

UNIVERSIDAD POLITÉCNICA DE PUEBLA
Ingeniería en Informática



**Proyecto de Estancia Práctica en
Desarrollador en Sistemas de Software y
Administrador de Redes**

“Interfaces entre Reloj checador y el sistema de Nóminas”

Área temática del CONACYT: VII
Ingenierías y tecnologías

Presenta:

Laura Zacatzontetl Hernández

Asesor técnico

Ing. Daniel Paniagua González

Asesor académico

M.C Rebeca Rodríguez Huesca

Juan C. Bonilla, Puebla, México.

19 de diciembre del 2018

Resumen

En este documento se describirá el problema, los objetivos generales y específicos, metodología usada y el desarrollo de cada una en los módulos solicitud y validación de horas extra, dichos módulos ayudaran en el proceso de nómina de la empresa Seamless Global Solutions (SGS), este software permitirá registrar y validar las horas extras de los empleados que soliciten tiempo extra para posteriormente ser diferenciadas de las horas trabajadas normalmente y así facilitar el proceso de nómina.

Índice

1. Introducción	4
1.1.Descripción del problema o necesidad	4
1.2 Justificación	5
1.3 Objetivo General y Específicos	5
2. Metodología y herramientas.....	7
2.1 Modelo en cascada	7
2.2 Herramientas Tecnologías Utilizadas	8
3. Resultados	11
3.1 Análisis y definición de requerimientos	11
3.2 Diseño del sistema del Software	14
3.3 Implementación y prueba de unidades.....	21
3.4 Integración y prueba del sistema	26
3.5 Funcionamiento y mantenimiento	27
4. Conclusiones y recomendaciones	40
5. Anexos	41
6. Referencias bibliográficas	48

1. Introducción

En este capítulo se describe el problema que tiene el área de recursos humanos de la empresa Seamless Global Solutions (SGS), en cuanto al proceso de generar las nóminas de los empleados, así como también se describe la solución propuesta, las herramientas con las que se cuenta para realizar el proyecto, el objetivo general y los objetivos específicos que ayudarán durante el desarrollo del proyecto.

1.1. Descripción del problema o necesidad

En el área de recursos humanos se lleva a cabo el desarrollo de nóminas de todos los empleados de la empresa Seamless Global Solutions (SGS), este proceso es realizado por una sola persona; sin embargo, el número de empleados es aproximadamente de 350, por lo cual a la encargada de realizar las nóminas se le dificulta validar los datos de asistencia y horas trabajadas cada catorce días de todos los empleados, así como también calcular el pago, puesto que existen más de dos turnos de trabajo y algunos empleados optan por turnos extra, así como también se realizan bonificaciones a algunos empleados por actividades destacadas; cabe mencionar que las bonificaciones no son registradas en la base de datos, sino que son registradas manualmente en un reporte de bonificaciones, también es necesario mencionar que la empresa cuenta con un área de comedor y los empleados que hacen uso del comedor se les realiza un descuento a su nómina, lo que requiere cálculos extra en su nómina. Actualmente se cuenta con un sistema desarrollado especialmente para la empresa, el cual tiene una pantalla para realizar las nóminas, aunque esta pantalla ayuda a la realización de nóminas no es del todo eficiente puesto sólo calcula las horas desde que el empleado registra la entrada en el reloj checador hasta cuando registra la hora de salida, por lo cual el reloj checador no diferencía entre las horas trabajadas y las horas extras.

Se pretende desarrollar una pantalla que registre las horas extra de los trabajadores en una base de datos la cual generara un reporte de los empleados que tienen registrado horas extras, también se desarrollará una pantalla que valore que efectivamente los empleados cumplieron las horas extras y se generara el reporte oficial con el cual se comprobará las horas extras, posteriormente con las horas extras registradas y las horas trabajadas normalmente se podrá calcular la nómina.

1.2 Justificación

El realizar la nómina de una manera eficaz es un punto muy importante puesto que el realizar el pago puntualmente a los empleados los motiva a seguir trabajando correctamente, si el proceso de su nómina se retrasa los empleados muestran molestia o disgusto por el retraso de sus pagos afectando en el proceso del producto, es importante mejorar el módulo del proceso de nóminas puesto que en algunas ocasiones ha habido errores humanos que causan problemas disfuncionales como: el extravío de la hoja de horas extras o el retraso del proceso de nómina, una solución óptima es que la encargada de realizar las nóminas tome los datos de la base de datos sin tener que ir a buscar a cada jefe de área para recopilar la información. Se desarrollará una pantalla que registre las horas extras en una base de datos, en ésta el jefe de área registrará el número de horas extras que solicitan los empleados, también en esta pantalla se podrá generar un reporte el cual deberá ser impreso para que los empleados firmen que están de acuerdo y no han sido obligados, también se realizará otra pantalla que al día siguiente de la jornada de trabajo el jefe de área validará que el empleado efectivamente realizó las horas extras, para facilitar el proceso de nómina se podrá generar un crystal report que contendrá los campos que necesita la encargada para el proceso de nómina. La empresa tendrá mayor eficacia en el área de recursos humanos puesto que la encargada de nómina tardará menos en la realización de la nómina y tendrá más tiempo para resolver otros asuntos relacionados con su área. La relevancia del estudio para el usuario es que debe contar con conocimientos previos en computación, así como la facilidad de adaptarse a nuevos entornos y software de trabajo.

Las herramientas con las que se cuenta para el desarrollo son una computadora portátil con software especializado en desarrollo de aplicaciones de escritorio (Visual Studio) así como también para el desarrollo de base de datos (SQL Server), también se cuenta con acceso a internet.

El proyecto impacta tanto en los empleados como en la empresa puesto que el tener un proceso de nómina eficiente evitará problemas con los empleados, de esta manera la empresa se verá beneficiada puesto que si los empleados trabajan contentos con la empresa se desempeñarán correctamente en la creación de productos de calidad.

1.3 Objetivo General y Específicos

Desarrollar nuevas funciones que ayuden a el proceso de nómina de la empresa Seamless Global Solutions (SGS) para mejorarlo.

Objetivos específicos:

- Analizar el funcionamiento del sistema actual

- Analizar las mejoras que se realizarán a el sistema.
- Diseñar la interfaz gráfica de las horas extras en Visual Studio
- Programar la pantalla de horas extras
- Programar las validaciones de la pantalla de horas extras
- Programar y diseñar el reporte en crystal report de horas extras
- Programar la pantalla de validar horas extras
- Programar las validaciones de la pantalla de validar horas extras
- Programar y diseñar el reporte en crystal report de validar horas extras
- Evaluar junto con los asesores los avances del proyecto
- Realizar pruebas de funcionalidad de cada módulo programado
- Realizar pruebas de validación de cada módulo implementado
- Importar todos los módulos programados a el proyecto general ya desarrollado de la empresa

2. Metodología y herramientas

En este capítulo se describen las definiciones, características, ventajas y desventajas del modelo en cascada al igual que las herramientas tecnológicas Visual Studio y SQL Server para el desarrollo del proyecto.

2.1 Modelo en cascada

El modelo en cascada es un proceso de desarrollo secuencial, en el que el desarrollo de software se concibe como un conjunto de etapas que se ejecutan una tras otra. Se le denomina así por las posiciones que ocupan las diferentes fases que componen el proyecto, colocadas una encima de otra, y siguiendo un flujo de ejecución de arriba hacia abajo, como una cascada. [1]

2.1.1 Etapas del modelo [2]:

Las principales etapas como se muestran en la figura 1 de este modelo se transforman en actividades fundamentales de desarrollo:

1. Análisis y definición de requerimientos: Los servicios, restricciones y metas del sistema se definen a partir de las consultas con los usuarios. Entonces se definen en detalle y sirven como una especificación del sistema.
2. Diseño del sistema del Software: el proceso de diseño del sistema divide los requerimientos en sistemas hardware o software. Establece una arquitectura completa del sistema software y sus relaciones.
3. Implementación y prueba de unidades: Durante esta etapa, el diseño del software se lleva a cabo como un conjunto o unidades de programas. La prueba de unidades implica verificar que cada una cumpla su especificación.
4. Integración y prueba del sistema: los programas o unidades individuales de programas se integran y prueban con un sistema completo para asegurar que se cumplan los requerimientos del software. Después de las pruebas, el sistema software se entrega al cliente.
5. Funcionamiento y mantenimiento: por lo general es la fase más larga del ciclo de vida. El sistema se instala y se pone en funcionamiento práctico. El mantenimiento implica corregir errores no descubiertos en las etapas anteriores del ciclo de vida, mejorar la implementación de las unidades del sistema y resaltar los servicios del sistema una vez que se descubran nuevos requerimientos.

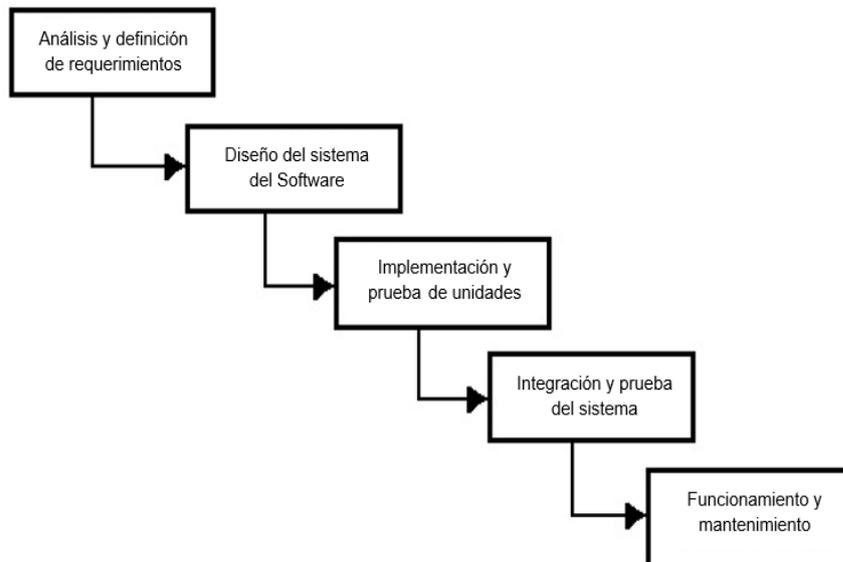


Figura 1. Las etapas del modelo en cascada

2.1.2 Características del modelo [3]:

- Primer modelo empleado por Walter Royce's 1970, también denominado ciclo de vida clásico y modelo lineal secuencial.
- Consiste en la ejecución secuencial de una serie de fases.
- Cada fase genera documentación para la siguiente la documentación debe ser aprobada.
- Una fase no comienza hasta que la anterior haya terminado.
- Requiere disponer de unos requisitos completos y precisos desde el inicio del desarrollo.

2.1.3 Ventajas del modelo de cascada [4]:

- Puedes comenzar con el software con bastante rapidez.
- Estimar calendarios y presupuestos con mayor precisión.
- Lograr un nivel de satisfacción del cliente más elevado que otros enfoques, desde el inicio.

2.1.4 Desventajas del modelo de cascada [3]:

- Los cambios son costosos de implementar.
- El modelo de cascada está conectado a altos esfuerzos.
- Las iteraciones requieren mucho esfuerzo y reelaboración.
- Requiere disponer de unos requisitos completos y precisos desde el inicio del desarrollo.

2.2 Herramientas Tecnológicas Utilizadas

Las herramientas tecnológicas son importantes en el desarrollo de proyectos informáticos a continuación se describen las herramientas que se utilizaron en el desarrollo de este proyecto.

2.2.1 Visual Studio [5]:

Es un editor gratuito de código fuente para desarrollo y depuración admite más de 30 lenguajes diferentes de programación y bases de datos, algunos de los cuales son: JavaScript, C #, C ++, PHP, Java, HTML, R, CSS, SQL, Markdown, TypeScript, Less, Sass, JSON, XML y Pitón.

Características [5]:

Incluye navegación, soporte de teclado con enlaces personalizables, resaltado de sintaxis, correspondencia de corchetes, sangría automática y fragmentos, con soporte para docenas de idiomas, comprensión y navegación, y refactorización de código.

Ventajas [5]:

- Soporte para desarrollo de ASP.NET Core con C #, Node.js con TypeScript y JavaScript
- Herramientas para tecnologías web: HTML, CSS, Less, Sass y JSON
- Se integra con gestores de paquetes y repositorios
- La facilidad del lenguaje permite crear aplicaciones para Windows de manera eficaz y menor inversión en tiempo como en dinero.
- Depuración integrada y optimizada
- Permite generar librerías dinámicas
- Provee soporte para apps web, Android, iOS, Windows Phone. Además de Visual Studio Tools

Desventajas

- La descarga es muy tardada a el igual que la instalación
- Requiere licencia

2.2.2 Microsoft SQL Server [6]

Microsoft SQL Server es un sistema de administración y análisis de bases de datos relacionales de Microsoft para soluciones de comercio electrónico, línea de negocio y almacenamiento de datos.

Características [7]:

- Alta Disponibilidad y Recuperación
- Facilidad de Uso
- Administración de Datos Maestros
- Calidad de Datos
- Procesamiento de Eventos Complejos

Ventajas [8]:

- Es útil para manejar y obtener datos de la red de redes.
- Nos permite olvidarnos de los ficheros que forman la base de datos.

- Si trabajamos en una red social nos permite agregar otros servidores de SQL Server. Por ejemplo, dos personas que trabajan con SQL Server,
- SQL permite administrar permisos a todo.

Desventajas [8]:

- Utiliza mucho la memoria RAM para las instalaciones
- Tamaño de página fijo y demasiado pequeño

Razones por las cuales utilizare la metodología cascada son los siguientes:

1. Análisis y definición de requerimientos: Aunque los requerimientos ya están en su mayoría establecidos no está demás analizar los requerimientos nuevamente con la encargada de nómina puesto que, aunque el asesor informo sobre los requerimientos, es necesario que la programadora aclare cualquier duda que tenga para poder avanzar correctamente con el siguiente paso.
 2. Diseño del sistema del Software: A el igual que los requerimientos el diseño de interfaz ya están estandarizados, puesto que las interfaces que se realizarán serán incorporadas a el sistema ya existente.
 3. Implementación y prueba de unidades: A el terminar de programar cada pantalla se le realizara las pruebas de validación de cada campo, también se validaran las fechas puesto que no se admitirán fechas pasadas en la pantalla de solicitud de horas extras, y se realizaran las pruebas de consistencia de datos en cada pantalla que inserte los datos en la base de datos, así como también al finalizar la programación de los reportes se verificara que los datos cargados en el reporte sean correctos.
 4. Integración y prueba del sistema: Las interfaces serán incorporadas a el sistema ya existente, se realizarán las pruebas de conexión a el servidor para posteriormente actualizarlo en cada una de las terminales.
 5. Funcionamiento y mantenimiento: Se instalará el sistema como los nuevos módulos en la terminal de la encargada de nóminas donde se pondrá a prueba para verificar que cumplan los requerimientos que fueron establecidos, si la encargada detecta algún error lo tendrá que comunicar con el asesor del proyecto para que el programador corrija los errores.
- Las razones por las que se eligió Visual Studio son porque en él se pueden crear aplicaciones de escritorio para Windows de manera eficaz, así como también los encargados del desarrollo de aplicaciones de la empresa Seamless Global Solution establecen este IDE para el desarrollo de las aplicaciones en lenguaje C#.
 - La razón por la cual se eligió SQL Server es porque todo el sistema que ha sido desarrollado usa SQL Server para el área departamento de tecnología de información es importante estandarizar por completo sus aplicaciones de escritorio

3. Resultados

En este capítulo se describe como se llevaron a cabo cada una de las etapas del modelo en cascada, también contiene imágenes en la etapa de implementación y prueba de unidades con sus respectivas descripciones que sirven de ayuda para comprender el funcionamiento y validación de cada módulo.

