posibilidad de no saber si una cierta persona se encuentra o no en la oficina, ya que, solo se registra la hora de entrada de la jornada laboral y la hora de salida al termino del día, con un sistema de marcado solo tendría que verificar su última salida.

Un sistema automatizado para ingresos y salidas, aportaría velocidad y confiabilidad para todo el personal.

b) Conteo horas extras

El hecho de que el proceso de ingreso de horas extras sea responsabilidad de cada persona, permite adulterar los datos ingresados, dejando errores o inconsistencias en los registros. En otras palabras, el proceso realizado en el sistema actual, habilita al personal a modificar sus horas.

c) Sala de reunión

Es necesario tener un control respecto a las salas, ya que, al no existir éste, la confusión, el desorden y la desorganización que genera se hace incontenible.

Los jefes de proyecto o alguno quien necesite usar la sala, deben preguntar si ésta se encuentra disponible a la secretaria o a personas quienes estén cerca, en ocasiones interrumpiendo conferencias debido a su falta de conocimiento con respecto a su disponibilidad.

1.4 DESCRIPCIÓN DEL SISTEMA PROPUESTO.

1.4.1 Objetivo Principal

Crear un sistema informático que permita registrar la asistencia del personal, asegurar confiabilidad en datos generados en relación al cálculo de las horas extras, y establecer un óptimo registro de solicitudes para la sala de reunión.

1.4.2 Objetivos específicos

- Registrar reserva de sala de reunión.
- Registrar ingresos y salidas del personal.
- Automatización de cálculo de horas extras.
- Control de usuarios.
- Generación de informes.

1.4.3 Beneficios del sistema.

El sistema propuesto permitirá al personal marcar sus ingresos o salidas por medio de una tarjeta, la cual deberá acercarse al lector para ser recibido por el sistema computacional, esto facilitará el ingreso o salida apoyando el flujo de movimiento en la oficina.

Al tener un sistema de ingresos y salidas facilita el cálculo de horas extras, ya que, es el sistema, con los datos almacenados, el que realiza los cálculos necesarios.

El registro de la sala de reunión será una opción para los jefes de proyecto, permitiendo reservar una hora sin necesidad de conversar con anterioridad con algún otro encargado, simplificando el problema que se generaba y ayudando a la fluidez del trabajo.

1.4.4 Descripción general de la solución propuesta

Se realizará parte de un sistema de administración para la oficina del área de desarrollo y QA de la empresa, el cual pueda registrar cada entrada o salida del personal durante su jornada laboral, éste contará con un lector de tarjetas, el cual, al acercar su tarjeta de identificación registrará su código e ingresará su hora, ya sea de ingreso o salida.

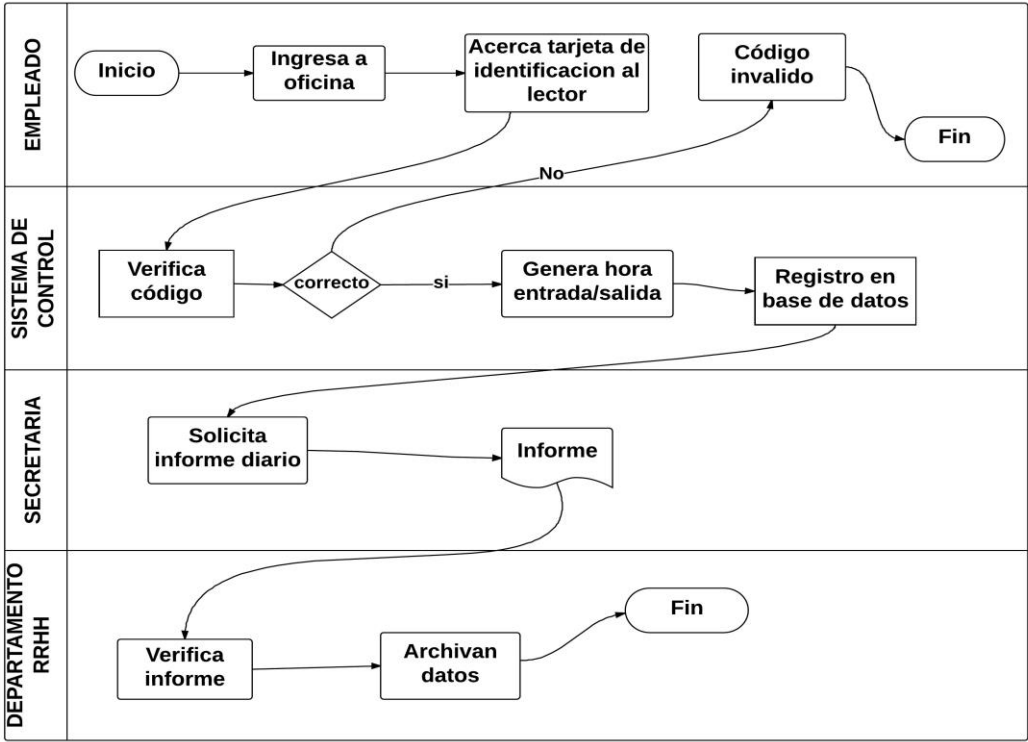
Cada jefe de proyecto y personas de rango alto contarán con una cuenta de usuario para ingresar.

Solo personas especificadas tendrán cuentas de administrador, el que contará, con un mantenedor de personal, salas de reuniones, parámetros, cargos y horarios, este último, tendría la posibilidad de ser un horario general o uno personalizado, que deberá contar con permiso de un superior, también tendrá la opción de justificar inasistencias.

Además, contará con elaboración de informes dando opciones para modificarlo, como el periodo en el cual se basará el informe.

1.4.5 Diagrama de flujo administrativo

CONTROL DE ASISTENCIA



Fuente: Elaboración Propia.

Figura 1-3 Diagrama de flujo administrativo – Control de asistencia.

1.4.6 Estructura funcional del sistema

A continuación, se describirá la estructura funcional del sistema propuesto:

- Mantenedor de Personal: Éste contará con ingreso de campos como, Rut, nombre, fecha nacimiento, apellidos, teléfono, correo electrónico, un campo “activo” que permite saber en qué estado se encuentra actualmente en la empresa, un campo que permita identificar su cargo, y por último, otro campo que identifique en qué oficina se encuentra, además, se le asignará una tarjeta de ingreso con su código.
- Mantenedor de Oficinas: Contendrá el número de la oficina junto con la dirección y fono de esta misma.

- Mantenedor de Cargos: En éste se podrán manejar los diferentes cargos (jefe proyecto, analista, etc.).
- Mantenedor de horario general: Aquel que la empresa define, por ejemplo, ingreso a las nueve de la mañana, una hora de colación y salida a las siete de la tarde.
- Mantenedor de horarios personalizados: Éste debe estar autorizado por personal superior, y el personal tendrá la posibilidad de modificar su horario por día, por ejemplo, día lunes, ingresará a las seis de la mañana y saldrá a las seis de la tarde, martes ingresará a las nueve de la mañana y saldrá a las siete de la tarde, y así, modificarlo para su comodidad, aprobado por su jefe inmediato.
- Registro de Justificación: Las inasistencias se generarán por medio de un procedimiento interno, el cual, leerá la tabla marca_entrada_salida y registrará en la tabla justificación el Rut y la fecha de la persona quien no se registre, este procedimiento se realizará diariamente para saber quiénes no asistieron el día anterior.
- Proceso de Justificación: Al ingresar a la opción de justificar, se desplegará un listado del personal, donde podrá elegir y a continuación escribir un breve comentario.
- Generación de Informes: Éste contará con opciones para la modificación de informes, tales como, el rango de fechas, los últimos 15 días por ejemplo y el tipo de informe, como el informe diario, ambos por medio de opciones ya definidas.
- Función de cálculo de horas extras: Se realizarán cálculos para las horas extras, como por ejemplo, sus horas trabajadas menos las horas exigidas, o sumar el total de horas extras dentro de un rango.
- Mantenedor de usuarios: Para personal de alto rango, secretarias y también cuentas administrador.

- Reserva de Salas: Con esto cada jefe de proyecto, por medio de su cuenta, podrá registrar su(s) hora(s) para el uso de la sala de reunión.
La lista de las reservas se irá actualizando al momento.
- Registro de marcación: Al acercar la tarjeta de identificación de la persona al lector, éste recibirá su código, verificará si es correcto para luego registrar su hora de entrada/salida en la base de datos.

1.4.7 Descripción de fórmulas y cálculos empleados

Para las horas extras:

Esto se empleará para el cálculo diario de sus horas extras en caso de que tuviese, las horas trabajadas (HT) se restarían con las horas laborales exigidas (HLE).

$$HT - HLE$$

En caso de que se trabajase un día domingo o un feriado, estas horas trabajadas (HT) se contarían como horas extra.

$$\sum HT$$

Informes:

Esta fórmula se usaría para calcular según el rango que se especifique las horas, para esto se necesita la fecha de inicio (Fi) y la fecha de término (Ft) del rango.

Horas Extras:

$$\sum_{Fi}^{Ft} HE$$

Horas Trabajadas:

$$\sum_{Fi}^{Ft} HT$$

Para la opción de calcular la media aritmética, se debe ingresar un periodo de fechas, para restringir el cálculo de la sumatoria de las horas, para luego ser dividido por la cantidad de días (CD).

$$\frac{\sum_{Fi}^{Ft} H}{CD}$$

Generales:

Para cálculo de edad, se necesita el año de nacimiento de la persona (AN) y el año actual (AA).

$$AA - AN$$

1.4.8 Información que se manejará

1.4.8.1. Entradas

- Datos del personal: Rut, nombre, teléfono, etc., datos personales del empleado.
- Datos de las solicitudes para la sala: El día, inicio y término del uso de la sala y también una descripción de quien la usará.
- Datos de horarios: El código del día, hora de entrada y salida, el tipo de horario (general o personalizado), si se encuentra activo y la fecha en la que se registró.
- Datos de justificación de asistencias: El Rut y el motivo de la inasistencia.
- Datos de cargos: Nombre del cargo.
- Datos de tarjetas: El código correspondiente a cada una.
- Datos de las oficinas: Dirección, fono, número de oficina.

- Datos de usuarios: Rut, nombre, contraseña.
- Datos de horas extras: La cantidad de horas trabajadas y extras por persona.

1.4.8.2. Salidas

- Listado de personas para justificar inasistencias u horas: Al acceder a la sección de justificar aparecerán las personas quienes se encuentren en la tabla de inasistencia.
- Listado del personal ingresado en la empresa: Habrá una sección donde estarán las personas activas e inactivas de la empresa, ésta tendrá el nombre, apellido, su Rut y su estado.
- Listado horas de ingreso/salida: Al consultar se debe mostrar todas las horas registradas de esa persona en particular.
- Listado de horarios de cada persona: Al dirigirse a consultar horario mostrará las horas de entrada y salida de cada día de la semana.