3.1 Análisis y definición de requerimientos

A continuación, se describen un análisis detallado del planteamiento del problema, módulos a desarrollar, alcances y limitaciones, usuarios, requerimientos funcionales y no funcionales.

Planteamiento del problema

La encargada de realizar el proceso de nómina realiza las siguientes funciones cada catorce días:

1. Elaborar nóminas
2. Control de asistencias de los empleados
3. Realizar bonificaciones
4. Control de horas extras
5. Control de retardos

Por lo que no se cuenta con el tiempo suficiente para tener control de cada una de las funciones puesto que son aproximadamente 350 empleados.

Módulos a desarrollar

Se desarrollarán nuevos módulos que permitirán automatizar y ahorrar tiempo en el proceso de horas extras estos módulos serán integrados en el sistema SIIP. Los procesos se muestran en la figura 2 y 3.

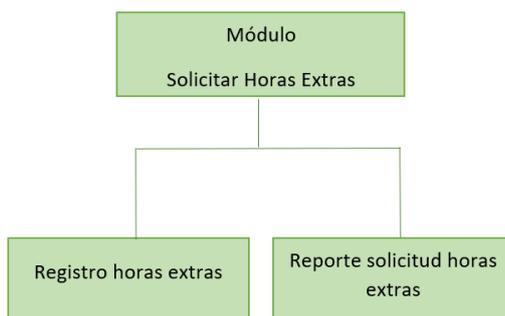


Figura 2. Procesos del módulo solicitar horas extras

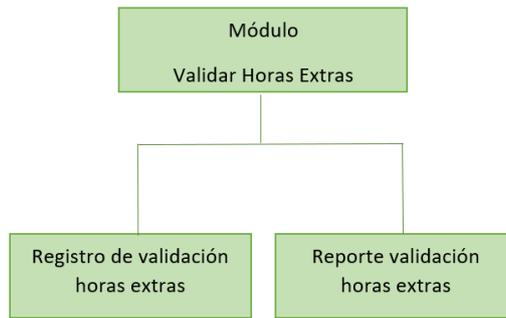


Figura 3. Procesos del módulo solicitar horas extras

Alcances y limitaciones

Alcances

El módulo solicitud y validación horas extras permitirán:

- Registrar horas extras del empleado que las solicite
- Validar horas extras
- Generar reportes

Limitaciones

- Resistencia al uso del software
- Mal uso del software

Usuarios de los módulos solicitud y validación de horas extras

- Encargada del proceso de nómina con altos conocimientos en computación y base de datos.
- Jefes de área con conocimientos en computación

Requerimientos

En esta etapa se explicarán los requerimientos funcionales y no funcionales para comprender mejor las necesidades del usuario

Requerimientos funcionales

RF1: Registro horas extras

- Seleccionar la fecha solicitada
- Teclear el número de horas extras
- Teclear las observaciones si es necesario
- Seleccionar el botón guardar
- Seleccionar el botón imprimir

RF2: Reporte solicitud horas extras

- Seleccionar la fecha
- Seleccionar el botón imprimir

RF3: Registro de validación horas extras

- Seleccionar la fecha
- Modificar horas extras si es necesario
- Seleccionar el Checkbox si se desea validar
- Teclear las observaciones si es necesario
- Seleccionar el botón guardar

RF4: Reporte validación horas extras

- Seleccionar la fecha
- Seleccionar el botón imprimir reporte

Requerimientos no funcionales

RNF1: Interfaz

- Interfaz amigable y sencilla con el usuario, además de que también contendrá el logo de la empresa, el logo del sistema SIIP y los colores ya establecidos para cada pantalla

RNF2: Desempeño

- Fácil instalación para el sistema operativo Windows

RNF3: Confiabilidad:

- Los reportes de solicitud y validación serán generados con el nombre del usuario que inicio sesión en el sistema SIIP y este nombre no será modificable

RNF4: Manejo de errores:

- Cada error que pueda surgir durante el uso de cada pantalla proporcionará un mensaje de ayuda con la información necesaria para resolver el error

RNF5: Restricciones:

- Sólo los jefes de área y la encargada de nómina tendrán permiso para acceder a los módulos solicitud y validación de horas extras

3.2 Diseño del sistema del Software

Diseño de interfaz

El diseño de interfaz ya está estandarizado, por lo cual se utilizará

- Fuente Microsoft Sans Serif, estilo de fuente normal, número ocho para todos los campos de cada pantalla.
- Fuente Lucida Handwriting, número diez para firma del programador.
- Propiedades y Colores para las pantallas son Back color control y border style fixed3D

Diseño arquitectónico

Arquitectura Cliente servidor [figura 4]: consiste básicamente en un cliente que realiza peticiones a otro programa (el servidor) que le da respuesta. La interacción cliente-servidor es el soporte de la mayor parte de la comunicación por redes. Ayuda a comprender las bases sobre las que están contruidos los algoritmos distribuidos. [9]

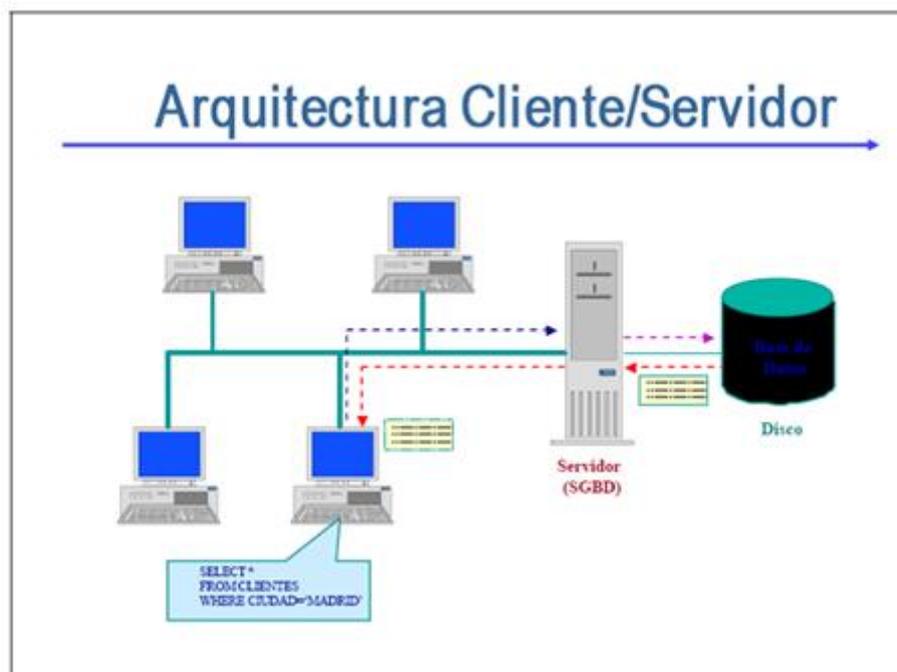


Figura 4. Arquitectura Cliente/Servidor

Este proyecto se desarrollará con la arquitectura Cliente/Servidor puesto que nos permite:

- Realizar tareas en zonas remotas, gracias a internet de manera cómoda y eficiente.
- Permite establecer una conexión directa a través de una red, con un servidor que aloja el programa.

- Muestra por pantalla toda la información que ha sido solicitada y procesada.

Casos de uso detallado

Los casos de uso detallado ayudaran a identificar los usuarios y funcionalidades de los módulos a programar. A continuación, en la figura 5 se muestra el caso de uso general de los módulos solicitud y validación de horas extras.

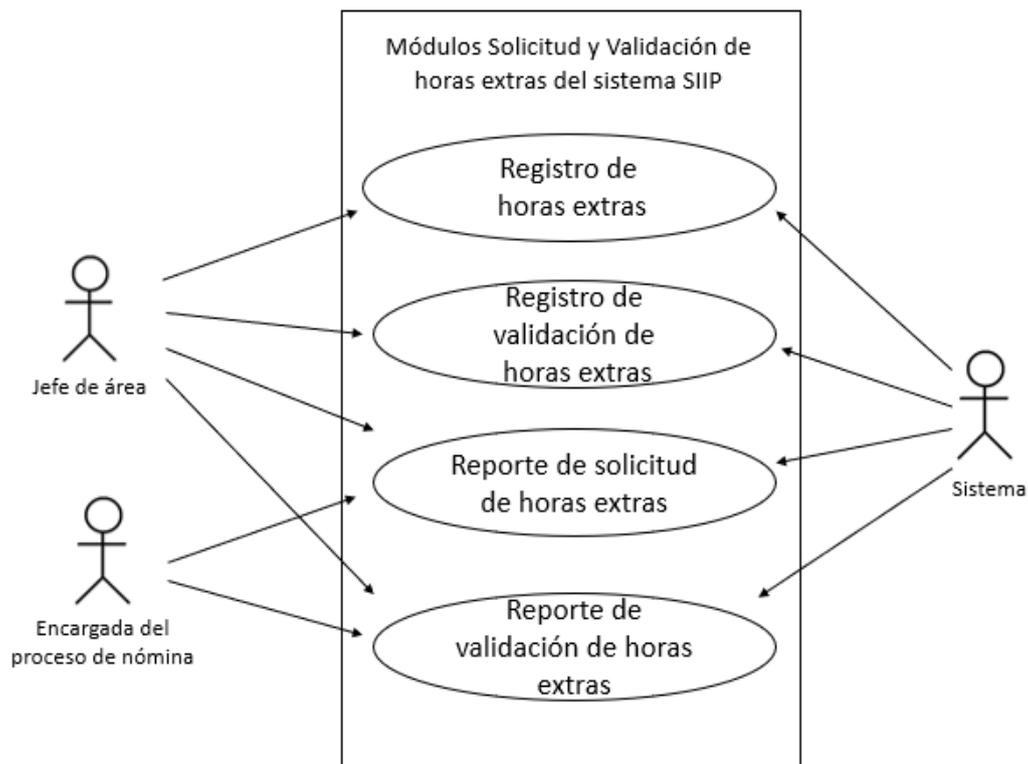


Figura 5. Casos de uso de los procesos de los nuevos módulos

A continuación, se describe el caso de uso registro de horas extras, dicho caso se muestra en la figura 6.



Figura 6. Registro horas extras

Caso de uso: Registro horas extras	
Actores:	Jefe de área, Sistema
Breve descripción:	Permite al actor jefe de área realizar el registro de horas extras y observaciones
Descripción paso a paso:	<ol style="list-style-type: none"> 1. El sistema muestra el número de línea, modulo, área, numero de nómina, nombre completo del empleado, fecha a seleccionar 2. El usuario seleccionará la fecha 3. El usuario tecleará el número de horas extras 4. El sistema validará que el número de horas sea mayor que cero y menor que diecisiete 5. El usuario tecleará las observaciones 6. El sistema validará que existan datos en el campo horas extras, si no existen datos el campo observaciones se limpiará automáticamente 7. El sistema guardará

El caso de uso reporte de solicitud de horas extras se muestra en la figura 7, para posteriormente describir el funcionamiento de dicho caso de uso.



Figura 7. Reporte de solicitud de horas extras

Caso de uso: Reporte de solicitud de horas extras	
Actores:	Jefe de área, encargada del proceso de nómina, Sistema
Breve descripción:	Permite al actor jefe de área, encargada del proceso de nómina y Sistema consultar y generar el reporte del registro de solicitud de horas extras
Descripción paso a paso:	<ol style="list-style-type: none"> 1. Seleccionar la fecha a consultar 2. El sistema generará el reporte automáticamente y mostrará el módulo, área, número de nómina y nombre completo del empleado, horas extras, observaciones y nombre de la persona que elaboró el reporte

A continuación, se describe el caso de uso registro de validación de horas extras, también se muestra en la figura 8

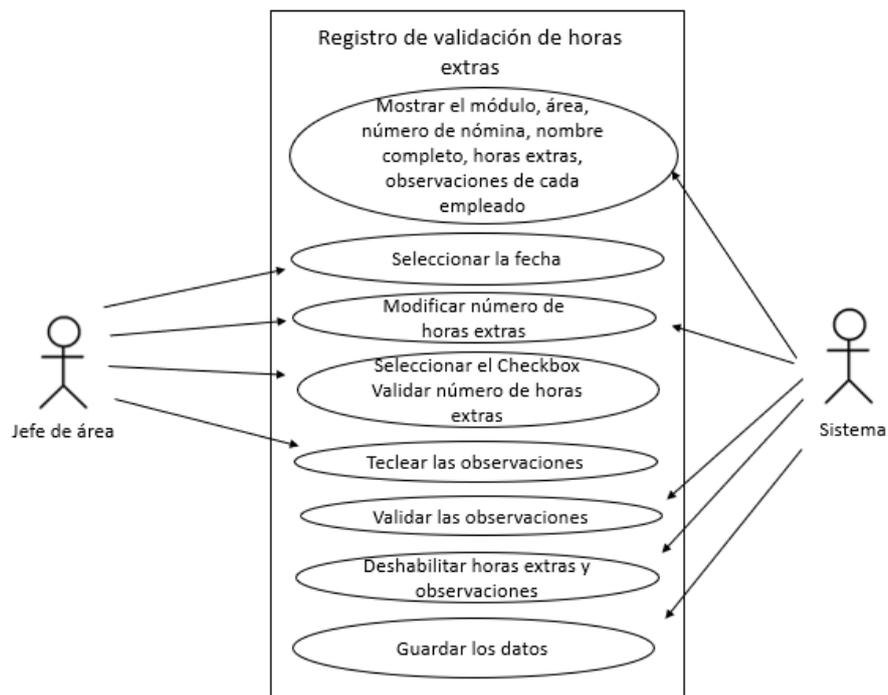


Figura 8. Registro de validación de horas extras

Caso de uso: Registro de validación de horas extras	
Actores:	Jefe de área, Sistema
Breve descripción:	Permite al actor jefe de área validar el número de horas extras realizadas por los trabajadores
Descripción paso a paso:	<ol style="list-style-type: none"> 1. El sistema muestra el número de línea, módulo, área, número de nómina, nombre completo, horas extras, observaciones, fecha a seleccionar 2. El usuario seleccionará la fecha 3. El usuario modificará el número de horas extras si así lo desea 4. El sistema validará que el número de horas sea mayor que cero y menor que diecisiete 5. El usuario tecleará las observaciones 6. El sistema validará que existan datos en el campo horas extras, si no existen datos el campo observaciones se limpiará automáticamente 7. El sistema deshabilitará los campos horas extras y observaciones 8. El sistema guardará las validaciones en la tabla horas extras

Por último, se describe el caso de uso reporte de validación de horas extras como se muestra en la figura 9



Figura 9. Reporte de validación de horas extras

Caso de uso: Reporte de validación de horas extras	
Actores:	Jefe de área, encargada del proceso de nómina, Sistema
Breve descripción:	Permite al actor jefe de área, encargada del proceso de nómina consultar y generar el reporte de validación de horas extras
Descripción paso a paso:	<ol style="list-style-type: none"> 1. Seleccionar la fecha a consultar 2. El sistema generará el reporte automáticamente de los datos validados y mostrara el módulo, área, número de nómina y nombre completo del empleado, horas extras, observaciones y nombre de la persona que elaboró el reporte