1.4.8.3. Entidades de información

- Modelo relacional

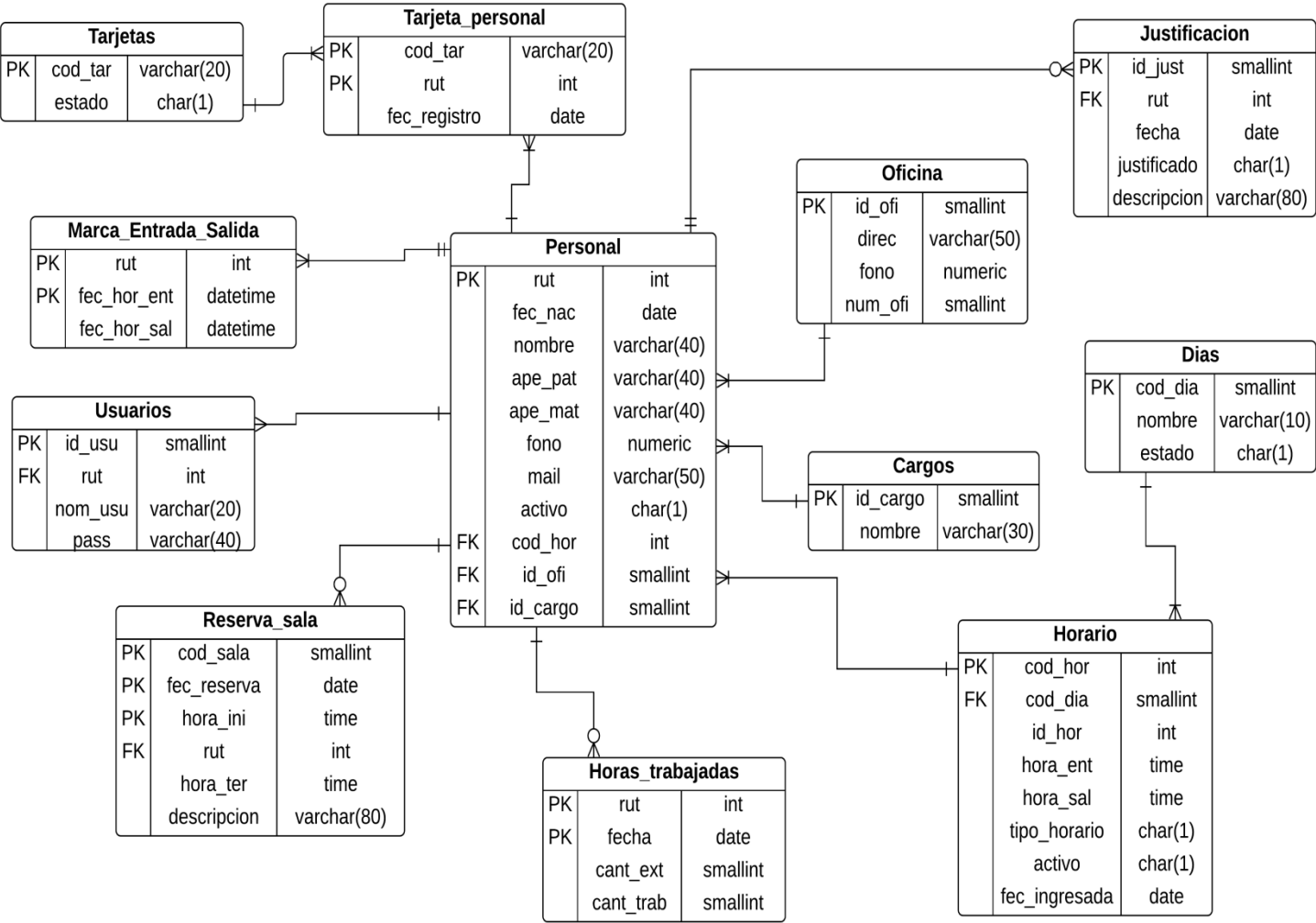


Figura 1-4 Modelo Relacional de Datos.

Fuente: Elaboración Propia.

1.4.9. Estructura de códigos

- Tabla Tarjetas
 - Clave primaria Cod_tar: Este corresponde al código de serie de la tarjeta de tipo varchar, siendo éste el código único que se genera al ser fabricada la tarjeta.

- Tabla Justificación
 - Clave primaria id_just: Número correlativo.

- Tabla Oficina
 - Clave primaria id_ofi: Número correlativo.

- Tabla Cargos
 - Clave primaria id_cargo: Número correlativo.

- Tabla Usuarios
 - Clave primaria id_usu: Número correlativo.

- Tabla Días
 - Clave primaria cod_dia: Número correlativo.

- Tabla Horario
 - Clave primaria cod_hor: Número correlativo.

1.4.10. Condicionantes de diseño

Para el sistema propuesto se usará un microcontrolador Arduino UNO y un lector RFID como lector de tarjetas, éste será de uso diario, por lo que debe ser rápido y de fácil manejo.

El lenguaje con el que se desarrollará será C# con base de datos SQL Server.

En caso de sufrir alguna pérdida de información, se realizarán respaldos de la base de datos una vez al mes, por medio de un disco duro externo.

CAPÍTULO 2: MEDIO AMBIENTE COMPUTACIONAL Y DESCRIPCIÓN DE ARCHIVOS

2. MEDIO AMBIENTE COMPUTACIONAL Y DESCRIPCIÓN DE ARCHIVOS.

2.1 DESCRIPCIÓN DE RECURSO COMPUTACIONAL.

A continuación, se describirán los recursos computacionales que se ocuparán para el desarrollo del sistema, refiriéndose al hardware y software.

a) Configuración del sistema.

- Procesador Intel core i5-3230M CPU @ 2.60GHz.
- Memoria RAM 8 GB
- Microcontrolador Arduino UNO
- Lector RFID

b) Software utilizado.

- Sistema Operativo Windows 10.
- Bases de datos, SQL server.
- HTML.
- C#.
- CSS y Bootstrap.

Para el uso del sistema se requerirá un computador con puerto USB disponible, conexión a internet y tener Microsoft Visual Studio instalado, lo recomendado es que sea 2015 en adelante, ya que, algunas funciones utilizadas puede no se encuentren en versiones anteriores.

Requerimientos mínimos para el uso del Sistema.

- Sistema Operativo Windows 7.
- Memoria RAM 2.5 GB.
- Espacio disponible entre 50 – 130 GB.
- Microcontrolador Arduino y Lector RFID.

2.2 DESCRIPCIÓN DE ARCHIVOS.

En ésta sección se describirán de manera detallada las tablas que serán utilizadas en el sistema, describiendo los campos y su respectivo objetivo dentro de ella.

a) Tabla Personal.

- Nombre Físico:personal
- Nombre Lógico:Personal
- Descripción:Esta tabla contendrá los datos personales de todos los empleados de la empresa
- Clave Primaria:rut
- Clave Foránea:

cod_hor: Referencia a la tabla Horarios.

id_cargo: Referencia a la tabla Cargos.

id_ofi: Referencia a la tabla Oficina.

Tabla 2-1 Descripción de campos tabla Personal.

CAMPOS			
Nombre	Tipo	Longitud	Descripción
Rut	INT	8	Parte numérica del Rut
Nombre	VARCHAR	30	Nombre del empleado
Ape_pat	VARCHAR	30	Apellido paterno del empleado
Ape_mat	VARCHAR	30	Apellido materno del empleado
Fono	NUMERIC	9	Número telefónico del empleado
Fec_nac	DATE	10	Fecha nacimiento del empleado
Mail	VARCHAR	50	Correo electrónico del empleado
Activo	CHAR	1	Estado en que se encuentra el empleado. S: Se encuentra activo. N: Se encuentra inactivo.
cod_hor	INT	3	Código generado que identifica su horario.
Id_ofi	SMALLINT	3	Número que identifica la oficina donde se ubica la persona.
Id_cargo	SMALLINT	3	Número que indica el cargo.

Fuente: Elaboración Propia.

b) Tabla Tarjetas.

Nombre Físico: tarjetas
Nombre Lógico: Tarjetas
Descripción: Esta tabla contendrá los códigos correspondientes a cada tarjeta de ingreso del personal y un estado de ésta.
Clave Primaria: cod_tar.
Clave Foránea: No tiene.

Tabla 2-2 Descripción de campos tabla tarjetas.

CAMPOS			
Nombre	Tipo	Longitud	Descripción
Cod_tar	VARCHAR	20	Código de serie de la tarjeta.
Estado	CHAR	1	Estado en que se encuentra la tarjeta. A: Activo. I: Inactivo.

Fuente: Elaboración Propia.

c) Tabla Marca entrada salida.

Nombre Físico: Marca_entrada_salida
Nombre Lógico: Marca entrada salida
Descripción: Esta tabla contendrá cada ingreso o salida del personal, es decir, se registrarán como un historial.
Clave Primaria: Rut + fec_hor_ent.
Clave Foránea: Rut: Referencia a tabla personal.

Tabla 2-3 Descripción de campos tabla Marca entrada salida.

CAMPOS			
Nombre	Tipo	Longitud	Descripción
rut	INT	8	Parte numérica del rut del empleado
Fec_hor_ent	DATETIME	10	Fecha y Hora de ingreso
Fec_hor_sal	DATETIME	10	Fecha y Hora de salida

Fuente: Elaboración Propia.

d) Tabla oficina.

Nombre Lógico: oficina
Nombre Físico: Oficina
Descripción: Esta tabla contendrá los datos de las oficinas de la empresa.

Clave Primaria: id_ofi.
Clave Foránea: No tiene.

Tabla 2-4 Descripción de campos tabla oficina

CAMPOS			
Nombre	Tipo	Longitud	Descripción
Id_ofi	SMALLINT	3	Número correlativo.
direc	VARCHAR	50	Dirección de la oficina.
fono	INTEGER	9	Teléfono de la oficina.
Num_ofi	SMALLINT	3	Número de la oficina.

Fuente: Elaboración Propia.

e) Tabla Cargos.

Nombre Físico:	cargos
Nombre Lógico:	Cargos
Descripción:	Esta tabla contendrá los nombres de los cargos en la empresa, por ejemplo, desarrollador, etc.
Clave Primaria:	id_cargo.
Clave Foránea:	No tiene.

Tabla 2-5 Descripción de campos tabla cargos.

CAMPOS			
Nombre	Tipo	Longitud	Descripción
Id_cargo	SMALLINT	3	Número correlativo.
Nombre	VARCHAR	30	Nombre del cargo.

Fuente: Elaboración Propia.

f) Tabla Días.

Nombre Físico:	días
Nombre Lógico:	Días
Descripción:	Esta tabla contendrá los nombres de los días de la semana (Lunes, Martes, etc.).
Clave Primaria:	cod_dia.
Clave Foránea:	No tiene.

Tabla 2-6 Descripción de campos tabla Días

CAMPOS			
Nombre	Tipo	Longitud	Descripción
Cod_dia	SMALLINT	3	Número correspondiente a un día de semana.
Nombre	VARCHAR	10	Nombre del día.
Estado	CHAR	1	Estado que especifica el estado del día de la semana. S: Activo. N: Inactivo.

Fuente: Elaboración Propia.

g) Tabla Usuarios.

- Nombre Físico: usuarios
- Nombre Lógico: Usuarios
- Descripción: Esta tabla contendrá los nombres de los usuarios quienes tendrán acceso al sistema.
- Clave Primaria: id_usu.
- Clave Foránea: Rut: Referencia a la tabla Personal.

Tabla 2-7 Descripción de campos tabla Usuarios.

CAMPOS			
Nombre	Tipo	Longitud	Descripción
Id_usu	SMALLINT	3	Número correlativo.
rut	INT	8	Parte numérica delRut.
Nom_usu	VARCHAR	20	Nombre de usuario.
pass	VARCHAR	40	Contraseña del usuario.

Fuente: Elaboración Propia.

h) Tabla Horas trabajadas.

Nombre Físico: Horas_trabajadas.
Nombre Lógico: Horas trabajadas.
Descripción: Esta tabla mantendrá un registro de las horas trabajadas y extras que una persona realizó en un día. .
Clave Primaria: Rut + fecha.
Clave Foránea: Rut: Referencia a la tabla Personal.

Tabla 2-8 Descripción de campos tabla horas trabajadas.

CAMPOS			
Nombre	Tipo	Longitud	Descripción
rut	INT	8	Parte numérica delRut.
fecha	DATE	10	Fecha del registro, mm/dd/aaaa.
Cant_ext	SMALLINT	3	Cantidad de horas extras trabajadas en el día.
Cant_trab	SMALLINT	3	Cantidad de horas normales trabajadas en el día.

Fuente: Elaboración Propia.

i) Tabla Horario.

Nombre Físico: Horario
Nombre Lógico: Horario.
Descripción: Esta tabla tendrá el horario para todos los empleados,diferenciando si es general o personalizado por un campo "tipo_horario" y se manejará la tabla por medio de un código generado.
Clave Primaria: cod_hor
Clave Foránea: cod_dia: Referencia a la tabla Dias.

Tabla 2-9 Descripción de campos tabla Horario.

CAMPOS			
Nombre	Tipo	Longitud	Descripción
Cod_hor	INT	8	Número correlativo.
Id_hor	INT	3	Código numérico generado al momento de ingresarlo.
Cod_dia	SMALLINT	3	Número correspondiente a un día de semana, referencia a tabla días.
Hora_ent	TIME	8	Hora de ingreso de la jornada laboral
Hora_sal	TIME	8	Hora de término de la jornada laboral
Tipo_horario	CHAR	1	Especifica el tipo de horario, todo empleado al ser ingresado tiene horario general hasta que éste es modificado: G: General. P: Personalizado.
activo	CHAR	1	Especifica si este horario se encuentra "activo". S: Activo. N: Inactivo.
Fec_ingresada	DATE	10	Fecha en el que se registró el horario.

Fuente: Elaboración Propia.

j) Tabla Reserva de sala.

Nombre Físico:	Reserva_sala
Nombre Lógico:	Reserva sala
Descripción:	Esta tabla contendrá las reservas de la sala para reuniones.
Clave Primaria:	cod_sala + fec_reserva + hora_ini.
Clave Foránea:	Rut: Referencia a la tabla Personal.

Tabla 2-10 Descripción de campos tabla Reserva de sala.