Diseño diagramas de entidad relación

La mayor parte de los diagramas de entidad relación ya han sido establecidos como se muestra en la figura 10, sin embargo se creo una nueva tabla llamada horas extras como se muestra en la figura 11 en la cual se muestra que la tabla horas extras está relacionada con la tabla empleados, en la figura también se muestran las tablas operadorModulo y minutos_trab las cuales fueron consultadas para elaborar las consultas y los reportes de cada pantalla, el script generado para la creación de la tabla y consultas será anexado en la siguiente etapa

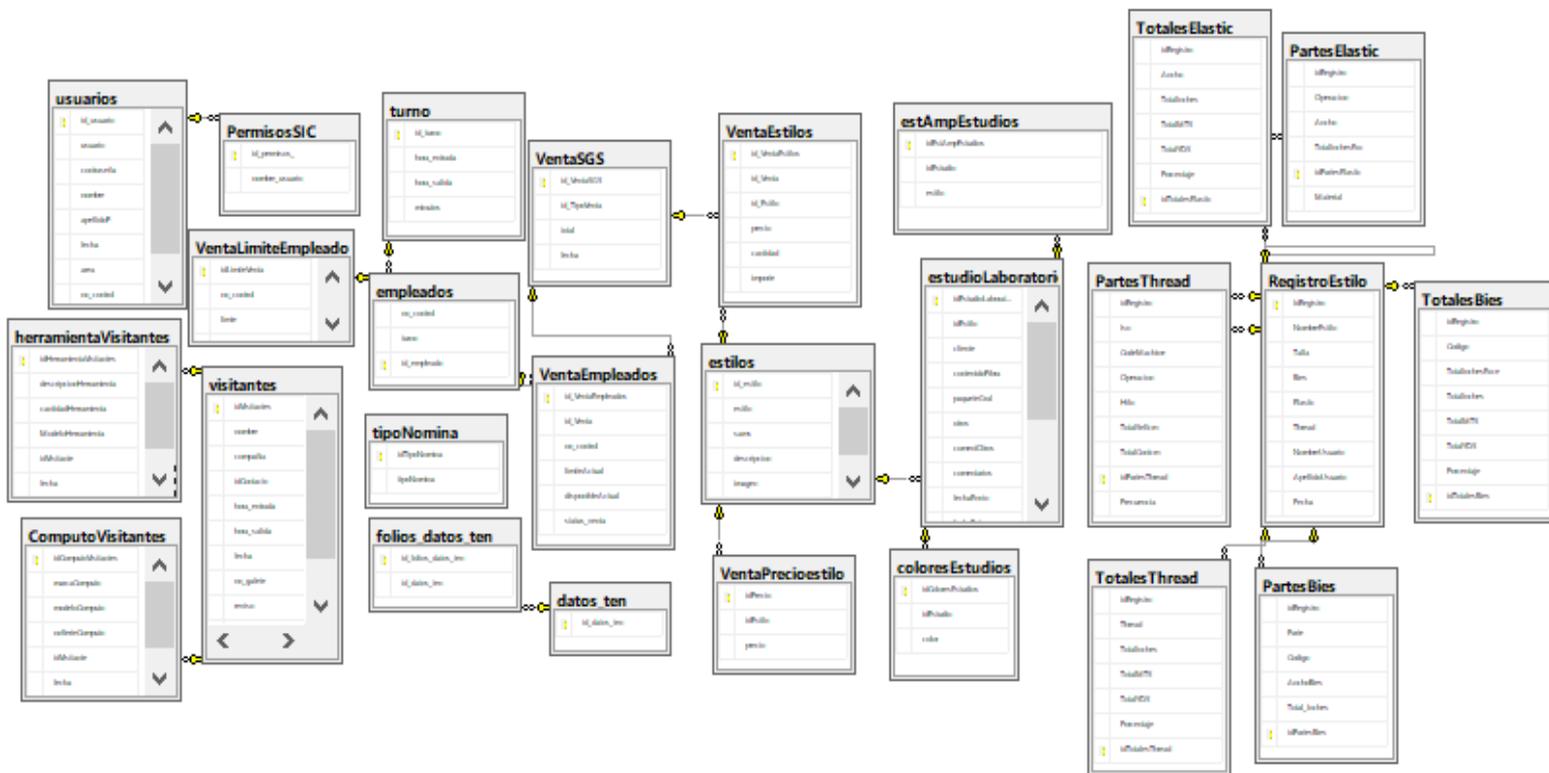


Figura 10. Diagrama de la base de datos SIIP

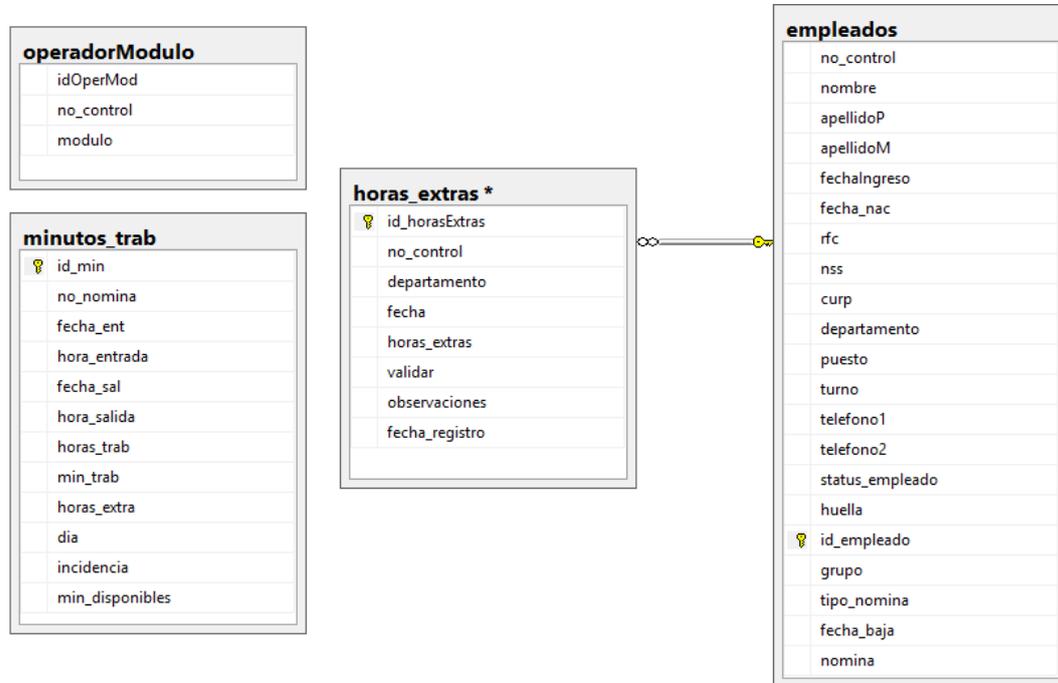


Figura 11. Diagramas de las tablas utilizadas para los módulos solicitud y validación horas extras

3.3 Implementación y prueba de unidades

En la implementación se describirá como se llevaron a cabo cada una de las etapas antes mencionadas en cada uno de los módulos de horas extras.

Tomando en cuenta los requisitos que se recolectaron en la etapa de análisis y diseño se desarrolló la tabla horas extras en la que se registrarán los nuevos datos, el script se muestra en la figura 12 y la tabla con datos registrados se muestra en la figura 13.

```
create table horas_extras(  
  id_horasExtras int primary key identity (1,1),  
  no_control int FOREIGN KEY REFERENCES empleados(no_control) not null,  
  departamento varchar(80) not null,  
  fecha date not null,  
  horas_extras float not null,  
  validar int not null,  
  observaciones varchar(300) not null,  
  fecha_registro date not null  
);
```

Figura 12. Script para generar la tabla horas extras

id_horasExtras	no_control	departamento	fecha	horas_extras	validar	observaciones	fecha_registro
9	153	TEJIDO	2018-10-30	2	0		2018-10-30
10	181	TEJIDO	2018-10-30	2	0		2018-10-30
11	200	TEJIDO	2018-10-30	8	0		2018-10-30
1004	113	TEJIDO	2018-10-31	2	1	Salvador tendra 30 minutos para almorzar	2018-10-31
1005	114	TEJIDO	2018-10-31	3	1	Daniel puede retirarse 10 minutos antes	2018-10-31
1006	116	TEJIDO	2018-10-31	2	1		2018-10-31
1007	119	TEJIDO	2018-10-31	2	1	Jose sera el encargado de supervisar a...	2018-10-31
1008	131	TEJIDO	2018-10-31	3	1	Esta observacion es de muestra	2018-10-31
1009	153	TEJIDO	2018-10-31	4	0		2018-10-31
1010	181	TEJIDO	2018-10-31	2	0		2018-10-31
1011	194	TEJIDO	2018-10-31	3	1	Alejandro supervisa a eliseo	2018-10-31
1012	200	TEJIDO	2018-10-31	4	1	Nueva observacion agregada en panta...	2018-10-31
1013	113	TEJIDO	2018-11-05	5	0		2018-11-05
1014	113	TEJIDO	2018-11-07	2	1		2018-11-07
1015	114	TEJIDO	2018-11-07	3	0		2018-11-07

Figura 13. Tabla horas extras con datos registrados

Posteriormente se desarrollaron las siguientes consultas para el funcionamiento de cada módulo. Las consultas se muestran en las figuras 14, 15, 16 y 17.

```

Update horas_extras set validar=0 , observaciones= 'prueba' where no_control=345
and fecha= '2018-03-10'

Select id_horasExtras, modulo,e.departamento as area,e.no_control as no_nomina,
nombre + ' ' + apellidoP + ' ' + apellidoM
as nombre_completo,fecha,horas_extras
from (( empleados as e inner join operadorModulo om on e.no_control= om.no_control)
inner join horas_extras as he on e.no_control= he.no_control)

Select e.no_control, e.nombre, e.apellidoP, e.apellidoM, he.observaciones
from empleados as e inner join horas_extras as he on e.no_control= he.no_control
where validar= 1

Select e.no_control, e.nombre, e.apellidoP, e.apellidoM,he.horas_extras, he.observaciones
from empleados as e inner join horas_extras as he on e.no_control= he.no_control

Select e.no_control, e.nombre, e.apellidoP, e.apellidoM,he.horas_extras, he.observaciones
from empleados as e inner join horas_extras as he on e.no_control= he.no_control
--consulta de mostrardgHE
Select modulo,departamento as area,empleados.no_control as no_nomina,nombre + ' '
+ apellidoP + ' ' + apellidoM as nombre_completo
from empleados left join operadorModulo on empleados.no_control= operadorModulo.no_control
where departamento= 'TEJIDO' and status empleado=1

```

Figura 14. Consultas en SQL Server para el funcionamiento de cada módulo

```

Select e.no_control, e.nombre, e.apellidoP, e.apellidoM,he.horas_extras, he.observaciones
from empleados as e inner join horas_extras as he on e.no_control= he.no_control
where fecha = '2018-10-11'
and e.departamento='CONFECCIÓN' and status_empleado=1

-----nueva consulta para el crystal report solicitud horas extras
select e.no_control, op.modulo,e.nombre + ' ' + e.apellidoP + ' ' + e.apellidoM as nombre,
horas_extras,observaciones
from empleados as e left join operadorModulo as op on e.no_control= op.no_control
inner join horas_extras as he on e.no_control= he.no_control
where fecha = '2018-10-11'
and e.departamento='CONFECCIÓN' and status_empleado=1

-----

-----consulta para el crystal report solicitud horas extras
--Select e.no_control,op.modulo, e.nombre, e.apellidoP, e.apellidoM,he.horas_extras,
--he.observaciones
--from empleados as e inner join horas_extras as he on e.no_control= he.no_control
--where fecha = '{?@fecha}'
--and e.departamento='{?@departamento}'
-----
Select om.modulo, he.no_control,he.departamento,nombre + ' ' + apellidoP + ' ' + apellidoM
as nombre_completo,
he.horas_extras, he.validar, he.observaciones from ((horas_extras as he left join operadorModulo
as om on he.no_control=om.no_control)
inner join empleados as e on he.no_control= e.no_control) where he.departamento='CONFECCIÓN'
and fecha='2018-10-10'and status empleado=1 and validar=1

```

Figura 15. Consultas en SQL Server para el funcionamiento de cada módulo

```

--consulta de crystal reports validar horas extras
Select e.no_control, e.nombre, e.apellidoP, e.apellidoM,he.horas_extras, he.observaciones
from empleados as e inner join horas_extras as he on e.no_control= he.no_control
where validar= 1 and fecha = '{?@fecha}' and e.departamento='{?@departamento}'
-----

--consulta para crystal reports validar horas extras ---con parametros de cryatal reoport
Select om.modulo, he.no_control,he.departamento,nombre + ' ' + apellidoP + ' ' + apellidoM
as nombre_completo,
he.horas_extras, he.validar, he.observaciones
from ((horas_extras as he left join operadorModulo as om on he.no_control=om.no_control)
inner join empleados as e on he.no_control= e.no_control)
where he.departamento='{?@departamento}' and fecha='{?@fecha}' and status_empleado=1
and validar= 1
-----

--consulta para crystal reports validar horas extras ---con parametros SQL_SERVER
Select om.modulo, he.no_control,he.departamento,nombre + ' ' + apellidoP + ' ' + apellidoM
as nombre_completo,
he.horas_extras, he.validar, he.observaciones
from ((horas_extras as he left join operadorModulo as om on he.no_control=om.no_control)
inner join empleados as e on he.no_control= e.no_control)
where he.departamento='CONFECCIÓN' and fecha='2018-10-16' and status_empleado=1
and validar= 1

SELECT MAX(fecha) FROM horas_extras

```

Figura 16. Consultas en SQL Server para el funcionamiento de cada módulo

```

SELECT MAX(fecha) FROM horas_extras where departamento= 'TEÑIDO'

Select * from horas_extras where fecha='2018-10-18' and no_control=1115 and horas_extras=8
and validar=1 and observaciones='hola'
select * from departamentos

-- consulta que uso para cargar el datagrid solicitar he
Select modulo,departamento as area,empleados.no_control as no_nomina,nombre + ' ' +
apellidoP + ' ' + apellidoM
as nombre_completo from empleados left join operadorModulo
on empleados.no_control= operadorModulo.no_control
where departamento='ALMACÉN P/T' and status_empleado=1

Select modulo,T0.departamento as area,T0.no_control
as no_nomina,T0.nombre + ' ' + T0.apellidoP + ' ' + T0.apellidoM as nombre_completo,horas_extras,observaciones,
fecha
from empleados as T0
left join operadorModulo on T0.no_control= operadorModulo.no_control
left join horas_extras as T2 on T0.no_control=T2.no_control
where T0.departamento='ALMACÉN P/T' and T0.status_empleado=1 and (fecha='2018-10-18' or fecha is null)

Update horas_extras set validar=0 where no_control=8097 and fecha= '2018-10-22'

```

Figura 17. Consultas en SQL Server para el funcionamiento de cada módulo

Para ingresar al sistema es necesario introducir el usuario y contraseña en el sistema SIIP, solo los empleados que tienen permiso podrán visualizar y hacer uso de los módulos de horas extras, cabe mencionar que las pantallas que se muestran a continuación tienen el mismo diseño de interfaz que se había mencionado en la etapa de diseño

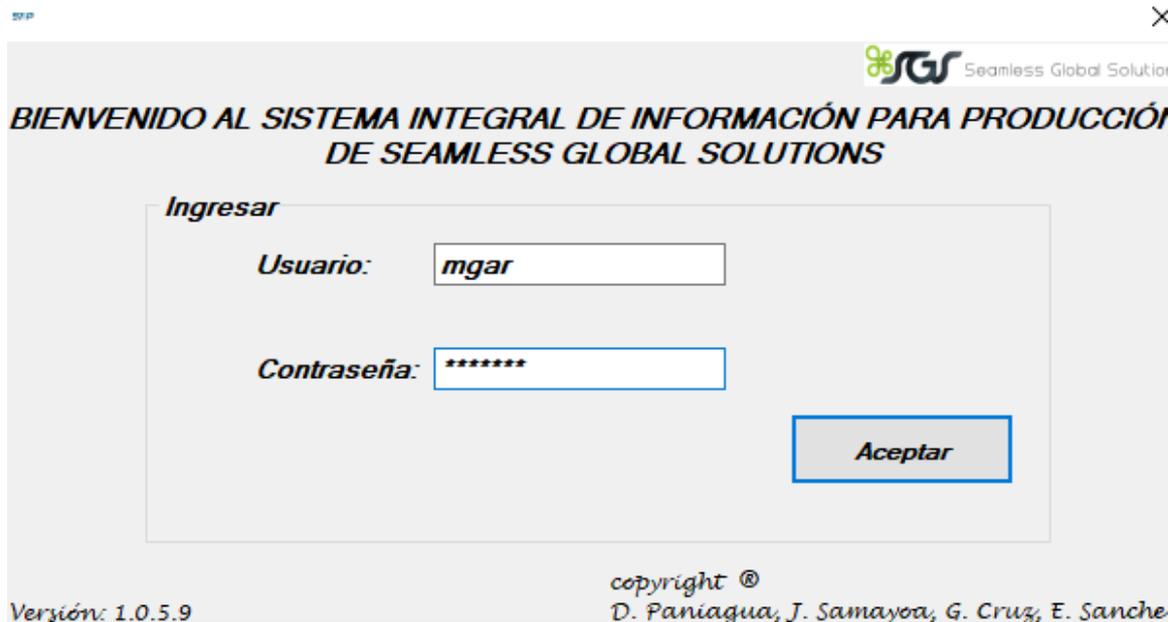


Figura 18. Inicio de sesión en el sistema SIIP

Al iniciar sesión se mostrará un menú seleccionamos recursos humanos seguido seleccionamos incidencias y posteriormente solicitud de horas extras enseguida se nos mostrará la siguiente pantalla como se muestra en la figura 19

El módulo solicitud de horas extras consiste en registrar el número de horas extras solicitadas por el empleado, registrar una observación si es necesaria, guardar los datos e imprimir un reporte en crystal report

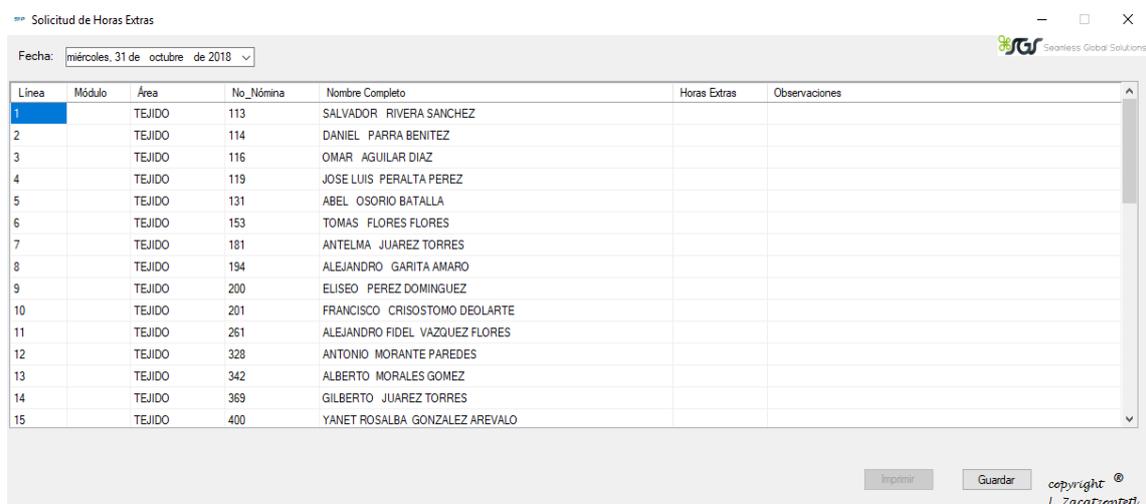


Figura 19. Interfaz solicitud de horas extras.

La pantalla reporte de solicitud de horas extras consiste en hacer una consulta ala tabla horas extras para mostrar los datos de los empleados a los que se les asignó tiempo extra como se muestra en la figura 20

N°	N° Empleado	Módulo	Nombre	Horas Extras	Observaciones	Firma
1	113		SALVADOR RIVERA SANCHEZ	2	Salvador tendra 30 minutos para almorzar	
2	114		DANIEL PARRA BENITEZ	3	Daniel puede retirarse 10 minutos antes	
3	116		OMAR AGUILAR DIAZ	2		
4	119		JOSE LUIS PERALTA PEREZ	2	Jose sera el encargado de supervisar a omar,abel y tomas.	
5	131		ABEL OSORIO BATALLA	3		
6	153		TOMAS FLORES FLORES	4		
7	181		ANTELMA JUAREZ TORRES	2		
8	194		ALEJANDRO GARITA AMARO	3	Alejandro supervisa a eliseo	
9	200		ELISEO PEREZ DOMINGUEZ	4		

Figura 20. Crystal Report formato de solicitud de tiempo extra

El módulo validación de horas extras consiste en validar las horas extras de los empleados, así como también permite generar un reporte en crystal report con los empleados que han sido figura 21

Línea	Módulo	Área	No_Nómina	Nombre Completo	Horas Extras	Validar	Observaciones
1		TEJIDO	113	SALVADOR RIVERA SANCHEZ	2	<input type="checkbox"/>	Salvador tendra 30 minutos para almorzar
2		TEJIDO	114	DANIEL PARRA BENITEZ	3	<input type="checkbox"/>	Daniel puede retirarse 10 minutos antes
3		TEJIDO	116	OMAR AGUILAR DIAZ	2	<input type="checkbox"/>	
4		TEJIDO	119	JOSE LUIS PERALTA PEREZ	2	<input type="checkbox"/>	Jose sera el encargado de supervisar a omar,abel y tomas.
5		TEJIDO	131	ABEL OSORIO BATALLA	3	<input type="checkbox"/>	
6		TEJIDO	153	TOMAS FLORES FLORES	4	<input type="checkbox"/>	
7		TEJIDO	181	ANTELMA JUAREZ TORRES	2	<input type="checkbox"/>	
8		TEJIDO	194	ALEJANDRO GARITA AMARO	3	<input type="checkbox"/>	Alejandro supervisa a eliseo
9		TEJIDO	200	ELISEO PEREZ DOMINGUEZ	4	<input type="checkbox"/>	

Figura. 21 Interfaz validación de horas extras.

La pantalla reporte de validación de tiempo extra consiste en hacer una consulta a la tabla horas extras para mostrar los datos de los empleados que realizaron tiempo extra como se muestra en la figura 22



Figura 22. Crystal Report formato de validación de tiempo extra

En el anexo 1 se muestra parte del código que da funcionamiento los módulos solicitud y validación de horas extras

3.4 Integración y prueba del sistema

El módulo solicitud y validación horas extras ha sido integrado al sistema SIIP, anteriormente las variables área y jefe de área eran estáticos durante la integración estas variables se adaptaron para obtener el jefe de área y el área de la persona que inicio sesión en el SIIP.

También se realizaron las pruebas de consistencia de las tablas nuevas con la base de datos ya existente como se muestra en la figura 23

Linea	Módulo	Área	No_Nómina	Nombre Completo	Horas Extras	Validar	Observaciones
1		TEJIDO	113	SALVADOR RIVERA SANCHEZ	2	<input checked="" type="checkbox"/>	Salvador tendra 30 minutos para almorzar
2		TEJIDO	114	DANIEL PARRA BENITEZ	3	<input checked="" type="checkbox"/>	Daniel puede retirarse 10 minutos antes
3		TEJIDO	116	OMAR AGUILAR DIAZ	2	<input checked="" type="checkbox"/>	
4		TEJIDO	119	JOSE LUIS PERALTA PEREZ	2	<input checked="" type="checkbox"/>	Jose sera el encargado de supervisar a omar,abel y tomas.
5		TEJIDO	131	ABEL OSORIO BATALLA	3	<input checked="" type="checkbox"/>	Esta observacion es de muestra
6		TEJIDO	153	TOMAS FLORES FLORES	4	<input type="checkbox"/>	
7		TEJIDO	181	ANTELMA JUAREZ TORRES	2	<input type="checkbox"/>	
8		TEJIDO	194	ALEJANDRO GARITA AMARO	3	<input checked="" type="checkbox"/>	Alejandro supervisa a elseo
9		TEJIDO	200	ELISEO PEREZ DOMINGUEZ	4	<input checked="" type="checkbox"/>	Nueva observacion agregada en pantalla validar horas extras

	id_horasExtras	no_control	departamento	fecha	horas_extras	validar	observaciones	fecha_registro
1	1004	113	TEJIDO	2018-10-31	2	1	Salvador tendra 30 minutos para almorzar	2018-10-31
2	1005	114	TEJIDO	2018-10-31	3	1	Daniel puede retirarse 10 minutos antes	2018-10-31
3	1006	116	TEJIDO	2018-10-31	2	1		2018-10-31
4	1007	119	TEJIDO	2018-10-31	2	1	Jose sera el encargado de supervisar a omar,abel ...	2018-10-31
5	1008	131	TEJIDO	2018-10-31	3	1	Esta observacion es de muestra	2018-10-31
6	1009	153	TEJIDO	2018-10-31	4	0		2018-10-31
7	1010	181	TEJIDO	2018-10-31	2	0		2018-10-31
8	1011	194	TEJIDO	2018-10-31	3	1	Alejandro supervisa a elseo	2018-10-31
9	1012	200	TEJIDO	2018-10-31	4	1	Nueva observacion agregada en pantalla validar h...	2018-10-31

Figura 23. Prueba de consistencia en la tabla horas extras

3.5 Funcionamiento y mantenimiento

Se instaló en la computadora de la encargada de nóminas y en la computadora de un jefe de área que registra frecuentemente horas extras, la nueva versión del SIIP con los nuevos módulos, para adaptarse a la nueva automatización del proceso de horas extras, el sistema estará sujeto a cambios si se detectan mejoras o nuevos requerimientos en un lapso de un mes. A continuación, se muestran imágenes del funcionamiento del sistema con su respectiva descripción.

La pantalla de solicitud de horas extras carga el número de línea, modulo, área, número de nómina y nombre completo con el botón de imprimir deshabilitado como se muestra en la figura 24, el área se carga de acuerdo con el departamento al que pertenece la persona que inicio sesión, para llenar el campo de horas extras lo primero que se debe de hacer es verificar que la fecha deseada este seleccionada, de no ser así se debe seleccionar la fecha deseada, es importante mencionar que la fecha no puede ser una fecha pasada, si la fecha es pasada el sistema no permitirá cargarla y mostrara la fecha actual, así como también mostrara un mensaje de error como se muestra en la figura 25.

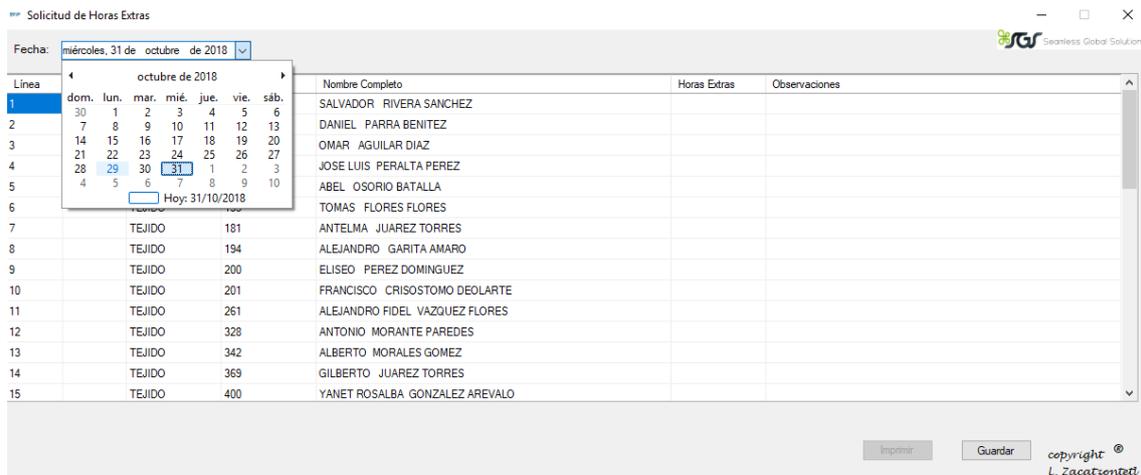


Figura 24. Datos de la interfaz solicitud de horas extras.



Figura 25. Error selecciona una fecha invalida

El campo horas extras sólo acepta números enteros o con decimal (0-16) si el usuario introduce un número mayor que 16 se muestra el error número fuera del rango, en la figura 26 el usuario intento introducir el número de horas extras con letra enseguida se muestra el error de validación con la línea correspondiente.

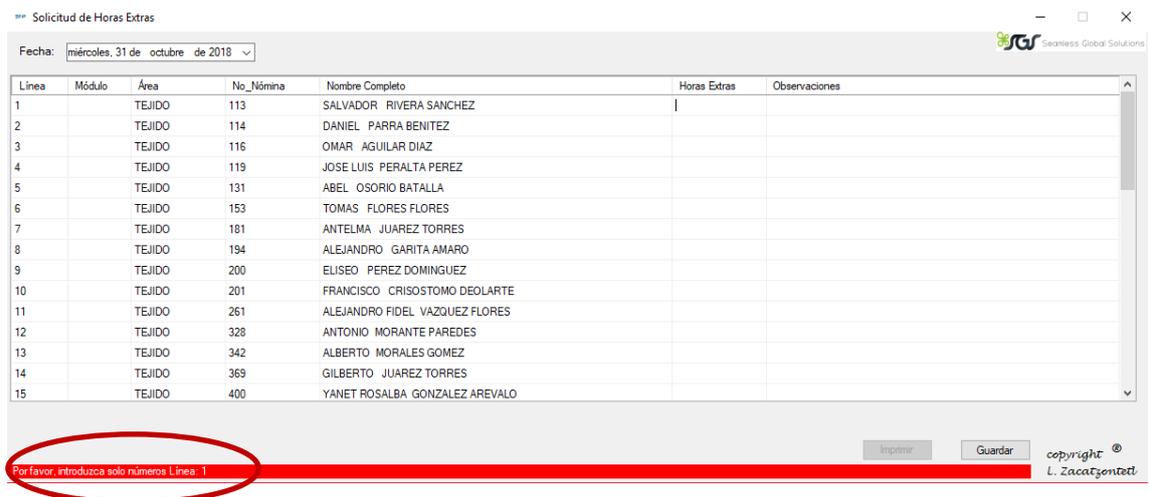


Figura 26. Se muestra el error de validación del campo horas extras.