CAMPOS			
Nombre	Tipo	Longitud	Descripción
Cod_sala	INT	5	Número correlativo de la sala.
fec_reserva	DATE	10	Fecha de la reserva.
Hora_ini	TIME	8	Hora en que comienza la reserva en la sala.
Hora_ter	TIME	8	Hora en que termina la reserva en la sala.
rut	INT	8	Parte numérica delRut de quien hizo la reserva.
descripcion	VARCHAR	80	Breve comentario con el motivo de la reunión.

Fuente: Elaboración Propia.

k) Tabla Tarjeta Personal.

Nombre Físico:	Tarjeta_personal
Nombre Lógico:	Tarjeta Personal
Descripción:	Esta tabla contendrá el código de la tarjeta, elRut de la persona a quien corresponda la tarjeta y la fecha de registro.
Clave Primaria:	Cod_tar + rut.
Clave Foránea:	Rut: Referencia a la tabla personal. Cod_tar: Referencia a la tabla Tarjetas.

Tabla 2-11 Descripción de campos tabla tarjetas personal.

CAMPOS			
Nombre	Tipo	Longitud	Descripción
Cod_tar	VARCHAR	20	Código de serie de la tarjeta.
rut	INT	8	Parte numérica del Rut.
Fec_registro	DATE	10	Fecha en la que se registró la tarjeta.

Fuente: Elaboración Propia.

I) Tabla Justificación.

Nombre Físico: Justificacion

Nombre Lógico: Justificación

Descripción: Esta tabla contendrá los empleados quienes no hayan registrado una marca el día anterior, éstos se almacenan en la tabla por un proceso interno y la justificación de esta ausencia.

Clave Primaria: id_just.

Clave Foránea: Rut: Referencia a la tabla Personal.

Tabla 2-12 Descripción de campos tabla Justificación.

CAMPOS			
Nombre	Tipo	Longitud	Descripción
id_just	SMALLINT	7	Número correlativo.
rut	INT	8	Parte numérica del rut.
fecha	DATE	8	Fecha en la que no se registró marca.
justificado	CHAR	1	Indica si la inasistencia fue justificada. S: justificado. N: No justificado.
descripcion	VARCHAR	80	Breve comentario del motivo de la inasistencia.

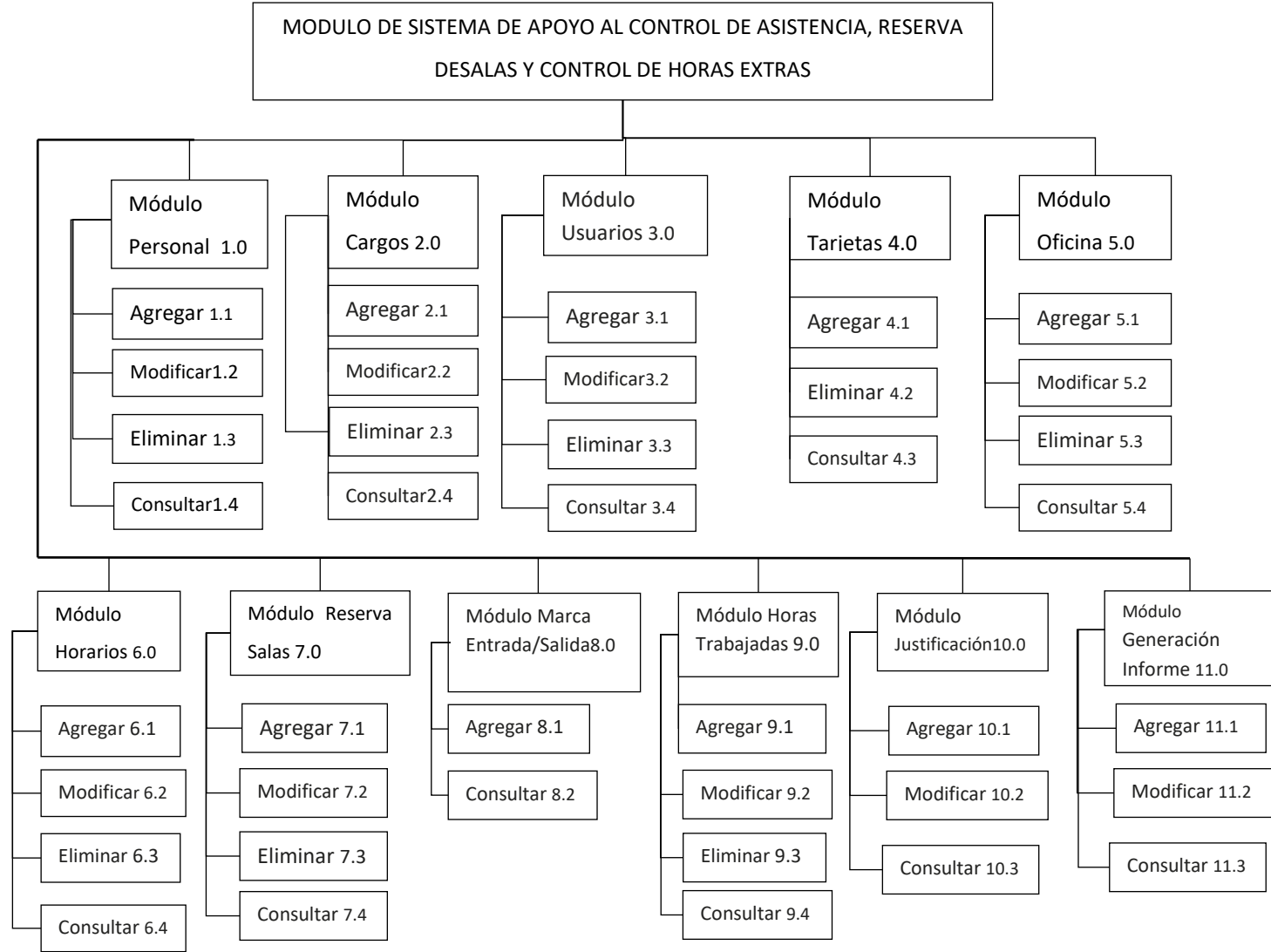
Fuente: Elaboración Propia.

CAPÍTULO 3:DESCRIPCIÓN DE PROGRAMAS

3. DESCRIPCIÓN DE PROGRAMAS.

En este capítulo se presentará la estructura general del sistema, el cual se compondrá por diagrama modular, diagrama de menú y la definición de los programas que conforman el sistema.

3.1 Diagrama Modular

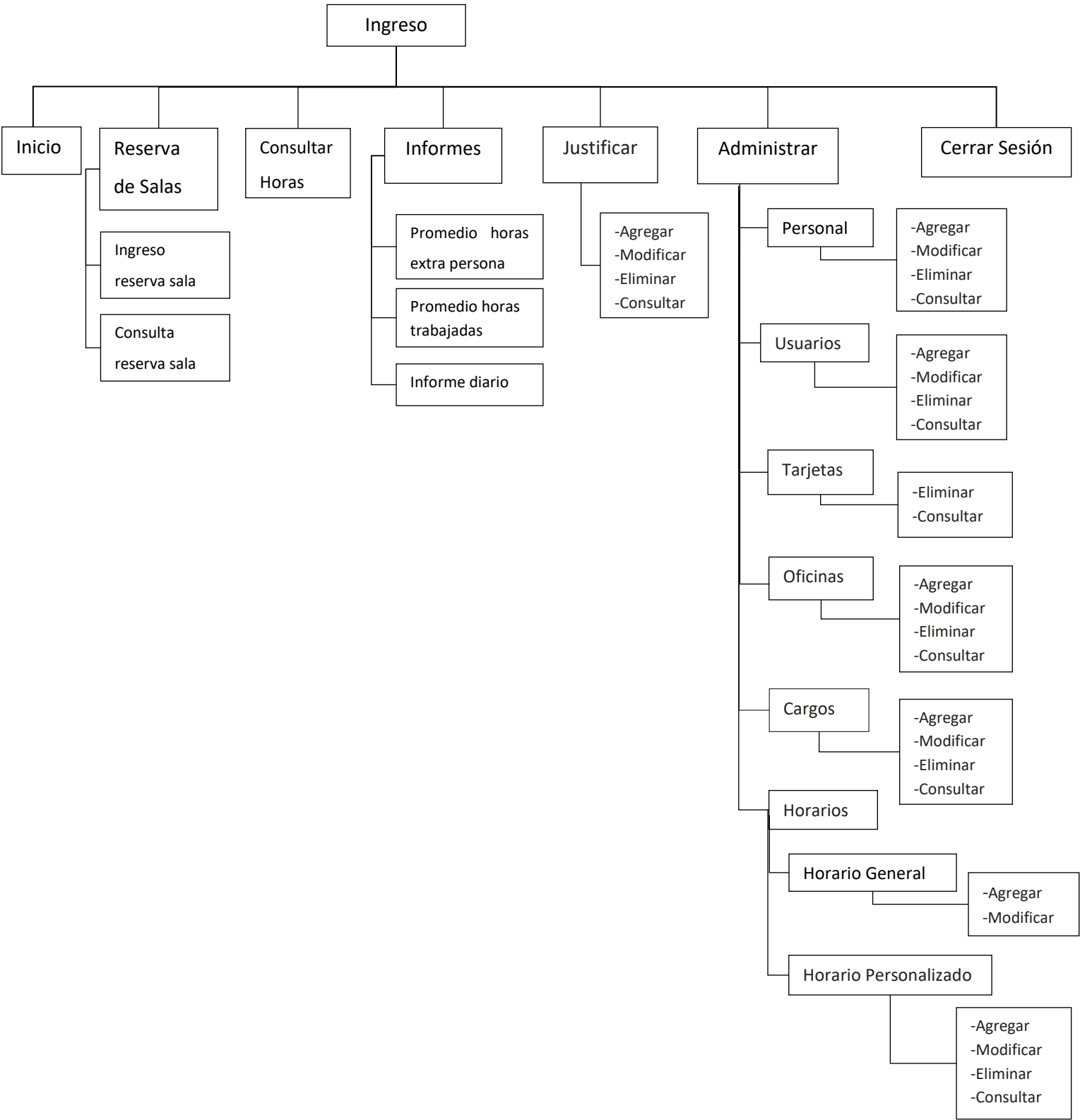


Fuente: Elaboración Propia

Diagrama 3-1. Diagrama Modular.

3.2Diagrama de menús.

3.2.1 Administrador



Fuente: Elaboración Propia

Diagrama 3-2. Diagrama menú para Usuario Administrador.

3.3Programas del Sistema

Tabla 3-1 Descripción de Programas del Sistema.

NOMBRE PROGRAMA	OBJETIVO
Default.aspx*	Proceso de marcación del personal.
Login.aspx*	Permitir el ingreso de Usuarios.
FormularioPersonal.aspx*	Ingresar un nuevo personal, además de agregar y vincular una tarjeta a la persona.
FormPersonalModificar.aspx	Permitir modificar la información personal de una persona.
FormPersonalEliminar.aspx	Marcar a una persona de estado activo a inactivo, para tener un historial.
FormPersonalConsultar.aspx*	Permitir listar la información de todo el personal o realizar una búsqueda específica.
FormularioUsuarios.aspx	Ingresar un nuevo Usuario para una persona específica.
FormUsuariosModificar.aspx	Modificar información de un Usuario.
FormUsuariosEliminar.aspx	Eliminar un usuario.
FormUsuariosConsultar.aspx	Listar los usuarios existentes.
FormularioHorario.aspx	Ingresar horario general de la empresa.
FormularioHorarioGenModificar.aspx	Modificar el horario general.
FormularioHorarioGenEliminar.aspx	Eliminar el horario general.
FormularioHorarioGenConsultar.aspx	Listar el horario general.
FormHorarioPerso.aspx	Ingresar un horario personalizado para una persona específica.
FormHorarioPersoModificar.aspx	Modificar un horario personalizado de una persona.
FormHorarioPersoEliminar.aspx	Eliminar un horario personalizado de una persona.
FormHorarioPersoConsultar.aspx	Listar los horarios.
FormularioCargos.aspx	Ingresar cargos del personal de la empresa.
FormularioCargosModificar.aspx	Modificar los cargos de la empresa.
FormularioCargosEliminar.aspx	Eliminar cargos.
FormularioCargosConsultar.aspx	Listar los cargos de la empresa.
FormularioReservaSala.aspx*	Permitir reservar una sala.
FormReservaSalaEliminar.aspx	Consultar, modificar y eliminar una reserva de sala.
FormularioJustificar.aspx*	Permitir justificar una inasistencia de una

	persona en un día específico.
FormularioRegistroHoras.aspx	Consultar según el Rut sus horas de ingreso/salida.
FormTarjetasConsultar.aspx*	Se listan todas las tarjetas y a quien pertenece.
FormTarjetasEliminar.aspx	Al ingresar el código permite desactivar la tarjeta.
FormularioOficinas.aspx	Ingresar otras Oficinas de la empresa.
FormOficinasConsultar.aspx	Permite consultar las oficinas ingresadas.
FormOficinasEliminar.aspx	Eliminar oficinas.
FormOficinasModificar.aspx	Modificar oficinas.
MasterPage.master	Permitir establecer un diseño estándar para todas las pantallas.

Fuente: Elaboración Propia.

*: Se describirán los programas seleccionados.

3.4 Descripción de Programas

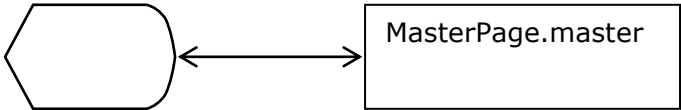
A continuación se describirán los programas anteriormente seleccionados.

3.4.1 Menú Principal.(Codigo fuente Pag. 53 en Anexos)

Nombre: MasterPage.master

Objetivo: Establecer un diseño estándar para todas las pantallas.

Diagrama de bloques



Fuente: Elaboración propia.

Diagrama 3-3. Diagrama Menú Principal.



Figura 3-1 Menú Principal.

3.4.2 Inicio Sesión.(Codigo fuente Pag. 57 en Anexos)

Nombre: login.aspx

Objetivo: Controlar el acceso al sistema por parte de un usuario.

Diagrama de bloques



Fuente: Elaboración propia.

Diagrama 3-4. Diagrama de inicio sesión.

Reglas del Proceso

Se muestra una pantalla para el ingreso de sus datos (Figura 3-2), éstos son validados que existan en tabla USUARIOS.

SI (Valores existen) ENTONCES

Hace ingreso y muestra el menú principal (Figura 3-1).

SINO Mostrar mensaje “Usuario o Contraseña Incorrectas.” (Figura 3-3).



Figura 3-2 Pantalla Ingreso.

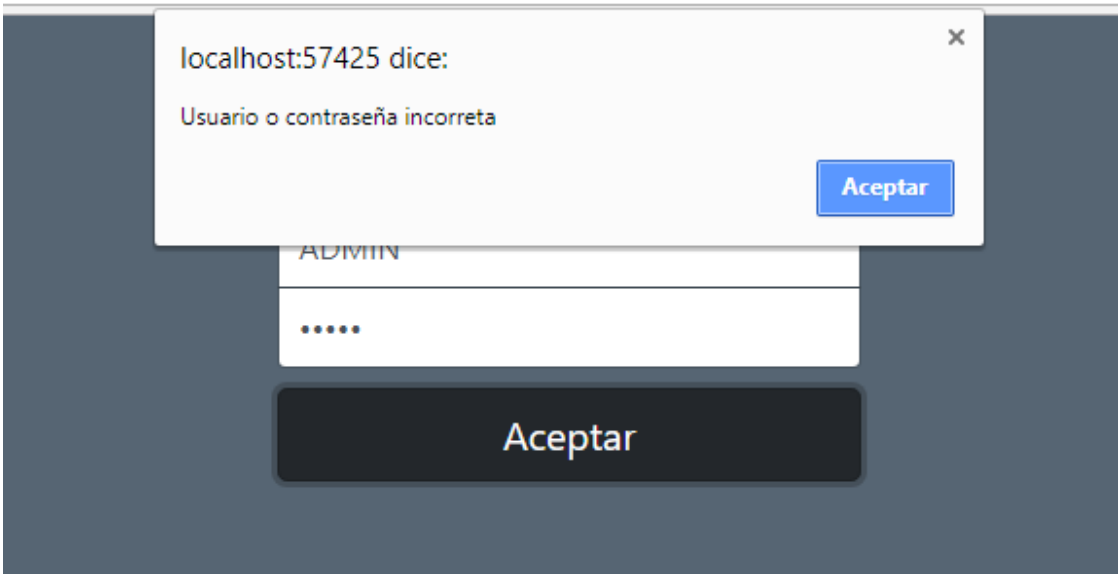


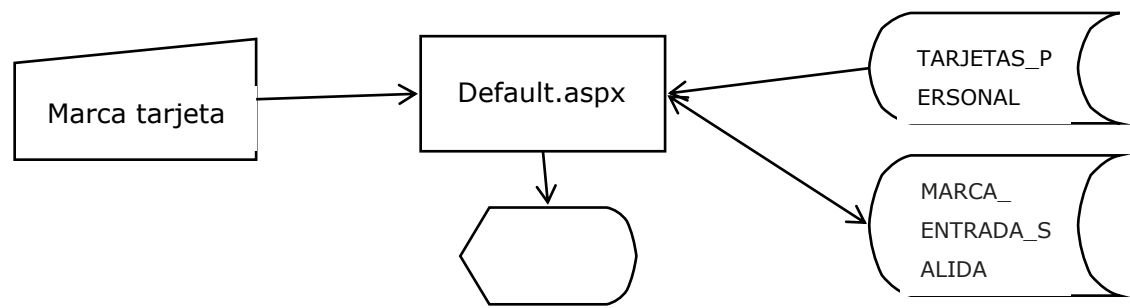
Figura 3-3 Pantalla Ingreso con aviso de error.

3.4.3 Marca Ingreso/Salida.(Codigo fuente Pag. 59 en Anexos)

Nombre: Default.aspx

Objetivo: Obtener y registrar la marca de ingreso y salida del personal.

Diagrama de bloques



Fuente: Elaboración propia.

Diagrama 3-5. Diagrama demarca entrada/salida.

Reglas del proceso

Se muestra la pantalla de inicio del menú(Figura 3.1), con ayuda del lector, se obtiene el código de la tarjeta, éste es validado que existay se encuentre en usoen tabla TARJETAS_PERSONAL.

SI (Código Existe) ENTONCES

Se ingresa la marca en tabla MARCA_ENTRADA_SALIDA, con el Rut vinculado a la tarjeta y la fecha con hora (Figura 3-4).

SINO Mostrar mensaje "Tarjeta Invalida."



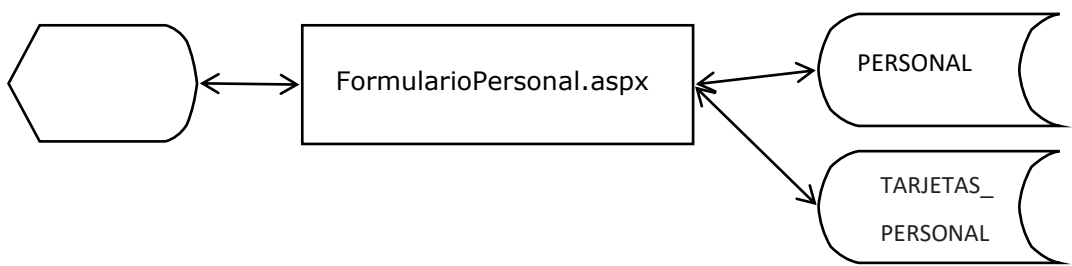
Figura 3-4 Pantalla Inicio Menú Principal

3.4.4Ingreso Personal.(Codigo fuente Pag. 62 en Anexos)

Nombre: FormularioPersonal.aspx

Objetivo: Ingresar un nuevo personal.

Diagrama de bloques



Fuente: Elaboración propia.

Diagrama 3-6. Diagrama de ingreso personal.

Reglas del proceso

En la pantalla de menú (Figura 3-1), al presionar la opción “Administrar” se despliega un submenú donde selecciona Personal>Agregar; Se dirige a la pantalla con el formulario para ingresar el personal (Figura 3-6).

Al ingresar el Rut está la opción de preguntar si anteriormente existió el personal en la empresapresionando la lupa y así recuperar sus datos (Figura 3-5), se valida si efectivamente mantuvo actividad en la empresa en tabla PERSONAL.

SI (Rut inactivo) ENTONCES

Se llenan los campos

SINO Mostrar mensaje “El Rut ya se encuentra activo.” O “Rut no registra datos en la base”, en caso de que estuviese activo o si nunca mantuvo actividad.

Los campos se validan que no se encuentren vacíos, que la tarjeta no se encuentre en uso y que el Rut sea válido y no esté activo.

SI (Valores Cumplen) ENTONCES

Se ingresa la persona en tabla PERSONAL, se actualiza el estado de la tarjeta en caso de estar inactiva o se agrega si no existía, se agrega en tabla

TARJETAS_PERSONAL el Rut de la persona, el código de la tarjeta y la fecha y se le agrega un horario.

SINO Mostrar mensaje “Debe completar los campos” o “El Rut ingresado ya existe” o “El Código de la tarjeta ya existe” o “Debe acercar una tarjeta al lector” o “La operación no pudo finalizar correctamente” según corresponda.

InicioReserva de SalasConsultar HorasInformesJustificarAdministrarCerrar Sesión

Agregar Personal

Rut12801001-7

Fono

NombreEscriba su nombre

Apellido PaternoApellido Paterno

Apellido MaternoApellido materno

Fecha Nacimientomm-dd-aaaa

Emailcorreo@email.cl

OficinaAV. LIBERTADOR BERNARDO OHIG

CargoSECRETARIA

Tarjeta no registrada

Registrar

El rut se encuentra en estado inactivo. ¿Desea recuperar sus datos?.

AceptarCancelar

Aceptar

Figura 3-5 Pantalla agregar personal-Recuperar datos.

InicioReserva de SalasConsultar HorasInformesJustificarAdministrarCerrar Sesión

Agregar Personal

Rut18554222-K

Fono964898075

NombreCONSTANZA

Apellido PaternoVALDERRAMA

Apellido MaternoENCELEGRE

Fecha Nacimiento06-20-1975

EmailCONSTANZA.VALDERRAMA.E@GMAIL.COM

OficinaAV. LIBERTADOR BERNARDO OHIG

CargoADMINISTRADOR

Tarjeta0333106121

Registrar

Aceptar

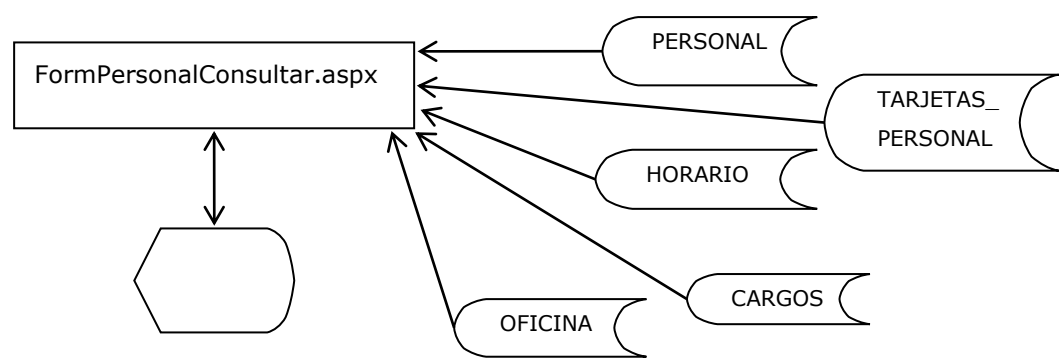
Figura 3-6 Pantalla agregar personal.

3.4.5 Consultar Personal. (Codigo fuente Pag. 72 en Anexos)

Nombre: FormPersonalConsultar.aspx

Objetivo: Consultar los datos del personal.

Diagrama de bloques



Fuente: Elaboración propia.

Diagrama 3-7. Diagrama consultar personal.

Reglas del proceso

En la pantalla de menú (Figura 3-1), al presionar la opción “Administrar” se despliega un submenú donde selecciona Personal>Consultar; se dirige a la pantalla con el listado del personal(Figura 3-7).

En los buscadores se valida que el dato a buscarexista y solo sea alfabético o numérico según corresponda.

Consultar Personal

Buscador

Rut (sin dígito verificador)

Nombre

Consultar

Rut	Fecha Nacimiento	Nombre	Apellido paterno	Apellido materno	Telefono	Mail	Estado	Tipo de horario	Tarjeta	Oficina	Cargo
17656502	13/03/1991	JORGE	VALDERRAMA	ENCELEGRE	898989898	JORGE.VALDERRAMA@GMAIL.COM	Activo	Personalizado	130154100 02	AV. LIBERTADOR BERNARDO OHIGGINS	DESARROLLADOR
19489600	01/12/1998	PRUEBA	PRUEBA	PRUEBA	45545121	DSDS@JDJD.COM	Activo	Personalizado	19411699 02	AV. LIBERTADOR BERNARDO OHIGGINS	DESARROLLADOR
6751416	16/12/1998	FACUNDO	CABRALES	TAPIA	99999999	HAHAHA@JDJD.COM	Activo	Personalizado	206249221232	AV. LIBERTADOR BERNARDO OHIGGINS	DESARROLLADOR
18554221	23/03/1994	VIVIANA	VALDERRAMA	ENCELEGRE	656565656	VIVL.VALDERRAMA@GMAIL.COM	Activo	Personalizado	221170250228	AV. LIBERTADOR BERNARDO OHIGGINS	SECRETARIA
18554222	23/03/1994	CONSTANZA	VALDERRAMA	ENCELEGRE	64898075	CONI@GMAIL.COM	Activo	Personalizado	901786641	AV. LIBERTADOR BERNARDO OHIGGINS	ADMINISTRADOR

Figura 3-7 Pantalla consultar personal.