En la figura 27 se presionó el botón guardar sin haber introducido algún dato en las celdas debido a esto se muestra el error no hay datos que guardar como se muestra en la figura 27, en la figura 28 se muestra la manera correcta de cómo deben llenarse los campos horas extra y observaciones, cabe mencionar que en el campo observaciones se puede o no agregar comentarios, en la figura 28 el botón guardar no ha sido presionado, cuando el botón guardar es presionado la pantalla muestra un mensaje de éxito que indica el número de empleados registrados para tiempo extras y se habilita el botón imprimir como se muestra en la figura 29.

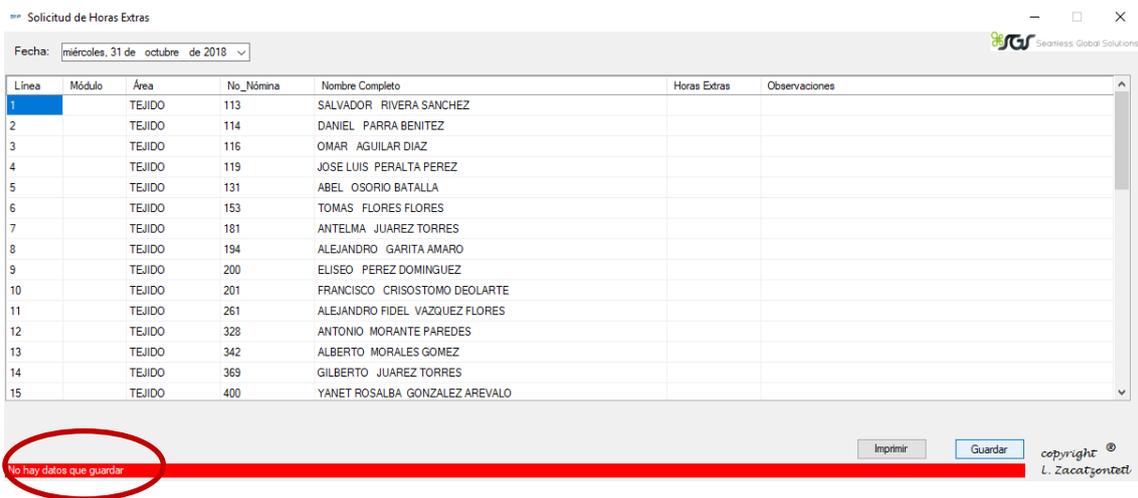


Figura 27. Se muestra el error no hay datos que guardar.

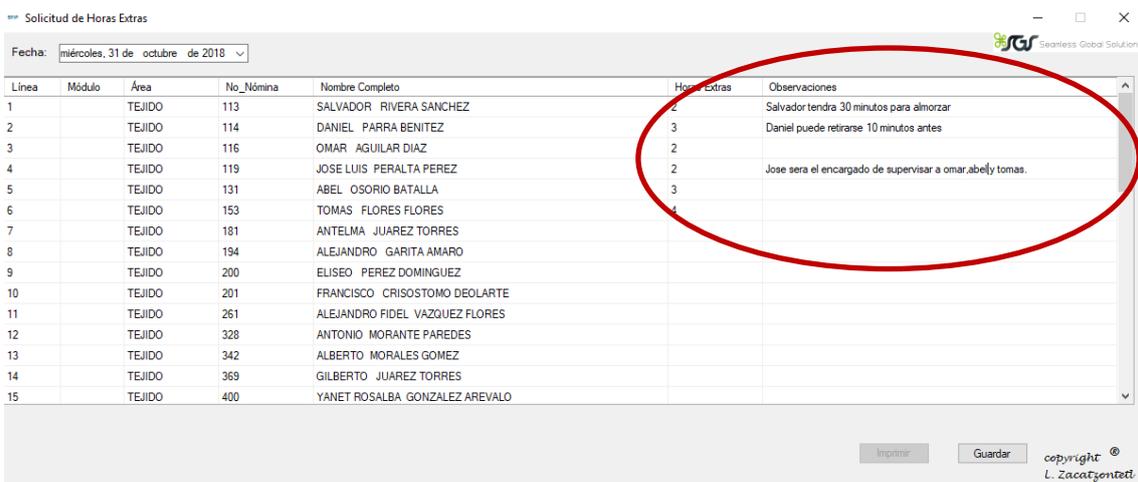


Figura 28. Datos introducidos correctamente en el campo horas extras y observaciones

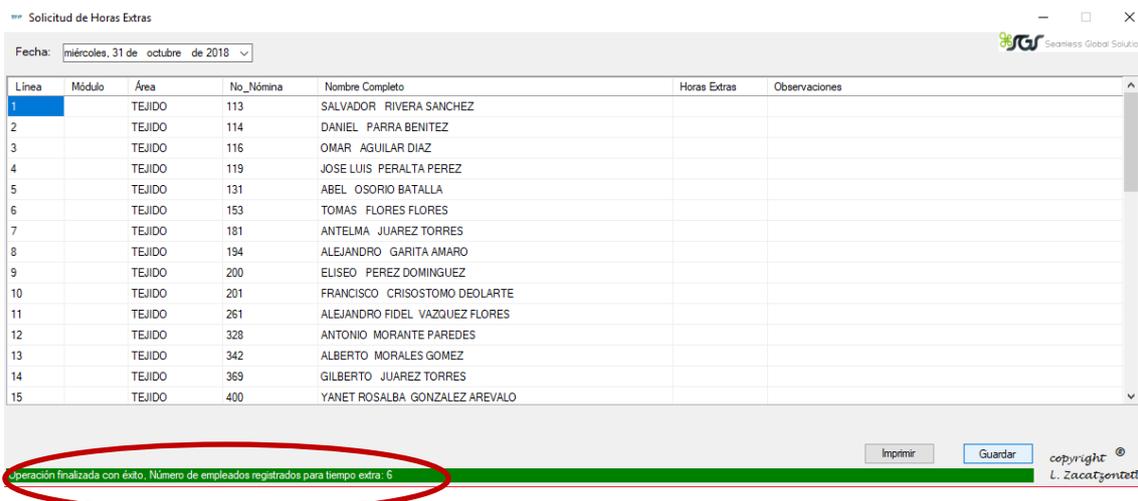


Figura 29. Se muestra el mensaje de éxito y el número de empleados registrados para tiempo extra

En la figura 30 se muestra como un usuario intento introducir nuevamente una cantidad en el campo horas extras, el campo horas extras valida si ya existe un dato con la misma fecha y numero de nómina, si ya existe un dato nos muestra

el número de línea y número de nómina que ya tiene horas extras asignadas, enseguida de mostrar el error se limpian las celdas horas extras y observaciones automáticamente

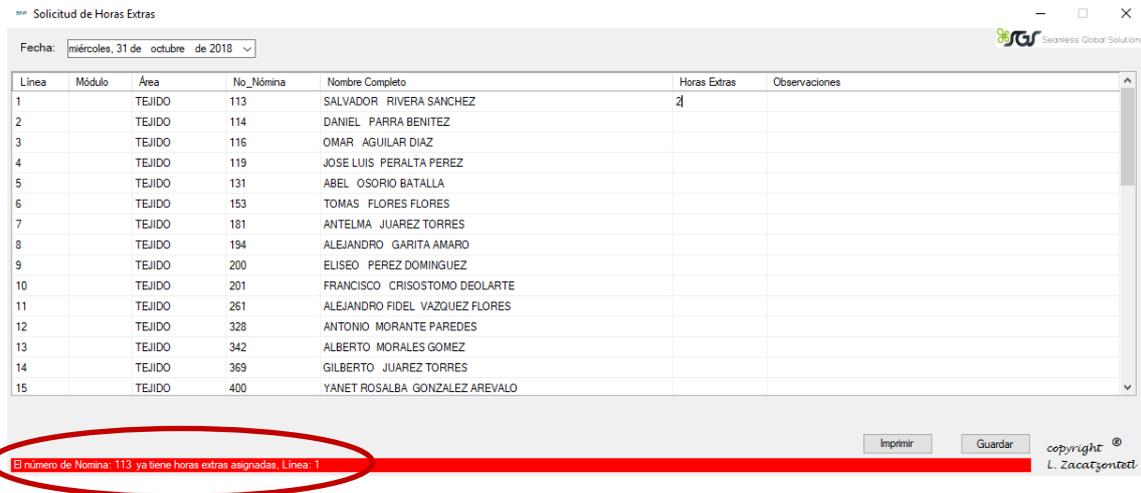


Figura 30. Se muestra el error ya tiene horas extras asignadas

En la interfaz solicitud horas extras se pueden introducir más datos a los empleados que no tengan asignadas horas extras con anterioridad como se muestra en la figura 31, la figura 32 muestra el mensaje de éxito.

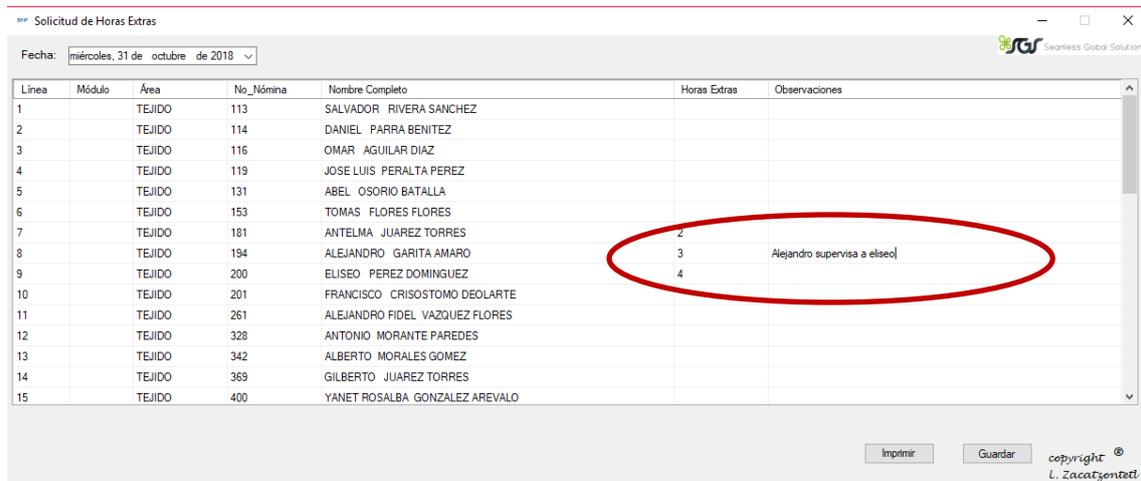


Figura 31. Nuevos datos introducidos correctamente en el campo horas extras y observaciones



Figura 32. Se muestra el mensaje de éxito y un número de empleados registrados.

El campo de observaciones no puede ser guardado si no existen datos en el campo horas extras de la misma línea como se muestra en la figura 33, nos mostrará un error como el de la figura 34

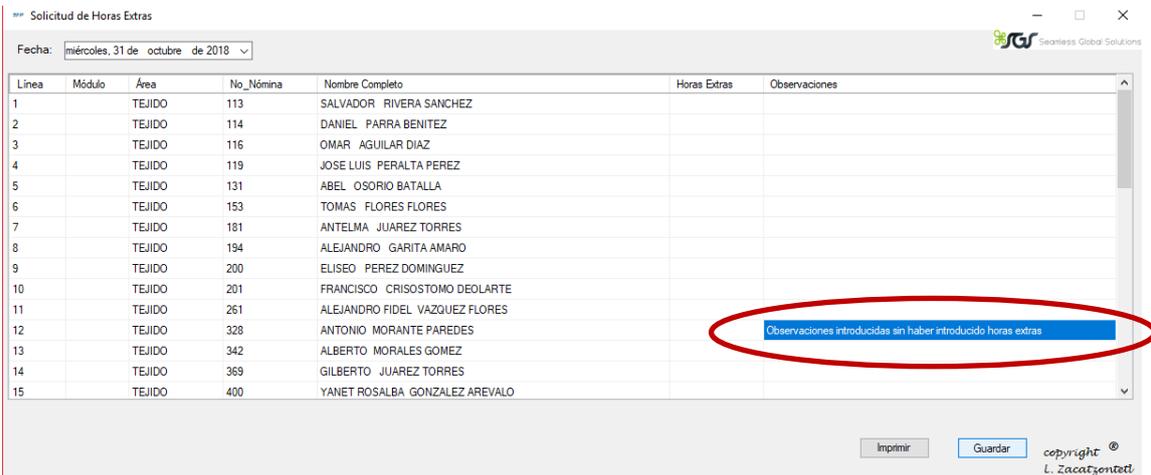


Figura 33. Observaciones sin horas extras



Figura 34. Se muestra el error no hay datos que guardar debido a que observaciones no contaba con horas extras

El botón imprimir nos genera un reporte en crystal report con los datos de los empleados que tienen horas extras, al presionarlo nos mostrará una barra de progreso que nos indica el nivel de carga del crystal report y un mensaje de información (cargando) como se muestra en la figura 35.

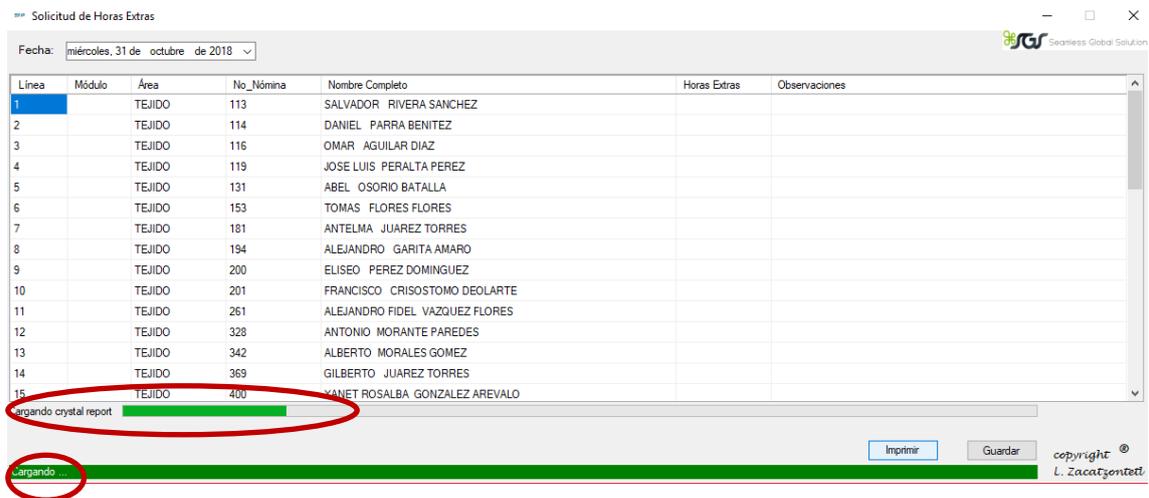


Figura 35. Se muestra el mensaje de cargando y la barra de progreso del crystal report

Posteriormente el crystal report se muestra maximizado en otra ventana con los datos de los empleados que tienen horas extras como se muestra en la figura 36



Figura 36. Crystal Report formato de solicitud de tiempo extra

Al cargar la pantalla validación de horas extras enseguida se nos mostrará los datos de la última fecha que registramos horas extras, también se mostrará el botón imprimir deshabilitó como se muestra en la figura 37.

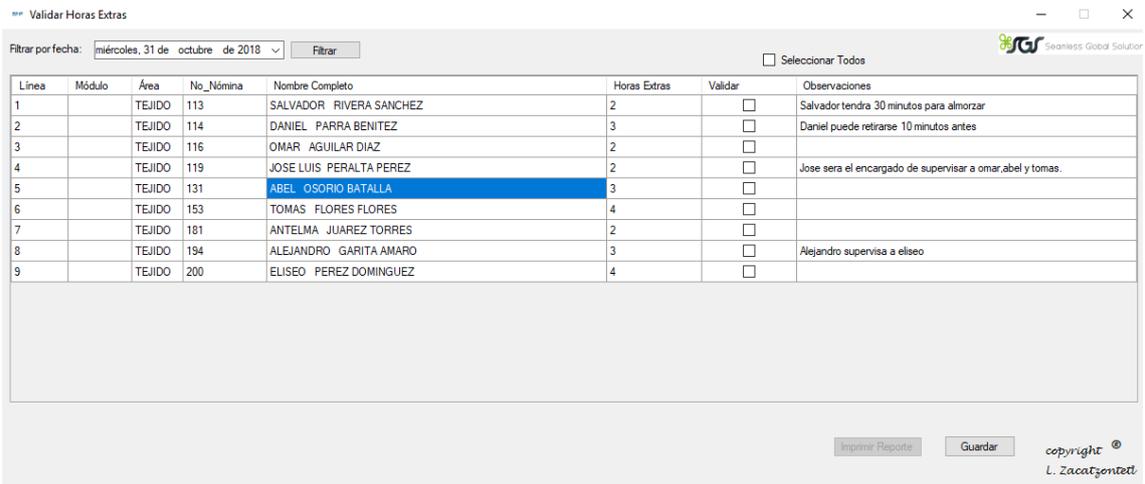


Figura 37. Interfaz validación de horas extras.

En la interfaz validación horas extras se pueden cargar datos con fechas pasadas seleccionando la fecha que se deseé consultar seguido de presionar el botón filtrar como se muestra en la figura 38, si la fecha seleccionada no contiene datos se mostrará un mensaje de informacion como se muestra en la figura 39

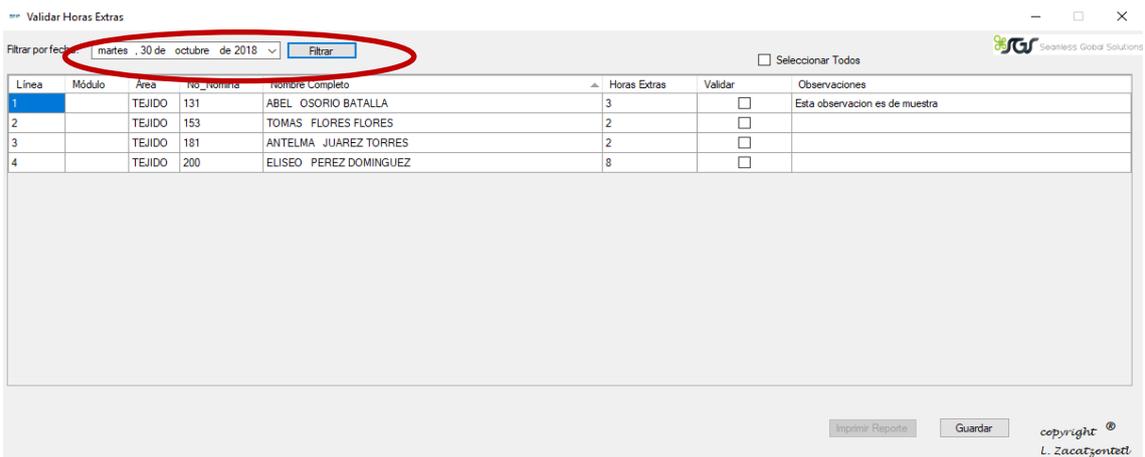


Figura 38. Datos cargados con una fecha seleccionada pasada

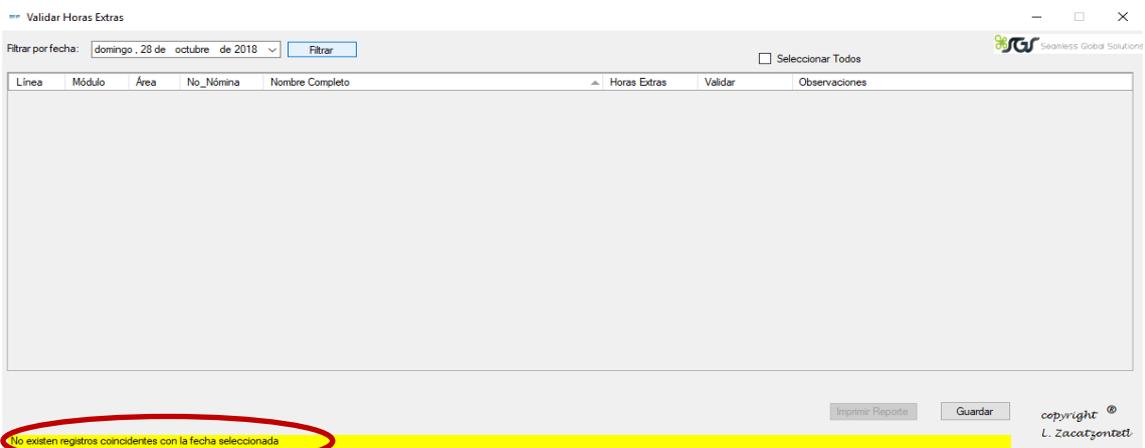


Figura 39. Se muestra el mensaje de no existen registros coincidentes

Al presionar el botón guardar nos muestra un mensaje de confirmación que contiene tres opciones Si, No y Cancelar como se muestra en la figura 40, si elegimos la opción Si pero no existen datos nos mostrara el mensaje no existen datos que guardar como se muestra en la figura 41, si elegimos la opción No se limpiaran las celdas que contenían nuevos datos, si elegimos la opción Cancel el mensaje se cerrará y nos permitirá seguir editando sin borrar ningún dato como se muestra en la imagen 42.

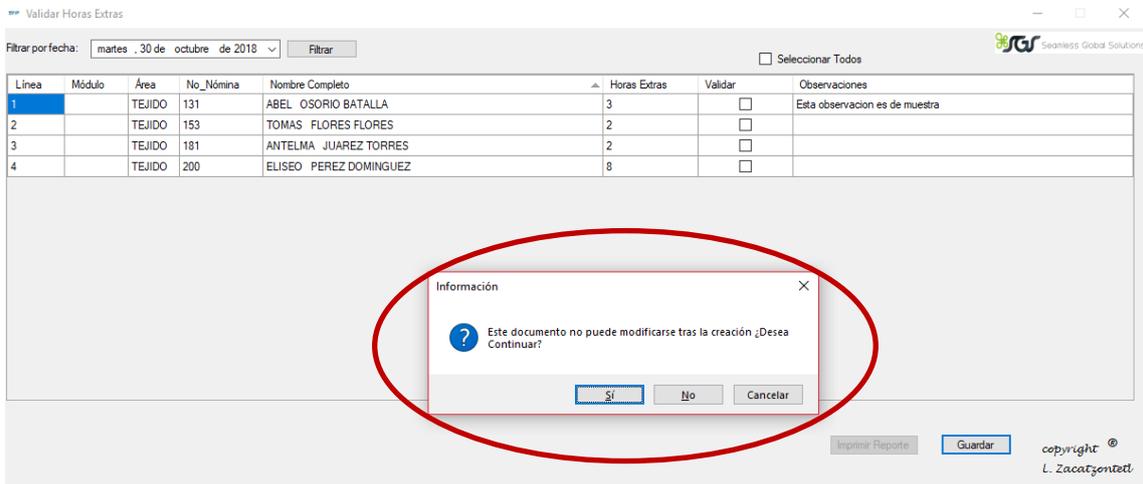


Figura 40. Se muestra el mensaje de confirmación

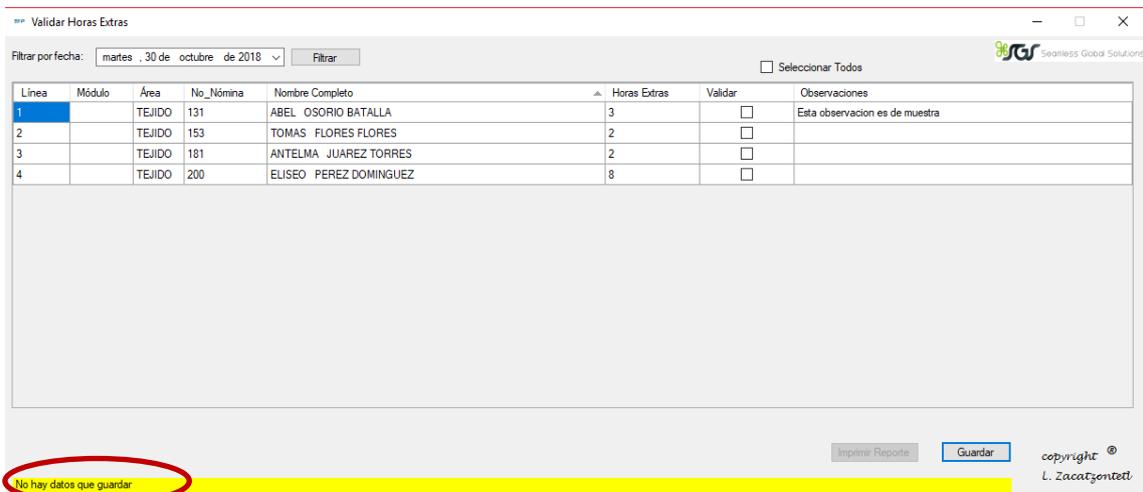


Figura 41. Se muestra el mensaje no existen datos que guardar

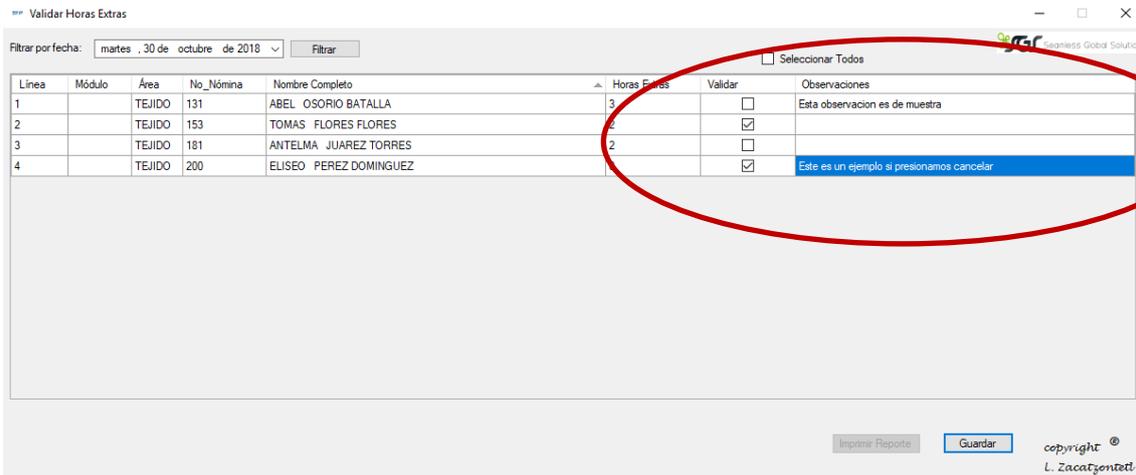


Figura 42. Datos introducidos antes y después de presionar la opción Cancelar

En la pantalla validación de horas extras tenemos la opción de seleccionar todos los campos de validación con el Checkbox Seleccionar todos como se muestra en la figura 43, con este componente podemos seleccionar todos y deseleccionar los que deseemos como se muestra en la figura 44

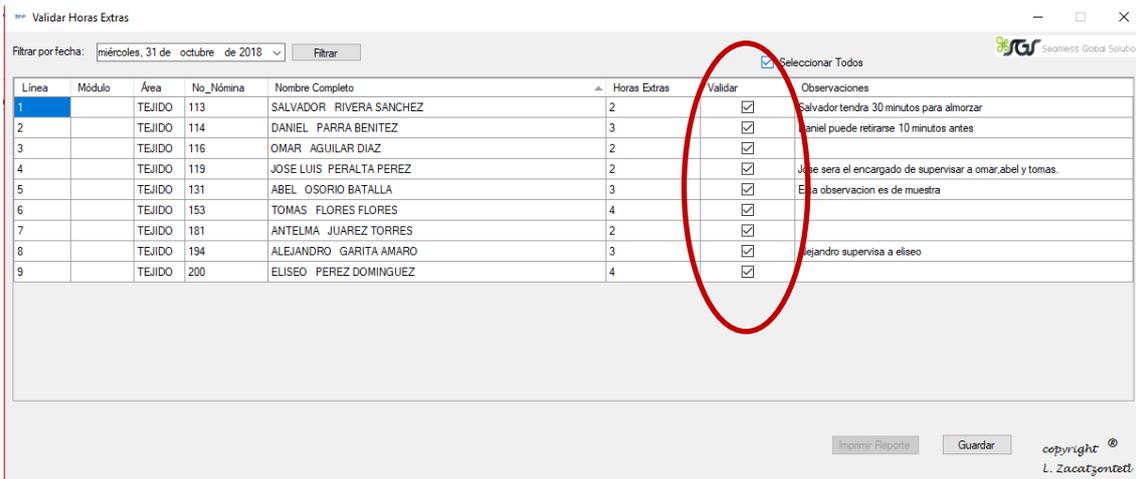


Figura 43. Todos los campos de validación seleccionados con el Checkbox seleccionar todos

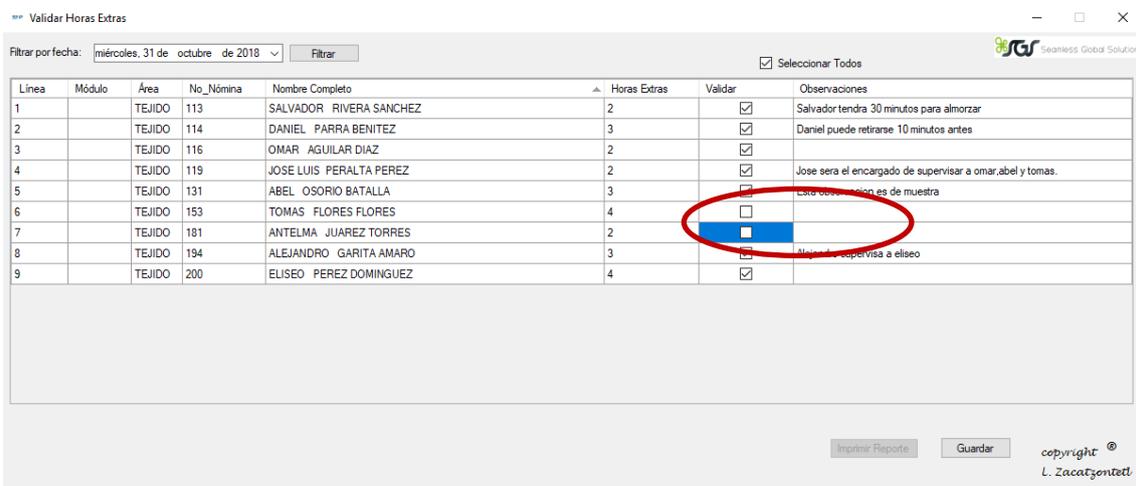


Figura 44. Campos de validación deseleccionados manualmente

En el campo horas extra de la pantalla validación de horas extras se puede modificar la cantidad de horas, en este campo se valida que la cantidad introducida no esté fuera del rango 0-16 como se muestra en la figura 45, en este campo también se valida que no pueda insertar letras como se muestra el error en la figura 46, por último se valida que el campo no quede vacío, si en el campo no se introduce una cantidad la validación no permitirá que el cursor salga de la casilla como se muestra en la figura 47

The screenshot shows a web application window titled "Validar Horas Extras". At the top, there is a date filter set to "miércoles, 31 de octubre de 2018" and a "Filtrar" button. Below the filter is a table with 9 rows. The columns are: Línea, Módulo, Área, No. Nómina, Nombre Completo, Horas Extras, Validar, and Observaciones. The "Horas Extras" column for the first row (Línea 1) contains the value "0", which is circled in red. Below the table, there is a red error message: "Error: No puede insertar un número fuera del siguiente rango 0-16, Línea 1". At the bottom right, there are buttons for "Imprimir Reporte" and "Guardar", and a copyright notice for "L. Zacatzontel".