3.4.6 Ingreso Reserva de Sala. (Codigo fuente Pag. 74 en Anexos)

Nombre: FormularioReservaSala.aspx

Objetivo: Ingresar una solicitud de reserva para la sala de reunión.

Diagrama de bloques



Fuente: Elaboración propia.

Diagrama 3-8. Diagrama ingresar reserva de sala.

Reglas del proceso

En la pantalla de menú(Figura 3-1), al presionar la opción Reserva de Salas > Ingresar Reserva, se dirige al formulario para ingresar la reserva (Figura 3-8).

Para realizar la reserva de sala se valida que la hora término sea mayor que la hora de inicio,así también que la fecha sea mayor a la actual y que en el horario seleccionado no exista ya una reserva.

SI (Valores Correctos) ENTONCES

Se ingresa una solicitud de reserva en tabla RESERVA_SALA.

SINO Mostrar mensaje “La operación no se pudo realizar, la hora puede estar ya registrada” (Figura 3-9).

InicioReserva de SalasConsultar HorasInformesJustificarAdministrarCerrar Sesión

Reserva de Salas

Hora inicio

16:00

Hora termino

16:40

Nombre

CONSTANZA VALDERRAMA

Fecha

01-15-2018

Descripcion Breve

Analizar situación actual

Aceptar

Figura 3-8 Pantalla Reserva Salas

InicioReserva de SalasConsultar HorasInformesJustificar

localhost:57425 dice:

La operacion no se pudo realizar, la hora puede estar ya registrada.

Aceptar

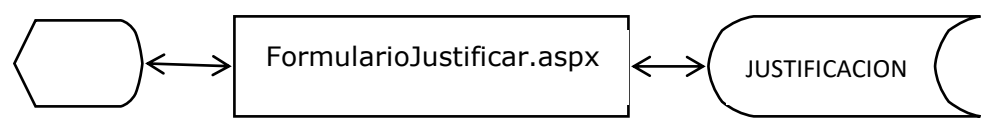
Figura 3-9 Pantalla Reserva Salas

3.4.7 Ingreso Justificación. (Codigo fuente Pag. 78 en Anexos)

Nombre: FormularioJustificar.aspx

Objetivo: Justificar una inasistencia de una persona.

Diagrama de bloques



Fuente: Elaboración propia.

Diagrama 3-9. Diagrama ingreso justificación.

Reglas del proceso

En el menú principal seleccionar “Justificar”, éste abrirá el formulario con una lista para justificar inasistencias (Figura 3-10).

En los buscadores se valida que los datos ingresados sean correctos. Se listan todas las inasistencias, ya sea justificadas o no (Figura 3.10), para justificar deben presionar el símbolo de editar y se abrirán los campos (Figura 3-11).

Se toma el Rut de la selección para actualizar el registro en la tabla JUSTIFICACION vinculado a ese Rut y fecha

Justificar Inasistencia

Rut

11111111-k

Fecha

mm-dd-aaaa

Aceptar

Rut	Fecha	Estado	Descripcion	
18554222	11-29-2017 0:00:00	SI	FUE A MEDICO	
18210645	12-03-2017 0:00:00	SI	ESTABLECER META	
18554222	12-03-2017 0:00:00	SI	EMERGENCIA MEDICA	
17656502	12-03-2017 0:00:00	SI	ORGANIZAR SALIDA GRUPAL	
17656502	12-05-2017 0:00:00	SI	ANALIZAR AVANCE	

Figura 3-10 Pantalla justificar inasistencia.

Justificar Inasistencia

Rut

11111111-k

Fecha

01-02-2018

Aceptar

Rut	Fecha	Estado	Descripción	
18554222	01-02-2018 0:00:00	SI	2 de enero	
18210645	01-02-2018 0:00:00	NO	JUSTIFICAR	
18554221	01-02-2018 0:00:00	NO		

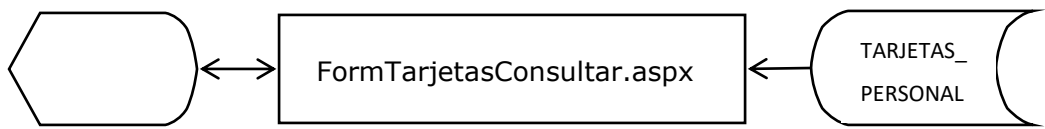
Figura 3-11 Pantalla justificar inasistencia.

3.4.8 Consultar Tarjetas. (Codigo fuente Pag. 82 en Anexos)

Nombre: FormTarjetasConsultar.aspx

Objetivo: Consultar que tarjetas están en uso y a quien pertenece.

Diagrama de bloques



Fuente: Elaboración propia.

Diagrama 3-10. Diagrama consultar tarjetas.

Reglas del proceso

En la pantalla de menú(Figura 3-1), al presionar la opción “Administrar” se despliega un submenú donde selecciona Tarjetas>Consultar; Se dirige a la pantalla donde se listan las tarjetas con sus respectivos propietarios (Figura 3-12).

Inicio Reserva de Salas ▾ Consultar Horas Informes Justificar Administrar ▾ Cerrar Sesión

Consultar Tarjetas

Nombre	Codigo Tarjeta	Estado
FACUNDO CABRALES	206249221232	Activo
JORGE VALDERRAMA	130154100 02	Activo
LUIS1 PAREDES	9923559217	Activo
VIVIANA VALDERRAMA	221170250228	Activo
CONSTANZA VALDERRAMA	901786641	Activo
PRUEBA PRUEBA	19411699 02	Activo

Figura 3-12 Pantalla Consultar Tarjetas.

CONCLUSIÓN

El propósito de haber desarrollado el proyecto ya expuesto, fue apoyar el trabajo administrativo realizado por la secretaria en la oficina ya mencionada de la empresa DIMENSION S.A, y mejorar la fluidez de dicha oficina.

Para lograr esto, se realizó un análisis de la situación actual, el cual arrojó la necesidad de agilizar el proceso de ingreso, y establecer una forma de organizar solicitudes para la reserva de la sala en que se desarrollan reuniones como también normalizar el proceso de registro de sus horas extras.

Durante el desarrollo del sistema, una de las dificultades que se presentó fue la nula experiencia con el microcontrolador arduino UNO, el cual se uso como lector para las tarjetas de ingreso del personal. Por ésto, se debió aprender el funcionamiento y manejo de éste, por otra parte, la poca experiencia en el lenguaje usado contribuyó a la demora en el desarrollo del sistema y a la simpleza de éste.

El proyecto realiza los procesos de registro de ingreso y salida, justifica inasistencias, reserva la sala para reuniones, administra al personal y genera informes básicos, siendo un sistema preliminar capaz de mejorar y robustecerse, además de ser un sistema fácilmente adaptable, pudiendo ser expandido al resto de las oficinas vinculándolos por medio de un servidor.

El sistema realizado se basó en una de las oficinas de la empresa DIMENSION S.A ubicada en la ciudad de Santiago, quienes dieron autorización para utilizar su nombre y situación, pero no está vinculada directamente con el proyecto, debido a esto el sistema antes expuesto, no fue implementado en dicha oficina.

BIBLIOGRAFÍA

- Andres Galante, Bardi Harborow, Connor Sears, Chris Rebert, fat, Gleb Mazovetskiy, Heinrich Fenkart, Justin Dorfman, Johann-S, Julian Thilo, Mark Otto, Patrick H. Lauke, Thomas McDonald, Zac Echola, BOOTSTRAP [en línea], Viernes, Agosto 19, 2011, <<https://getbootstrap.com>> [Última consulta: 08 enero 2018]
- Refsnes Data, W3Schools [en línea], 1998, <www.w3schools.com> [Última consulta: 08 enero 2018]
- MCI Electronics, ARDUINO [en línea], <<http://arduino.cl/>> [Última consulta: 08 enero 2018]
- Microsoft, Microsoft[en línea], Last updated: October 2013 <<https://social.msdn.microsoft.com>> [Última consulta: 08 enero 2018]
- Orlando, HETPRO [en línea], abril 25, 2014, <<https://hetpro-store.com/TUTORIALES/modulo-lector-rfid-rc522-rf-con-arduino/>> [Última consulta: 08 enero 2018]
- Michael Kellogg, WordReference [en línea], 1999, <<http://www.wordreference.com>> [Última consulta: 14 enero 2018]

ANEXOS – PROGRAMAS FUENTE

▪ **Menú principal**

MasterPage.master

```
<head runat="server">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<!-- Ajax -->
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js">
</script>

<!-- Required meta tags -->
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<title>Aplicacion</title>

<!-- Bootstrap CSS -->
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta/css/bootstrap.min.css" integrity="sha384-Y6pD6FV/Vv2HJnA6t+vsIU6fwYXjCFtcEpHbNJ0lyAFsXTsjBbfaDjzALeQsN6M" crossorigin="anonymous">

<link href="style/MenuDesplegable.css" rel="stylesheet" />

<script type="text/javascript">

function alerta(str) {
alert(str);
}
</script>

<asp:ContentPlaceHolder ID="head" runat="server">
</asp:ContentPlaceHolder>

</head>
<body>

<!-- Barra menu -->

<nav class="navbar navbar-expand-md navbar-dark bg-dark" id="nav">

<a class="navbar-brand" href="Default.aspx">Inicio </a>

<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNavDropdown" aria-controls="navbarNavDropdown" aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>
</button>

<div class="collapse navbar-collapse" id="navbarNavDropdown">
<ul class="navbar-nav multi-level">

<li class="nav-item dropdown">
```

```

<a class="nav-link dropdown-toggle text-white" href="#" id="DropdownMenuR"
data-toggle="dropdown" aria-haspopup="true"aria-expanded="false">
Reserva de Salas
</a>
<ul class="dropdown-menu" aria-labelledby="DropdownMenuR">
<li><a class="text-dark ml-2 my-2" href="FormularioReservaSala.aspx">Ingresar
Reserva</a></li>
<li><a class="text-dark ml-2 my-2"
href="FormResevaSalaEliminar.aspx">Consultar Reserva</a></li>
</ul>
</li>

<li class="nav-item"><a class="nav-link text-white"
href="FormularioRegistroHoras.aspx">Consultar Horas </a>
</li>
<li class="nav-item"><a class="nav-link text-white"
href="FormularioInforme.aspx">Informes</a>
</li>
<li class="nav-item"><a class="nav-link text-white"
href="FormularioJustificar.aspx">Justificar</a>
</li>

<li class="nav-item dropdown">
<a class="nav-link dropdown-toggle text-white" href="#"
id="navbarDropdownMenuLink" data-toggle="dropdown" aria-haspopup="true"
aria-expanded="false">
Administrar
</a>

<ul class="dropdown-menu" aria-labelledby="navbarDropdownMenuLink">

<li class="dropdown-item dropdown-submenu ">
<a class="dropdown-toggle text-dark" data-toggle="dropdown"
href="#">Personal</a>

<ul class="dropdown-menu">
<li><a class="text-dark ml-1 mr-2"
href="FormularioPersonal.aspx">Agregar</a></li>

<li><a class="text-dark ml-1 mr-2"
href="FormPersonalModificar.aspx">Modificar</a></li>

<li><a class="text-dark ml-1 mr-2"
href="FormPersonalEliminar.aspx">Eliminar</a></li>

<li><a class="text-dark ml-1 mr-2"
href="FormPersonalConsultar.aspx">Consultar</a></li>
</ul>
</li>

<li class="dropdown-item dropdown-submenu ">
<a class="dropdown-toggle text-dark" data-toggle="dropdown"
href="#">Usuarios</a>

<ul class="dropdown-menu">
<li><a class="text-dark ml-1 mr-2"
href="FormularioUsuarios.aspx">Agregar</a></li>

<li><a class="text-dark ml-1 mr-2"
href="FormUsuariosModificar.aspx">Modificar</a></li>

<li><a class="text-dark ml-1 mr-2"
href="FormUsuariosEliminar.aspx">Eliminar</a></li>

```

```

<li><a class="text-dark ml-1 mr-2"
href="FormUsuariosConsultar.aspx">Consultar</a></li>
</ul>
</li>

```

```

<li class="dropdown-item dropdown-submenu ">
<a class="dropdown-toggle text-dark" data-toggle="dropdown"
href="#">Tarjetas</a>

```

```

<ul class="dropdown-menu">
<li><a class="text-dark ml-1 mr-2"
href="FormTarjetasEliminar.aspx">Eliminar</a></li>
<li><a class="text-dark ml-1 mr-2"
href="FormTarjetasConsultar.aspx">Consultar</a></li>
</ul>
</li>

```

```

<li class="dropdown-item dropdown-submenu ">
<a class="dropdown-toggle text-dark" data-toggle="dropdown"
href="#">Cargos</a>

```

```

<ul class="dropdown-menu">
<li><a class="text-dark ml-1 mr-2"
href="FormularioCargos.aspx">Agregar</a></li>

```

```

<li><a class="text-dark ml-1 mr-2"
href="FormCargosModificar.aspx">Modificar</a></li>

```

```

<li><a class="text-dark ml-1 mr-2"
href="FormCargosEliminar.aspx">Eliminar</a></li>

```

```

<li><a class="text-dark ml-1 mr-2"
href="FormCargosConsultar.aspx">Consultar</a></li>
</ul>
</li>

```

```

<li class="dropdown-item dropdown-submenu ">
<a class="dropdown-toggle text-dark" data-toggle="dropdown"
href="#">Oficinas</a>

```

```

<ul class="dropdown-menu">
<li><a class="text-dark ml-1 mr-2"
href="FormularioOficinas.aspx">Agregar</a></li>

```

```

<li><a class="text-dark ml-1 mr-2"
href="FormOficinasModificar.aspx">Modificar</a></li>

```

```

<li><a class="text-dark ml-1 mr-2"
href="FormOficinasEliminar.aspx">Eliminar</a></li>

```

```

<li><a class="text-dark ml-1 mr-2"
href="FormOficinasConsultar.aspx">Consultar</a></li>
</ul>
</li>

```

```

<li class="dropdown-item dropdown-submenu ">
<a href="#" class="dropdown-toggle text-dark" data-
toggle="dropdown">Horarios</a>

```

```

<ul class="dropdown-menu">

```

```

<li class="dropdown dropdown-submenu">
<a class="text-dark ml-1 mr-2" href="#">Horario General</a>

```

```

<ul class="dropdown-menu">
<li><a class="text-dark ml-1 mr-2"
href="FormularioHorario.aspx">Agregar</a></li>

<li><a class="text-dark ml-1 mr-2"
href="FormHorarioModificar.aspx">Modificar</a></li>

<li><a class="text-dark ml-1 mr-2" href="FormHorarioConsultar.aspx"
>Consultar</a></li>
</ul>
</li>

<li class="dropdown dropdown-submenu">
<a class="text-dark ml-1 mr-2" href="#">Horario Personalizado</a>

<ul class="dropdown-menu">
<li><a class="text-dark ml-1 mr-2"
href="FormHorarioPerso.aspx">Agregar</a></li>

<li><a class="text-dark ml-1 mr-2"
href="FormHorarioPersoModificar.aspx">Modificar</a></li>

<li><a class="text-dark ml-1 mr-2"
href="FormHorarioPersoEliminar.aspx">Eliminar</a></li>

<li><a class="text-dark ml-1 mr-2"
href="FormHorarioPersoConsultar.aspx">Consultar</a></li>
</ul>
</li>
</ul>
</li>
</ul>
</li>
</ul>
</li>

<li class="nav-item ">
    <a class="nav-link" href="login.aspx">Cerrar Sesión</a>
</li>
</ul>
</div>
</nav>

<!-- Fin Barra menu -->

<div>
<asp:ContentPlaceHolder ID="ContentPlaceHolder1" runat="server">
</asp:ContentPlaceHolder>
</div>

<!-- Optional JavaScript -->
<!-- jQuery first, then Popper.js, then Bootstrap JS -->
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-
KJ3o2DKtIkVYIK3UENzmM7KCKRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN"
crossorigin="anonymous"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.11.0/umd/popper.min.js"
integrity="sha384-
b/U6ypiBEHpOf/4+1nzFpr53nxSS+GLCKfwbDfNtXtclqgenISfwAzpKaMNFNmj4"
crossorigin="anonymous"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-
beta/js/bootstrap.min.js" integrity="sha384-
h0AbiXch4ZDo7tp9hKZ4TsHbi047NrKGLO3SEJAg45jXxnGIfYzk4Si90RDIqNm1"
crossorigin="anonymous"></script>

```

```
<script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js">
</script>

</body>
```

▪ **Inicio Sesión**

Login.aspx.cs

```
using System;
using System.Data;
using System.Data.SqlClient;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class login : System.Web.UI.Page
{
    Persona per = new Persona();
    protected void Page_Load(object sender, EventArgs e)
    { }

    protected void Button1_Click(object sender, EventArgs e)
    {
        try
        {
            DataSet ds = per.ValidaNomUsuario(usuario.Value.Trim().ToUpper(),
            pass.Value.Trim().ToUpper());

            if (ds.Tables[0].Rows.Count != 0)
            {
                Session["NomUsuario"] = usuario.Value.ToString();
                //Response.Redirect("Default.aspx", false);

                string str = "window.open('Default.aspx','')";
                EjecutaScript(this, str);
            }
            else
            {
                //Mensaje
                string script = @"<script type='text/javascript'>
                alert('Usuario o contraseña incorrecta');
                </script>";
                ScriptManager.RegisterStartupScript(this, typeof(Page), "alerta", script, false);
            }
        }
        catch (Exception a) { }
    }

    public void EjecutaScript(Page Pg, String nombrefuncion)
    {
        String Script;
        Script = "<script language=JavaScript>" + nombrefuncion + "</script>";
        Pg.RegisterStartupScript("Pgn", Script);
    }
}
```

Login.aspx

```

<head runat="server">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>
<title></title>

<!--Ajax -->
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>

<!-- Bootstrap CSS -->
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-
beta/css/bootstrap.min.css"
integrity="sha384-
/Y6pD6FV/Vv2HJnA6t+vsIU6fwYXjCFtcEpHbNJ0lyAFsXTsjBbfaDjzALeQsN6M"
crossorigin="anonymous"/>

<link href="style/InicioSesion.css" rel="stylesheet" />

<script type="text/javascript">

function alerta(str) {
alert(str);
}
</script>

</head>
<body>
<div class="container ">
<!-- -->
<form class="form-signin" method="post" runat="server" >

<h2 class="form-signin-heading" style="text-align:center;">Ingreso</h2>

<label for="usuario" class="sr-only">Usuario</label>
<input type="text" id="usuario" class="form-control" placeholder="Usuario"
required="" autofocus="" runat="server" />

<label for="pass" class="sr-only">Contraseña</label>
<input type="password" id="pass" class="form-control" placeholder="Contraseña"
required="" runat="server"/>

<asp:Button ID="Button1" runat="server" Text="Aceptar" class="btn btn-lg btn-
dark btn-block" OnClick="Button1_Click" />

</form>
<!-- -->
</div>

<!-- jQuery first, then Popper.js, then Bootstrap JS -->
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-
KJ3o2DKtIkVYIK3UENzmM7KCKRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN"
crossorigin="anonymous"></script>

```

```

<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.11.0/umd/popper.min.js"
integrity="sha384-
b/U6ypiBEHpOf/4+1nzFpr53nxSS+GLCKfwbDfNtXtclqgenISfwAzpKaMNFNmj4"
crossorigin="anonymous"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-
beta/js/bootstrap.min.js" integrity="sha384-
h0AbiXch4ZDo7tp9hKZ4TsHbi047NrKGL03SEJAg45jXxnGIfYzk4Si90RDIqNm1"
crossorigin="anonymous"></script>

<script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js">
</script>

</body>

```

▪ **Marca Entrada/Salida**

Default.aspx

```

<%@ Page Title="" Language="C#" MasterPageFile="~/MasterPage.master"
AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="_Default" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">

<style>
body {
padding-top: 40px;
padding-bottom: 40px;
background-color: #566573;
}

.form-signin {
max-width: 330px;
padding: 15px;
margin: 0 auto;
}

.form-signin .form-control {
position: relative;
height: auto;
-webkit-box-sizing: border-box;
box-sizing: border-box;
padding: 10px;
font-size: 16px;
}
</style>

<!-- -->
<form runat="server" class="form-signin">
<asp:ScriptManager ID="ScriptManager1" runat="server" />
<div class="">
<input type="text" id="txt" runat="server" visible="false"/>

<div class="jumbotron">
<asp:Label ID="Label1" runat="server" CssClass="text-muted h4">

```

```

        </asp:Label>
    </div>

    <asp:Button ID="Button1" runat="server" Text="Agregar" Visible="false"
        CssClass="btn btn-dark btn-lg"/>

    <asp:Timer ID="Timer1" runat="server" Interval="5000"
    OnTick="Timer1_Tick"></asp:Timer>
    <%--<asp:Timer ID="Timer1" runat="server" Interval="1500"
    OnTick="Button1_Click"></asp:Timer>--%>
</div>
</form>
<!-- -->
</asp:Content>

```

Default.aspx.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Windows;

public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        try
        {
            ObtencionDatos obtencion = new ObtencionDatos();
            obtencion.CerrarPuerto();
            obtencion.AbrePuerto();
            Label1.Text = "";
        }
        catch (Exception a)
        {
            string script = @"<script type='text/javascript'>
            alert('Lector desconectado');
            </script>";
            ScriptManager.RegisterStartupScript(this, typeof(Page), "alerta", script, false);
        }
    }

    //Sin uso...
    protected void Button1_Click(object sender, EventArgs e)
    {
        try
        {
            ObtencionDatos obtencion = new ObtencionDatos();
            Persona Per = new Persona();
            string reg_tarjeta = "";

            reg_tarjeta = obtencion.TomarDatos();

```

```

        reg_tarjeta = reg_tarjeta.ToString().Trim();

if (reg_tarjeta != "Error")
{
    int filas = Per.BuscaTarjeta(reg_tarjeta, "");

    if (filas > 0)
    {
        if (Per.RegistraHora(reg_tarjeta))
        {
            Label1.Text = "Marca registrada.. Bip!";
        }
    }
    else
    {
        Label1.Text = "Fallo registro, marque tarjeta nuevamente.";
    }
}
else
{
    Label1.Text = "Tarjeta invalida";
}
}
reg_tarjeta = "Error";
}
catch (Exception a) { }
}
//.....

```

```

protected void Timer1_Tick(object sender, EventArgs e)
{
    try
    {
        ObtencionDatos obtencion = new ObtencionDatos();
        Persona Per = new Persona();
        string reg_tarjeta = "";

        reg_tarjeta = obtencion.TomarDatos();
        reg_tarjeta = reg_tarjeta.ToString().Trim();

        if (reg_tarjeta != "Error")
        {
            int filas = Per.BuscaTarjeta(reg_tarjeta, "");

            if (filas > 0)
            {
                if (Per.RegistraHora(reg_tarjeta))
                {
                    Label1.Text = "Marca registrada.. Bip!";
                }
            }
            else
            {
                Label1.Text = "Fallo registro, marque tarjeta nuevamente.";
            }
        }
        else
        {
            Label1.Text = "Tarjeta invalida";
        }
    }
    reg_tarjeta = "Error";
}
catch (Exception a) { }
}

```