Línea	Módulo	Área	No. Nómina	Nombre Completo	Horas Extras	Validar	Observaciones
1		TEJIDO	113	SALVADOR RIVERA SANCHEZ	0	<input type="checkbox"/>	Salvador tendrá 30 minutos para almorzar
2		TEJIDO	114	DANIEL PARRA BENITEZ	3	<input type="checkbox"/>	Daniel puede retrarse 10 minutos antes
3		TEJIDO	116	OMAR AGUILAR DIAZ	2	<input type="checkbox"/>	
4		TEJIDO	119	JOSE LUIS PERALTA PEREZ	2	<input type="checkbox"/>	Jose sera el encargado de supervisar a omar,abel y tomas.
5		TEJIDO	131	ABEL OSORIO BATALLA	3	<input type="checkbox"/>	Esta observacion es de muestra
6		TEJIDO	153	TOMAS FLORES FLORES	4	<input type="checkbox"/>	
7		TEJIDO	181	ANTELMA JUAREZ TORRES	2	<input type="checkbox"/>	
8		TEJIDO	194	ALEJANDRO GARITA AMARO	3	<input type="checkbox"/>	Alejandro supervisa a eliseo
9		TEJIDO	200	ELISEO PEREZ DOMINGUEZ	4	<input type="checkbox"/>	Nueva observacion agregada en pantalla validar horas extras

Figura 45. Se muestra el error fuera de rango al introducir un cero en horas extras

The screenshot shows the same web application window as Figure 45. The "Horas Extras" field for the first row (Línea 1) now contains the value "|", which is circled in red. Below the table, there is a red error message: "Por favor, introduzca solo números, Línea 3". The rest of the interface, including the date filter, table, and buttons, remains the same.

Línea	Módulo	Área	No. Nómina	Nombre Completo	Horas Extras	Validar	Observaciones
1		TEJIDO	113	SALVADOR RIVERA SANCHEZ	2	<input type="checkbox"/>	Salvador tendrá 30 minutos para almorzar
2		TEJIDO	114	DANIEL PARRA BENITEZ	3	<input type="checkbox"/>	Daniel puede retrarse 10 minutos antes
3		TEJIDO	116	OMAR AGUILAR DIAZ		<input type="checkbox"/>	
4		TEJIDO	119	JOSE LUIS PERALTA PEREZ	2	<input type="checkbox"/>	Jose sera el encargado de supervisar a omar,abel y tomas.
5		TEJIDO	131	ABEL OSORIO BATALLA	3	<input type="checkbox"/>	Esta observacion es de muestra
6		TEJIDO	153	TOMAS FLORES FLORES	4	<input type="checkbox"/>	
7		TEJIDO	181	ANTELMA JUAREZ TORRES	2	<input type="checkbox"/>	
8		TEJIDO	194	ALEJANDRO GARITA AMARO	3	<input type="checkbox"/>	Alejandro supervisa a eliseo
9		TEJIDO	200	ELISEO PEREZ DOMINGUEZ	4	<input type="checkbox"/>	Nueva observacion agregada en pantalla validar horas extras

Figura 46. Se muestra el error solo números.

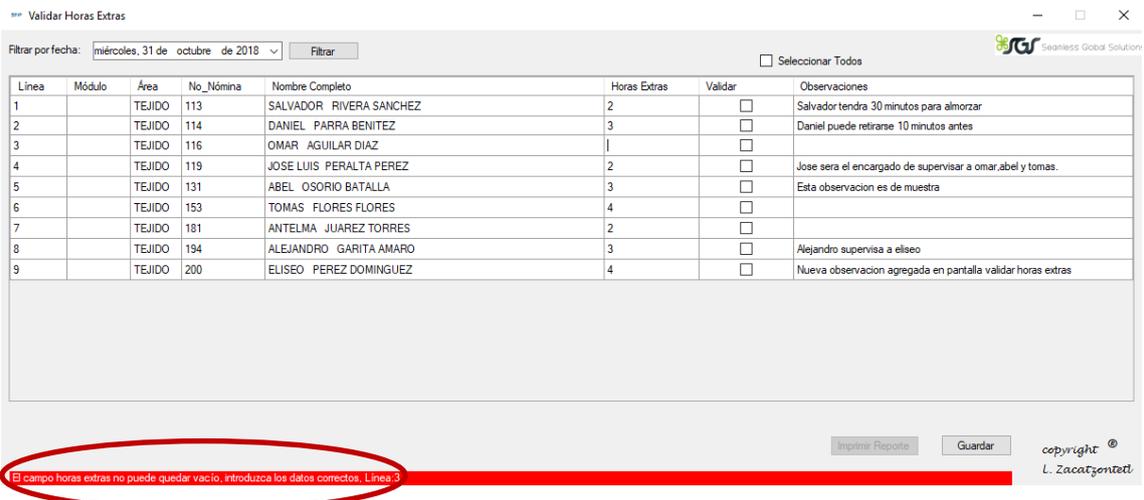


Figura 47. Se muestra el error el campo horas extras no puede quedar vacío

Al presionar el botón guardar se modifican las horas extras si es el caso y se guardan las observaciones, estas modificaciones se realizan siempre y cuando el campo validar este seleccionado, si el campo de validación no está seleccionado no se guardaran las observaciones ni la modificación de horas extras borrando las nuevas observaciones y asignando las horas extras que tenía con anterioridad, en la figura 48 el usuario válido varios campos y agregó una nueva observación a un campo no validado, es importante mencionar que una vez validado los datos y guardados se deshabilitaran para no poder modificarse tal como se muestra en la imagen 49, en la figura 50 se aprecia como la observación validada no se guardó, también se muestra un mensaje de éxito con el número de datos validados, deshabilitando el botón de guardar y habilitando el botón imprimir

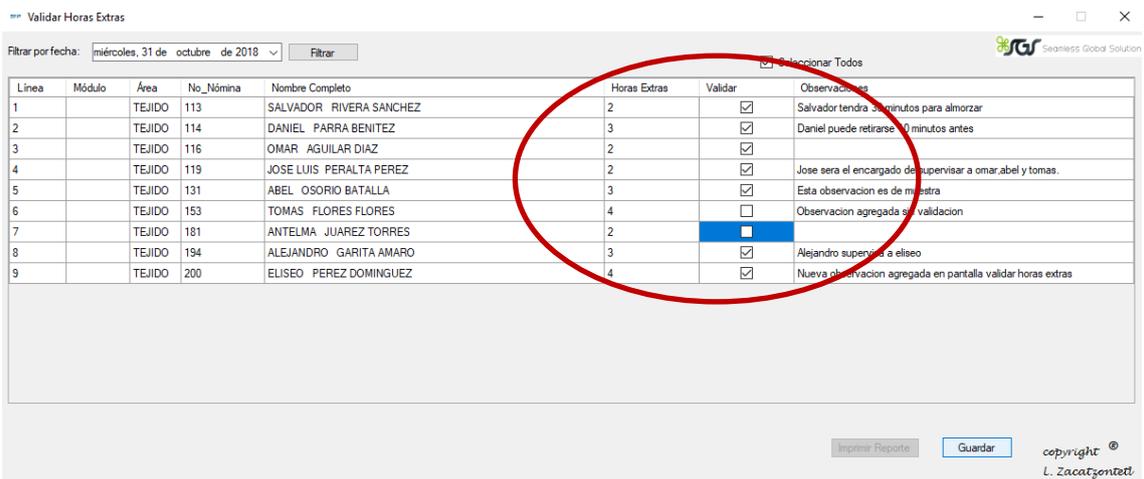


Figura 48. Campos validados y una nueva observación a un campo no validado

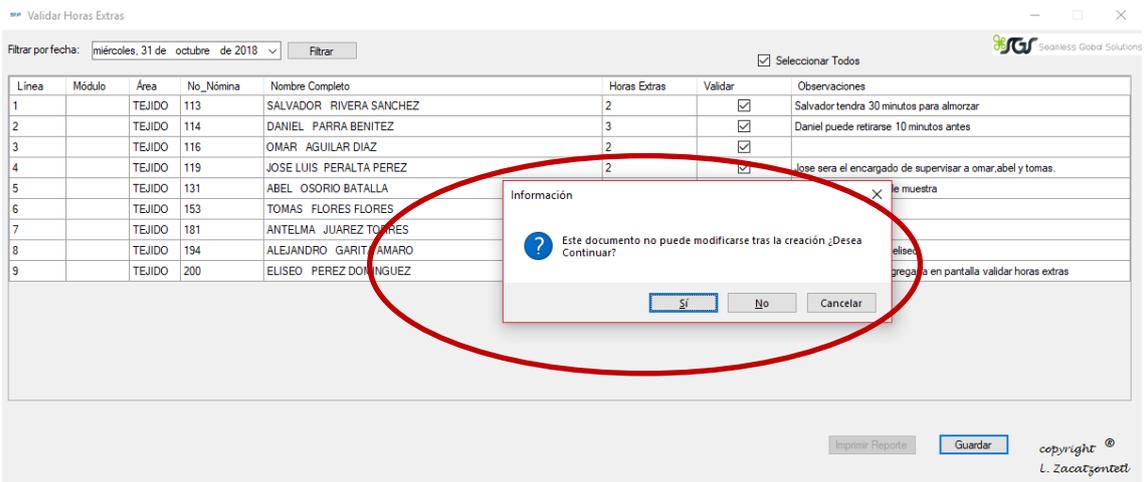


Figura 49. Los datos y el documento no puede modificarse una vez validado y guardado

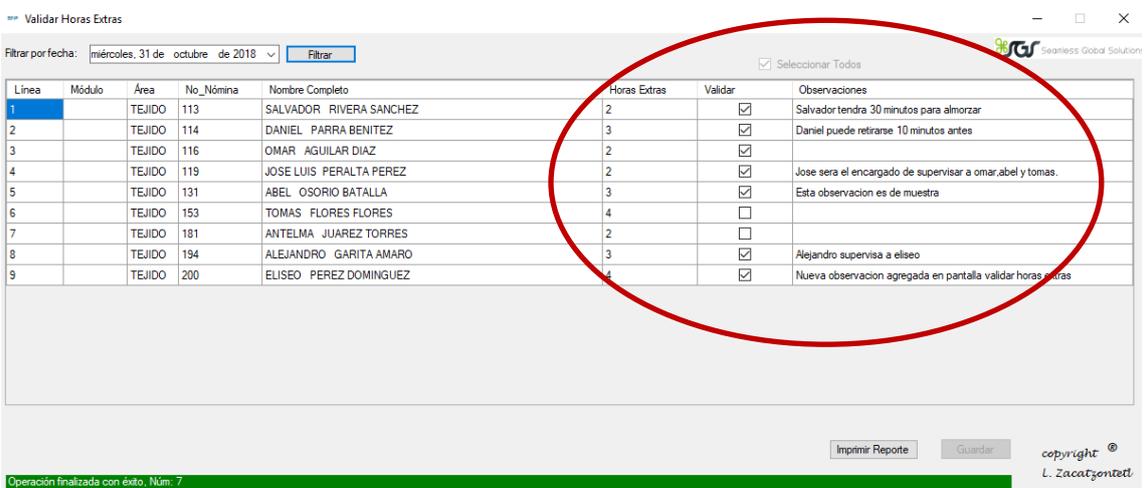


Figura 50. Muestra mensaje de éxito, el botón de guardar deshabilitado y el botón de imprimir habilitado

El botón imprimir nos genera un reporte en crystal report con los datos validado de los empleados que realizaron las horas extras asignadas, al presionarlo nos mostrará una barra de progreso que nos indica el nivel de carga del crystal report y un mensaje de información (cargando) como se muestra en la figura 51

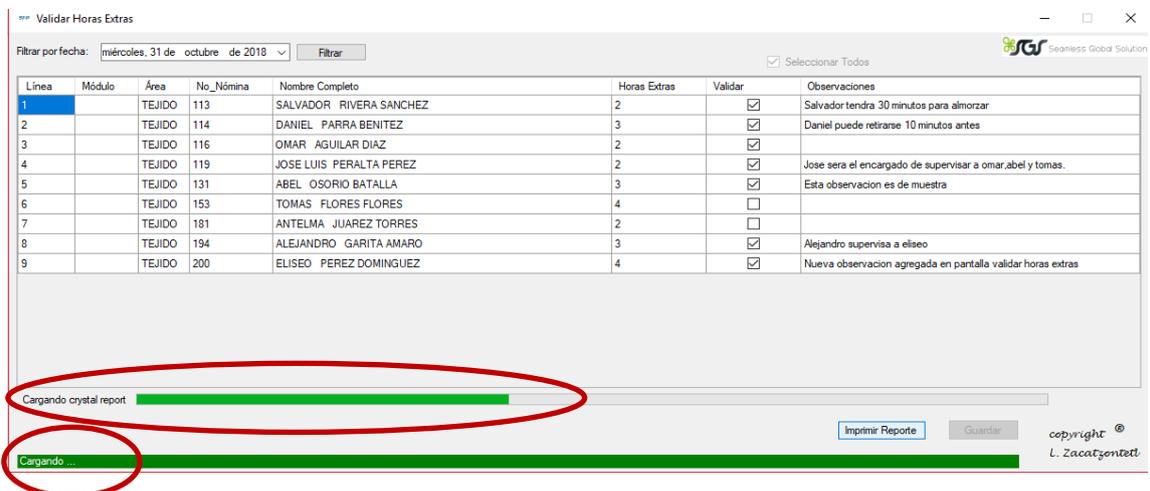


Figura 51. Se muestra el mensaje de cargando y la barra de progreso del crystal report

Posteriormente el crystal report se muestra maximizado en otra ventana con los datos de los empleados que tienen validadas las horas extras que realizaron como se muestra en la figura 52

Reporte de validación de tiempo extra

SAP CRYSTAL REPORTS*

Informe principal

FORMATO VALIDACION DE TIEMPO EXTRA Código: FT-RHSTE-011 Versión: 00

ÁREA: TEJIDO FECHA: 2018-10-31
 JEFE DE ÁREA: ZACATZONTTEL

OTROS: _____

Nº	Nº Empleado	Modulo	Nombre	Horas Extras	Observaciones
1	113		SALVADOR RIVERA SANCHEZ	2	Salvador tendra 30 minutos para almorzar
2	114		DANIEL PARRA BENITEZ	3	Daniel puede retrarse 10 minutos antes
3	116		OMAR AGUILAR DIAZ	2	
4	119		JOSE LUIS PERALTA PEREZ	2	Jose sera el encargado de supervisar a omar abel y tomas
5	131		ABEL OSORIO BATALLA	3	Esta observacion es de muestra
6	194		ALEJANDRO GARITA AMARO	3	Alejandro supervisa a eliseo
7	200		ELISEO PEREZ DOMINGUEZ	4	Nueva observacion agregada en pantalla validar horas extras

Nº de página actual: 1 Nº total de páginas: 1 Factor de zoom: 100%

Figura 52. Crystal Report formato de validación de tiempo extra

4. Conclusiones y recomendaciones

El desarrollar nuevos módulos de un sistema requiere un profundo análisis y diseño de los requisitos que se solicitan puesto que si desde el inicio han sido establecidos correctamente no surgirán problemas durante su implementación, los módulos de horas extra ayudaran en el proceso de nóminas, sin embargo es necesario hacer buen uso del software puesto que si los jefes de área alteran la información a su conveniencia no se podrá tener un control total de las horas extras y el proceso de nomina se vera afectado.