```
}
```

▪ **Ingreso Personal**

FormularioPersonal.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/MasterPage.master"
AutoEventWireup="true" CodeFile="FormularioPersonal.aspx.cs"
Inherits="FormularioPersonal" %>
<%@ Register Assembly="AjaxControlToolkit" Namespace="AjaxControlToolkit"
TagPrefix="cc1" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" runat="Server">
</asp:Content>

<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
runat="Server">

<script src="style/validaRut.js"></script>
<script src="style/SoloLetras.js"></script>

<!-- Formulario -->
<form runat="server">
<h3 class="display-4 mt-2 ml-2">Agregar Personal</h3>

<!-- -->
<asp:ScriptManager ID="ScriptManager1" runat="server">
</asp:ScriptManager>
<!-- -->

<div class="row mt-5 ml-2">
<div class="col-6">

<div class="form-group row">
<label for="" class="col-3 col-form-label">Rut</label>
<input id="txt_rut" type="text" placeholder="11111111-k" class="form-control col-
4"
maxlength="10" oninput="checkRut(this)" runat="server" />

<!-- Lupa -->
<asp:ImageButton ID="btn_mensaje" runat="server"
ImageUrl="~/imagenes/buscar.png"
Height="22" Width="22" CssClass="ml-1 mt-1"
OnClick="ImageButton1_Click" />

<!-- Modal -->
<asp:Panel ID="pnl_mensaje" runat="server" Style="width:330px; height:150px;
display:none" BackColor="#e9e9e9" >
<p class="lead ml-2">
<asp:Label runat="server" ID="lbl_mensaje">
</asp:Label>
</p>

<asp:Button ID="btn_aceptarMensaje" runat="server" Text="Aceptar" class="btn
btn-lg btn-dark ml-2"
```

```

                                OnClick="btn_aceptarMensaje_Click" autoPostBack
="true"/>
<asp:Button ID="btn_cancelar" runat="server" Text="Cancelar" class="btn btn-lg
btn-dark" autoPostBack="true" />
</asp:Panel>

<asp:HiddenField ID="HiddenField1" runat="server"/>

<cc1:ModalPopupExtender ID="pnl_mensaje_ModalPopupExtender"
runat="server"
    Enabled="True"
    TargetControlID="HiddenField1"
    BackgroundCssClass="modalBackground"
    PopupControlID="pnl_mensaje"
    CancelControlID="btn_cancelar"
>
</cc1:ModalPopupExtender>
<!-->
<!-- Lupa -->

</div>
<div class="form-group row">
<label for="" class="col-3 col-form-label">Nombre</label>
<input id="txt_nom" type="text" placeholder="Escriba su nombre" class="form-
control col-8"
maxlength="40" runat="server" onkeypress="return lettersOnly(event)"/>

</div>
<div class="form-group row">
<label for="" class="col-3 col-form-label">Apellido Paterno</label>
<input id="txt_ape_pat" type="text" placeholder="Apellido Paterno" class="form-
control col-8"
maxlength="40" runat="server" onkeypress="return lettersOnly(event)"/>

</div>
<div class="form-group row">
<label for="" class="col-3 col-form-label">Apellido Materno</label>
<input id="txt_ape_mat" type="text" placeholder="Apellido materno" class="form-
control col-8"
maxlength="40" runat="server" onkeypress="return lettersOnly(event)"/>

</div>
<div class="form-group row">
<label for="" class="col-3 col-form-label">Fecha Nacimiento</label>
<input id="fecha" type="date" class="form-control col-6"
max="1999-01-01" runat="server" />

</div>
<div class="form-group row">
<label for="" class="col-3 col-form-label">Email</label>
<input id="txt_email" type="email" placeholder="correo@email.cl" class="form-
control col-8"
maxlength="50" runat="server" />

</div>

<div class="form-group row mt-5 ">
<asp:Button ID="btn_aceptar" runat="server" Text="Aceptar" class="btn btn-lg
btn-dark "
                                OnClick="btn_aceptar_Click" />

</div>

```

```

</div>
<div class="col-6">

<div class="form-group row">
<label for="" class="col-3 col-form-label">Fono</label>
<input id="txt_fono" type="number" placeholder="Numero de contacto"
class="form-control col-5"
maxlength="12" runat="server" />

</div>

<div class="form-group row">
<label for="" class="col-3 col-form-label">Oficina</label>
<asp:DropDownList ID="ddl_oficina" runat="server" CssClass="form-control col-5"></asp:DropDownList>

</div>
<div class="form-group row">
<label for="" class="col-3 col-form-label">Cargo</label>
<asp:DropDownList ID="ddl_cargo" runat="server" CssClass="form-control col-5">
</asp:DropDownList>

</div>

<div class="form-group row">
<label for="" class="col-3 col-form-label">Tarjeta</label>
<asp:TextBox runat="server" id="Txt_tarjeta" placeholder="Tarjeta no registrada"
CssClass="form-control col-3"
MaxLength="20" Enabled="false" ></asp:TextBox>

<asp:Button ID="BtnRegTarjeta" runat="server" Text="Registrar" class="btn btn-lg
btn-dark col-2 ml-2"
OnClick="BtnRegTarjeta_Click"/>
</div>

</div>
</div>

</form>
<!-- Fin Formulario -->

</asp:Content>

```

FormularioPersonal.aspx.cs

```

using System;
using System.Data;
using System.Data.SqlClient;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

```

```

public partial class FormularioPersonal : System.Web.UI.Page
{
    Conexion conexion = new Conexion();
    Consultas consultas = new Consultas();
    Persona Per = new Persona();

    protected void Page_Load(object sender, EventArgs e)
    {
        SqlCommand cmd = new SqlCommand();
    try
        {
            if (!IsPostBack)
            {
                cmd.Connection = conexion.GetConexion();
                conexion.AbrirConexion();

                //LLENA EL DROPDOWNLIST DE CARGOS
                ddl_cargo.DataSource = Per.DevuelveCargos();
                ddl_cargo.DataTextField = "Nombre del Cargo";
                ddl_cargo.DataValueField = "Numero";
                ddl_cargo.DataBind();

                //LLENA EL DROPDOWNLIST DE OFICINA
                ddl_oficina.DataSource = Per.DevuelveOficina();
                ddl_oficina.DataTextField = "Dirección";
                ddl_oficina.DataValueField = "ID";
                ddl_oficina.DataBind();

                txt_rut.Disabled = false;
                conexion.CerrarConexion();
            }
        }
    catch (Exception) { }
    }

    protected void LimpiaPantalla()
    {
        txt_rut.Value = ""; txt_email.Value = "";
    }
}

```

```

        txt_nom.Value = ""; txt_ape_pat.Value = "";
        txt_fono.Value = ""; txt_ape_mat.Value = "";
    }

protected void btn_aceptar_Click(object sender, EventArgs e)
{
    try
    {
        if (txt_nom.Value == "" || txt_ape_pat.Value == "" || fecha.Value == "" ||
            txt_ape_mat.Value == "" || txt_fono.Value == "" || txt_email.Value ==
            "")
        {
            //Mensaje
            string script = @"<script type='text/javascript'>
            alert('Debe completar los campos.');

```

```

tarjeta, int.Parse(ddl_oficina.SelectedValue.ToString()),
int.Parse(ddl_cargo.SelectedValue.ToString()));

if (filas1 > 0)
    {
        //Mensaje
        string script = @"<script type='text/javascript'>
        alert('La operacion finalizó correctamente.');";
        ScriptManager.RegisterStartupScript(this, typeof(Page), "alerta", script, false);
        LimpiaPantalla();
    }
else
    {
        //Mensaje
        string script = @"<script type='text/javascript'>
        alert('La operacion no pudo finalizar correctamente.');";
        ScriptManager.RegisterStartupScript(this, typeof(Page), "alerta", script, false);
    }
    }
    //si existe el rut pero "activo" esta con "N" entra
else if (ds.Tables[0].Rows.Count > 0 &&
ds.Tables[0].Rows[0]["ACTIVO"].Equals("N"))
{
    int filas2 = Per.ActivarPersonalInactivo(int.Parse(rut), tarjeta,
int.Parse(ddl_cargo.SelectedValue.ToString()));

    if (filas2 > 0)
        {
            //Mensaje
            string script = @"<script type='text/javascript'>
            alert('La operacion finalizó correctamente.');";
            ScriptManager.RegisterStartupScript(this, typeof(Page), "alerta", script, false);
            LimpiaPantalla();
        }
    else
        {

```

```

        //Mensaje
        string script = @"<script type='text/javascript'>
        alert('La operacion no pudo finalizar correctamente.');
```

```

protected void BtnRegTarjeta_Click(object sender, EventArgs e)
{
    try
    {
        ObtencionDatos obtencion = new ObtencionDatos();
        obtencion.CerrarPuerto();
        obtencion.AbrePuerto();
        Txt_tarjeta.Text = obtencion.TomarDatos();
    }
    catch (Exception)
    {
        //Mensaje
        string script = @"<script type='text/javascript'>
        alert('Lector desconectado.');

```

```

if (ds.Tables[0].Rows.Count > 0)
{
    //Mensaje modal
    lbl_mensaje.Text = "El rut se encuentra en estado inactivo. ¿Desea
recuperar sus datos?.";
    pnl_mensaje_ModalPopupExtender.Show();
}
else
{
    DataSet dsV = Per.DevuelvePersonal(int.Parse(rut));
    if (dsV.Tables[0].Rows.Count > 0)
    {
        //Mensaje
        string script = @"<script type='text/javascript'>
alert('El rut ya se encuentra activo.');

```

```

string rut = txt_rut.Value.Remove(txt_rut.Value.IndexOf('-'), 2);
DataSet dsLL = Per.DevuelveTodoPersonalInactivo(rut);

//se llenan los campos
txt_nom.Value = dsLL.Tables[0].Rows[0]["NOMBRE"].ToString();
txt_ape_pat.Value = dsLL.Tables[0].Rows[0]["APELLIDO
PATERNO"].ToString();
txt_ape_mat.Value = dsLL.Tables[0].Rows[0]["APELLIDO
MATERNO"].ToString();
txt_email.Value = dsLL.Tables[0].Rows[0]["MAIL"].ToString();
txt_fono.Value = dsLL.Tables[0].Rows[0]["TELEFONO"].ToString();
//fecha.Text = ds.Tables[0].Rows[0]["FECHA NACIMIENTO"].ToString();

ddl_cargo.DataSource = Per.DevuelveCargos();
ddl_cargo.DataTextField = "Nombre del Cargo";
ddl_cargo.DataValueField = "Numero";
ddl_cargo.DataBind();

ddl_oficina.DataSource = Per.DevuelveOficina();
ddl_oficina.DataTextField = "Dirección";
ddl_oficina.DataValueField = "ID";
ddl_oficina.DataBind();

DataSet ds2 = Per.DevuelveFechaNac(int.Parse(rut));
fecha.Value = ConvierteFecha(ds2.Tables[0].Rows[0][0].ToString());
}
catch (Exception a) { txt_rut.Disabled = false; }
}

public string ConvierteFecha(string ds)
{
string mes, dia, anno, fecha;
mes = ds.Remove(2);
dia = ds.Substring(3, 2);
anno = ds.Substring(6, 4);
fecha = anno + "-" + mes + "-" + dia;
return fecha; }
//-----
}

```

- **Consultar Personal**

FormPersonalConsultar.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/MasterPage.master"
AutoEventWireup="true" CodeFile="FormPersonalConsultar.aspx.cs"
Inherits="FormPersonalConsultar" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">

<script src="style/SoloLetras.js"></script>
<script src="style/SoloNumeros.js"></script>

<form runat="server">

<h3 class="display-4 mt-2 ml-2">Consultar Personal</h3>

<div class="row">

<div class="form-group row ml-3 my-3 ">
<h5 class="col-2"><label for="" class="col-form-label ">Buscador</label></h5>
<asp:TextBox runat="server" ID="txt_rut" placeholder="Rut (sin dígito
verificador)" CssClass="form-control col-4 "
onkeypress="return NumberOnly(event)" MaxLength="8"></asp:TextBox>

<asp:TextBox runat="server" ID="txt_nom" placeholder="Nombre"
CssClass="form-control col-3 ml-1"
onkeypress="return lettersOnly(event)" MaxLength="30"></asp:TextBox>

<asp:Button ID="BtnAceptar" runat="server" Text="Consultar" class="btn btn-lg
btn-dark col-2 ml-2"
OnClick="BtnAceptar_Click" />

</div>

<div class="col-9 ml-2 my-3">

<asp:GridView ID="dgv_consulta" runat="server" AutoGenerateColumns="true"
ShowHeaderWhenEmpty="True"
Font-Size="Small" CssClass="table table-responsive table-
inverse">

</asp:GridView>

</div>
</div>
</form>
</asp:Content>
```

FormPersonalConsultar.aspx.cs

```

using System;
using System.Data;
using System.Data.SqlClient;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class FormPersonalConsultar : System.Web.UI.Page
{
    Conexion conexion = new Conexion();
    Consultas consultas = new Consultas();
    Persona per = new Persona();

    protected void Page_Load(object sender, EventArgs e)
    {

    }

    protected void BtnAceptar_Click(object sender, EventArgs e)
    {
        try
        {
            DataSet ds = per.DevuelveTodoPersonal(txt_rut.Text, txt_nom.Text);
            dgv_consulta.DataSource = ds;
            dgv_consulta.DataBind();
        }
        catch (Exception a) { }
    }
}

```

▪ **Ingreso Reserva de Sala**FormularioReservaSala.aspx

```

<%@ Page Title="" Language="C#" MasterPageFile="~/MasterPage.master"
AutoEventWireup="true" CodeFile="FormularioReservaSala.aspx.cs"
Inherits="FormularioReservaSala" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">

<form runat="server">

<h3 class="display-4 mt-2">Reserva de Salas</h3>

<div class="row mt-5 ml-2">
<div class="col-6 ">

```

```

<div class="form-group ">
<label for="" class="col-form-label">Hora inicio</label>
<asp:TextBox TextMode="Time" CssClass="form-control col-6" ID="txt_inicio"
runat="server" required="required">
</asp:TextBox>
</div>

<div class="form-group ">
<label for="" class="col-form-label">Nombre</label>
<asp:DropDownList ID="ddl_rut" runat="server" CssClass="form-control col-6"
AutoPostBack="true" >
</asp:DropDownList>

</div>

<div class="form-group ">
<label for="" class="col-form-label">Fecha</label>
<asp:TextBox ID="txt_fecha" TextMode="Date" runat="server" class="form-control
col-6" required="required"
OnTextChanged="txt_fecha_TextChanged" AutoPostBack="true"
></asp:TextBox>
</div>
</div>

<div class="col-6 ">

<div class="form-group ">
<label for="" class="col-form-label">Hora termino</label>
<asp:TextBox TextMode="Time" CssClass="form-control col-6" ID="txt_termino"
runat="server" required="required" OnInput="txt_termino_Load">
</asp:TextBox>
</div>

</div>

<div class="col-6 ">

<div class="form-group">
<label for="txt_descrip">Descripcion Breve</label>
<textarea class="form-control col-7" id="txt_descrip" rows="3" maxlength="80"
runat="server"></textarea>
</div>

<div class="form-group row mt-3 ">
<asp:Button ID="btn_aceptar" runat="server" Text="Aceptar" class="btn btn-lg
btn-dark " OnClick="btn_aceptar_Click"/>
</div>

</div>
</div>

</form>

</asp:Content>

```

FormularioReservaSala.aspx.cs

```

using System;
using System.Data;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class FormularioReservaSala : System.Web.UI.Page
{
    Persona per = new Persona();

    protected void Page_Load(object sender, EventArgs e)
    {
        try
        {
            if(!IsPostBack)
            {
                ddl_rut.DataSource = per.DevuelveRutyNombreParaReseva();
                ddl_rut.DataTextField = "NOMBRE";
                ddl_rut.DataValueField = "RUT";
                ddl_rut.DataBind();
            }
        }
        catch(Exception a) { }
    }

    protected void btn_aceptar_Click(object sender, EventArgs e)
    {
        try
        {
            DateTime inicio = DateTime.Parse(txt_inicio.Text);
            DateTime termino = DateTime.Parse(txt_termino.Text);

            if (inicio.CompareTo(termino) > 0 || inicio.CompareTo(termino) == 0)
            {
                //Mensaje
                string script = @"<script type='text/javascript'>
                alert('La hora de termino debe ser superior a la de inicio.');

```

```

//si se modificaron filas
if (filas > 0)
{
//Mensaje
string script = @"<script type='text/javascript'>
alert('La operacion se finalizó correctamente.');"
</script>";
ScriptManager.RegisterStartupScript(this, typeof(Page), "alerta", script, false);

limpiar();
}
else
{
//Mensaje
string script = @"<script type='text/javascript'>
alert('La operacion no se pudo realizar, la hora puede estar ya registrada.');"
</script>";
ScriptManager.RegisterStartupScript(this, typeof(Page), "alerta", script, false);
}
}
}
catch(Exception a) { }
}

```

```

protected void txt_termino_Load(object sender, EventArgs e)
{
try
{
DateTime inicio = DateTime.Parse(txt_inicio.Text);
DateTime termino = DateTime.Parse(txt_termino.Text);

if (inicio.CompareTo(termino) > 0 || inicio.CompareTo(termino) == 0)
{
//Mensaje
string script = @"<script type='text/javascript'>
alert('La hora de termino debe ser superior a la de inicio.');"
</script>";
ScriptManager.RegisterStartupScript(this, typeof(Page), "alerta", script, false);
}
}
catch (Exception a) { }
}

```

```

protected void txt_fecha_TextChanged(object sender, EventArgs e)
{
try
{
DateTime fecha = DateTime.Parse(txt_fecha.Text);
if (fecha.Date.Date < DateTime.Now.Date)
{
//Mensaje
string script = @"<script type='text/javascript'>
alert('La fecha debe ser desde hoy en adelante.');"
</script>";
ScriptManager.RegisterStartupScript(this, typeof(Page), "alerta", script, false);

txt_fecha.Text = "";
}
}
}
catch (Exception a) { }
}

```

```
protected void limpiar()
{
    txt_descrip.Value = "";
    txt_fecha.Text = "";
    txt_inicio.Text = "";
    txt_termino.Text = "";
}
}
```

▪ **Ingreso Justificación**

FormularioJustificar.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/MasterPage.master"
AutoEventWireup="true" CodeFile="FormularioJustificar.aspx.cs"
Inherits="FormularioJustificar" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">
```

```
<script src="style/validaRut.js"></script>
```

```
<form runat="server">
<h3 class="display-4 mt-2">Justificar Inasistencia </h3>
```

```
<div class="row mt-5 ml-2">
<div class="col-10">
```

```
<div class="form-group row">
<label for="" class="col-2 col-form-label">Rut</label>
<asp:TextBox runat="server" ID="txt_rut" placeholder="11111111-k"
CssClass="form-control col-3" MaxLength="10" oninput="checkRut(this)"
></asp:TextBox>
</div>
```

```
<div class="form-group row">
<label for="" class="col-2 col-form-label">Fecha </label>
<asp:TextBox ID="fecha" runat="server" TextMode="Date" class="form-control
col-3">
</asp:TextBox>
</div>
```

```
<div class="form-group row mt-5 ml-3" >
<asp:Button ID="btn_aceptar" runat="server" Text="Aceptar" class="btn btn-lg
btn-dark" OnClick="btn_aceptar_Click" />
</div>
```

```
<asp:gridview id="dgv_consulta" CssClass="table table-responsive table-inverse
ml-3 mt-5 mr-3" runat="server"
AutoGenerateColumns="false" PagerSettings-
Mode="NumericFirstLast" ShowHeaderWhenEmpty="True"
OnRowCancelingEdit="dgv_consulta_RowCancelingEdit">
```

```

        OnRowEditing="dgv_consulta_RowEditing"
OnRowUpdating="dgv_consulta_RowUpdating"
        DataKeyNames="id_just" AllowPaging="true" PageSize="10"
OnPageIndexChanging="dgv_consulta_PageIndexChanging">

<Columns>
<asp:TemplateField HeaderText="ID" Visible="false">
<ItemTemplate>
<asp:Label ID="lbl_id" runat="server" Text='<% #Eval("ID_JUST") %>'
Visible="false"> </asp:Label>
</ItemTemplate>
</asp:TemplateField>

<asp:TemplateField HeaderText="Rut">
<ItemTemplate>
<asp:Label ID="lbl_rut" runat="server" Text='<% #Eval("RUT")
%>'> </asp:Label>
</ItemTemplate>
</asp:TemplateField>

<asp:TemplateField HeaderText="Fecha">
<ItemTemplate>
<asp:Label ID="lbl_fecha" runat="server" Text='<% #Eval("FECHA")
%>'> </asp:Label>
</ItemTemplate>
</asp:TemplateField>

<asp:TemplateField HeaderText="Estado">
<ItemTemplate>
<asp:Label ID="lbl_estado" runat="server" Text='<% #Eval("JUSTIFICADO")
%>'> </asp:Label>
</ItemTemplate>
</asp:TemplateField>

<asp:TemplateField HeaderText="Descripcion">
<ItemTemplate>
<asp:Label ID="lbl_descrip" runat="server" Text='<% #Eval("DESCRIPCION")
%>'> </asp:Label>
</ItemTemplate>
<EditItemTemplate>
<asp:TextBox ID="txt_descrip" runat="server" Text='<% #Eval("DESCRIPCION")
%>'> </asp:TextBox>
</EditItemTemplate>
</asp:TemplateField>

<asp:TemplateField>
<ItemTemplate>
<asp:ImageButton ID="btn_editar" ImageUrl="~/imagenes/edit2.png"
Height="30" Width="30" runat="server" CommandName="Edit" ToolTip="Edit"/>
</ItemTemplate>
<EditItemTemplate>
<asp:ImageButton ID="btn_guardar" ImageUrl="~/imagenes/save2.png"
Height="30" Width="30" runat="server" CommandName="Update"
ToolTip="Update"/>

<asp:ImageButton ID="btn_cancelar" ImageUrl="~/imagenes/cancel2.png"
Height="26" Width="26" runat="server" CommandName="Cancel"
ToolTip="Cancel"/>
</EditItemTemplate>
</asp:TemplateField>

</Columns>
</asp:gridview>
</div>

```

```

</div>
</form>
</asp:Content>

```

FormularioJustificar.aspx.cs

```

using System;
using System.Data;
using System.Data.SqlClient;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Net;

public partial class FormularioJustificar : System.Web.UI.Page
{
    Persona per = new Persona();
    int sw;

    protected void Page_Load(object sender, EventArgs e)
    {
    }

    protected void btn_aceptar_Click(object sender, EventArgs e)
    {
        try
        {
            DateTime fec = DateTime.Now;
            string fech = Convert.ToString(fec).Substring(11, 5);

            if (fech.Equals("18:00"))
            {
                sw = 1;

                if (sw == 1)
                {
                    int filas = per.LlenaTablaJustificacion();
                    sw = 0;
                }

                if (txt_rut.Text == "" && fecha.Text == "")
                {
                    DataSet ds = per.LlenaGrillaJust(txt_rut.Text, fecha.Text);

                    if (ds.Tables.Count > 0)
                    {
                        {
                            dgv_consulta.DataSource = ds;
                            dgv_consulta.DataBind();
                        }
                    }
                    else
                    {
                        //Mensaje
                        string script = @"<script type='text/javascript'>
                        alert('No se encontrador registros.');

```

```

else if (txt_rut.Text == "" && fecha.Text != "")
{
    string rut = "";
    DataSet ds = per.LlenaGrillaJustYaJustificada(rut, fecha.Text);

    if (ds.Tables[0].Rows.Count > 0)
    {
        dgv_consulta.DataSource = ds;
        dgv_consulta.DataBind();
    }
    else
    {
        //Mensaje
        string script = @"<script type='text/javascript'>
        alert('No se encontrador registros con los datos ingresados.');

```

```

        dgv_consulta.DataSource = ds;
dgv_consulta.DataBind();
    }

protected void dgv_consulta_RowUpdating(object sender, GridViewUpdateEventArgs
e)
{
try
    {
int codigo =
Convert.ToInt32(dgv_consulta.DataKeys[e.RowIndex].Value.ToString());
string des = (dgv_consulta.Rows[e.RowIndex].FindControl("txt_descrip") as
TextBox).Text.Trim();

int filas = per.ModificarJustificacion(codigo,des);
dgv_consulta.EditIndex = -1;

        DataSet ds = per.LlenaGrillaJust(txt_rut.Text, fecha.Text);

        dgv_consulta.DataSource = ds;
dgv_consulta.DataBind();
    }
catch(Exception a) { }
}

protected void dgv_consulta_PageIndexChanging(object sender,
GridViewPageEventArgs e)
{
    dgv_consulta.PageIndex = e.NewPageIndex;

DataSet ds = per.LlenaGrillaJust(txt_rut.Text, fecha.Text);
    dgv_consulta.DataSource = ds.Tables[0];
dgv_consulta.DataBind();
}
}

```

▪ **Consultar Tarjetas**

FormTarjetasConsultar.aspx

```

<%@ Page Title="" Language="C#" MasterPageFile="~/MasterPage.master"
AutoEventWireup="true" CodeFile="FormTarjetasConsultar.aspx.cs"
Inherits="FormTarjetasConsultar" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">

<form runat="server">

<h3 class="display-4 mt-2 ml-2">Consultar Tarjetas</h3>

<div class="row ml-3 my-4">
<div class="col-6">

```

```

<asp:GridView ID="dgv_consulta" runat="server" AutoGenerateColumns="true"
ShowHeaderWhenEmpty="True"
                Font-Size="medium" CssClass="table table-responsive table-
inverse">

</asp:GridView>

</div>
</div>
</form>

</asp:Content>

```

FormTarjetasConsultar.aspx.cs

```

using System;
using System.Data;
using System.Data.SqlClient;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class FormTarjetasConsultar : System.Web.UI.Page
{
    Persona per = new Persona();

    protected void Page_Load(object sender, EventArgs e)
    {
        try
        {
            DataSet ds = per.DevuelveTarjeta();
            dgv_consulta.DataSource = ds;
            dgv_consulta.DataBind();
        }
        catch (Exception) { }
    }
}

```