Durante el análisis de este proyecto se detectaron nuevos módulos que podrían desarrollarse como por ejemplo automatizar el control de bonificaciones y el control de retardos, una mejora necesaria que debería de implementarse es juntar los módulos horas extra, bonificaciones, retardos y nóminas en un sólo módulo para mejorar el proceso de nómina.

Durante este proyecto he adquirido nuevos conocimientos, pues desde el inicio el proyecto ha sido un reto personal, puesto que yo no tenía conocimientos en el lenguaje C# y tuve que aprender por mi parte el lenguaje, así como también tuve que tomar un curso intensivo impartido por mi asesor para aprender a desarrollar reportes en crystal report y así implementar el código de cada uno de los módulos. También fue necesario familiarizarme con el lenguaje de consulta SQL Server puesto que nunca había realizado consultas en dicho lenguaje.

En este proyecto desarrolle mis primeros módulos funcionales con requerimientos necesarios y reales por lo que he adquirido experiencia en el análisis diseño e implementación de software, así como también conocí el mundo laboral y las necesidades tecnológicas de una empresa, puesto que, aunque la mayoría de los procesos están automatizados siempre surgirán nuevas mejoras y procesos que automatizar.

5. Anexos

Anexo 1

En este anexo se muestra el código en lenguaje C# de las consultas que realizan el funcionamiento de los módulos solicitud y validación de horas extras

```
class consultasHE
{
    System.Data.DataTable dt = new System.Data.DataTable();
    SqlDataAdapter da;
    public void mostrardgHE(DataGridView dg, String departamento, String fecha, Button
btnImprimirR)
    {
        try
        {
            SqlConnection con = conectarBD.conex();
            da = new SqlDataAdapter("Select modulo,departamento as
area,empleados.no_control as no_nomina,nombre + ' ' + apellidoP + ' ' + apellidoM as
nombre_completo from empleados left join operadorModulo on empleados.no_control=
operadorModulo.no_control where departamento='" +departamento +"' and
status_empleado=1", con);
            System.Data.DataTable dt = new System.Data.DataTable();
            da.Fill(dt);

            foreach (DataRow item in dt.Rows)
            {
                int n = dg.Rows.Add();

                dg.Rows[n].Cells[0].Value = n+1;
                dg.Rows[n].Cells[1].Value = item["modulo"].ToString();
                dg.Rows[n].Cells[2].Value = item["area"].ToString();
                dg.Rows[n].Cells[3].Value = item["no_nomina"].ToString();
                dg.Rows[n].Cells[4].Value = item["nombre_completo"].ToString();

                if (validarSiExisteSHE(fecha) > 0)
                {
                    btnImprimirR.Enabled = true;
                }
            }
            con.Close();
        }
        catch (Exception ex)
        {
            MessageBox.Show("Error, No se pudo llenar el Data Grid" + ex.Message);
        }
    }
    public void mostrardgValidarHE(DataGridView dg)
    {
        try
```

```

    {
        SqlConnection con = conectarBD.conex();
        da = new SqlDataAdapter("Select id_horasExtras, modulo,e.departamento as
area,e.no_control as no_nomina,nombre +' ' + apellidoP +' ' + apellidoM as
nombre_completo,fecha,horas_extras from (( empleados as e inner join operadorModulo om
on e.no_control= om.no_control) inner join horas_extras as he on e.no_control=
he.no_control )", con);
        System.Data.DataTable dt = new System.Data.DataTable();
        da.Fill(dt);
        foreach (DataRow item in dt.Rows)
        {
            int n = dg.Rows.Add();

            dg.Rows[n].Cells[0].Value = item["id_horasExtras"].ToString();
            dg.Rows[n].Cells[1].Value = item["modulo"].ToString();
            dg.Rows[n].Cells[2].Value = item["area"].ToString();
            dg.Rows[n].Cells[3].Value = item["no_nomina"].ToString();
            dg.Rows[n].Cells[4].Value = item["nombre_completo"].ToString();
            dg.Rows[n].Cells[5].Value = item["horas_extras"].ToString();
        }
        con.Close();
    }
    catch (Exception ex)
    {
        MessageBox.Show("Error, No se pudo llenar el Data Grid" + ex.Message);
    }
}
public void mostrardgValidarHEPorFecha(DataGridView dg, String fecha, String
departamento, CheckBox cb, Button btn, Button btnImprimirR)
{
    try
    {
        SqlConnection con = conectarBD.conex();
        da = new SqlDataAdapter("Select om.modulo,
he.no_control,he.departamento,nombre +' ' + apellidoP +' ' + apellidoM as
nombre_completo,he.horas_extras, he.validar, he.observaciones from ((horas_extras as he
left join operadorModulo as om on he.no_control=om.no_control) inner join empleados as e
on he.no_control= e.no_control) where he.departamento='" + departamento + "' and fecha='"
+ fecha + "'and status_empleado=1", con);
        System.Data.DataTable dt = new System.Data.DataTable();
        da.Fill(dt);
        foreach (DataRow item in dt.Rows)
        {
            int n = dg.Rows.Add();

            dg.Rows[n].Cells[0].Value = n + 1;
            dg.Rows[n].Cells[1].Value = item["modulo"].ToString();
            dg.Rows[n].Cells[2].Value = item["departamento"].ToString();
            dg.Rows[n].Cells[3].Value = item["no_control"].ToString();
            dg.Rows[n].Cells[4].Value = item["nombre_completo"].ToString();
            dg.Rows[n].Cells[5].Value = item["horas_extras"].ToString();

            if (validarSiExisteV(fecha) > 0)

```

```

        {
            btn.Enabled = false;
            cb.Enabled = false;
            btnImprimirR.Enabled = true;
            if (Convert.ToInt32(item["validar"].ToString()) == 1)
            {
                dg.Rows[n].Cells[6].Value = true;
                dg.Rows[n].Cells[5].ReadOnly = true;
                dg.Rows[n].Cells[6].ReadOnly = true;
                dg.Rows[n].Cells[7].ReadOnly = true;
            }
            else
            {
                dg.Rows[n].Cells[6].Value = false;
                dg.Rows[n].Cells[5].ReadOnly = true;
                dg.Rows[n].Cells[6].ReadOnly = true;
                dg.Rows[n].Cells[7].ReadOnly = true;
            }
        }
        else
        {
            btn.Enabled = true;
            cb.Enabled = true;
            dg.Rows[n].Cells[6].Value = false;
        }

        dg.Rows[n].Cells[7].Value = item["observaciones"].ToString();
    }
    con.Close();

}
catch (Exception ex)
{
    MessageBox.Show("Error, No se pudo llenar el Data Grid" + ex.Message);
}
}

```

```

public void guardarHE(int no_control, String departamento, String fecha, double
horas_extras, String observaciones, String fechaRegistro)
{
    try
    {
        SqlConnection con = conectarBD.conex();
        SqlCommand cmdc = new SqlCommand("insert into horas_extras values (" +
no_control + ", " + departamento + ", " + fecha + ", " + horas_extras + ", " + observaciones +
", " + fechaRegistro + ")", con);
        SqlDataReader dr = cmdc.ExecuteReader();
        con.Close();
    }
    catch (Exception ex)
    {

```

```

        MessageBox.Show("Error" + ex.Message);
    }
}

public void guardarHE2(int no_control, String departamento, String fecha, double
horas_extras,String fechaRegistro)
{
    try
    {
        SqlConnection con = conectarBD.conex();
        SqlCommand cmdc = new SqlCommand("insert into horas_extras values (" +
no_control + "," + departamento + "," + fecha + "," + horas_extras + "," + fechaRegistro +
"")", con);
        SqlDataReader dr = cmdc.ExecuteReader();
        con.Close();
    }
    catch (Exception ex)
    {
        MessageBox.Show("Error" + ex.Message);
    }
}

public int validarHE(int no_control, String fecha)
{
    int contador = 0;
    SqlConnection con= conectarBD.conex();
    SqlCommand cmdc = new SqlCommand("Select * from horas_extras where
no_control=" + no_control + "and fecha=" + fecha + """, con);
    SqlDataReader dr;
    dr= cmdc.ExecuteReader();
    while (dr.Read())
    {
        contador++;
    }
    con.Close();
    return contador;
}

public int validarSinObservaciones(int horasExtras, int validar, int no_control, String fecha)
{
    int contador = 0;
    SqlConnection con = conectarBD.conex();
    SqlCommand cmdc = new SqlCommand("Select * from horas_extras where
fecha=" + fecha + "" and no_control=" + no_control + "" and horas_extras=" + horasExtras + ""and
validar=" + validar + """, con);
    SqlDataReader dr;
    dr = cmdc.ExecuteReader();
    while (dr.Read())
    {
        contador++;
    }
    con.Close();
    return contador;
}
}

```

```

    public int validarConObservaciones(int horasExtras, int validar, String observaciones, int
no_control, String fecha)
    {
        int contador = 0;
        SqlConnection con = conectarBD.conex();
        SqlCommand cmdc = new SqlCommand("Select * from horas_extras where fecha=" +
fecha + " and no_control=" + no_control + " and horas_extras=" + horasExtras + "and validar="
+ validar + " and observaciones=" + observaciones + """, con);
        SqlDataReader dr;
        dr = cmdc.ExecuteReader();
        while (dr.Read())
        {
            contador++;
        }
        con.Close();
        return contador;
    }

```

```

String fechaV;
public String validarFecha(String departamento)
{
    SqlConnection con = conectarBD.conex();
    SqlCommand cmd = new SqlCommand("SELECT MAX(fecha) FROM horas_extras
where departamento=" + departamento + """, con);
    SqlDataReader dr;
    dr = cmd.ExecuteReader();
    while (dr.Read())
    {
        fechaV = dr[0].ToString(); // obtener el valor de
if (fechaV == String.Empty)
        {
            DateTime hoy = DateTime.Today;
            fechaV = hoy.ToString("yyyy-MM-dd");
        }
    }
    dr.Close();
    con.Close();
    return fechaV;
}

```

```

public int validarSiExisteV(String fecha)
{
    int contador = 0;
    try
    {
        SqlConnection con = conectarBD.conex();
        SqlCommand cmd = new SqlCommand("Select * from horas_extras where
fecha=" + fecha + " and validar=1", con);
        SqlDataReader dr;
        dr = cmd.ExecuteReader();
        while (dr.Read())
        {
            contador++;
        }
    }
}

```

```

        con.Close();
    }
    catch (Exception ex)
    {
        MessageBox.Show("Erro: " + ex.ToString());
    }
    return contador;
}
public int validarSiExisteSHE(String fecha)
{
    int contador = 0;
    try
    {
        SqlConnection con = conectarBD.conex();
        SqlCommand cmd = new SqlCommand("Select * from horas_extras where fecha=" +
fecha + """, con);
        SqlDataReader dr;
        dr = cmd.ExecuteReader();
        while (dr.Read())
        {
            contador++;
        }
        con.Close();
    }
    catch (Exception ex)
    {
        MessageBox.Show("Error: " + ex.ToString());
    }
    return contador;
}
public void actualizarValidacion(DataGridView dg,double horasExtras ,int validar, String
observaciones, int no_control, String fecha)
{
    SqlConnection con = conectarBD.conex();
    string sc = "Update horas_extras set horas_extras="+horasExtras+", validar=" + validar +
", observaciones =" + observaciones + "" where no_control=" + no_control + "and fecha= " +
fecha + """;
    SqlCommand cmd = new SqlCommand(sc, con);
    cmd.ExecuteNonQuery();
    con.Close();
}

public void actualizarValidacionSinO(DataGridView dg, double horasExtras, int validar, int
no_control, String fecha)
{
    SqlConnection con = conectarBD.conex();
    string sc = "Update horas_extras set horas_extras=" + horasExtras + ", validar=" + validar
+ " where no_control=" + no_control + "and fecha= " + fecha + """;
    SqlCommand cmd = new SqlCommand(sc, con);
    cmd.ExecuteNonQuery();
    con.Close();
}
}

```

```

public int validarSiYaExisteLaObservacion(String observaciones, int no_control)
{
    int contador = 0;
    try
    {
        SqlConnection con = conectarBD.conex();
        SqlCommand cmd = new SqlCommand("Select * from horas_extras where
observaciones='"+observaciones+"' and no_control='"+no_control+"", con);
        SqlDataReader dr;
        dr = cmd.ExecuteReader();
        while (dr.Read())
        {
            contador++;
        }
        con.Close();
    }
    catch (Exception ex)
    {
        MessageBox.Show("Error: " + ex.ToString());
    }
    return contador;
}
}

```

6. Referencias bibliográficas

- [1] URL: <https://openclassrooms.com/en/courses/4309151-gestiona-tu-proyecto-de-desarrollo/4538221-en-que-consiste-el-modelo-en-cascada>.
Página principal de la plataforma de aprendizaje en línea Open Classrooms, en ella se puede consultar la gestión de un proyecto en desarrollo. Fecha de consulta: 9/Octubre/2018
- [2] Sommerville, Ian. “Ingeniería Del Software” Editorial PEARSON EDUCACION, Madrid,2005
- [3] Petersen, Kai. “The Waterfall Model in Large-ScaleDevelopment” Editorial Blekinge Institute of Technology, Suecia,2015
- [4] URL: <https://www.obs-edu.com/int/blog-project-management/metodologia-agile/pros-y-contras-de-la-metodologia-en-cascada> Página de OBS Business School escuela en línea de negocios en ella se puede consultar información acerca de la Gestión de proyectos
- [5] Tobias Kahlert, Kay Giza. “Visual Studio Code Tips & Tricks Vol. 1” Editorial Microsoft, Alemania, 2016
- [6] URL: <https://msdn.microsoft.com/es-mx/library/bb545450.aspx> Pagina oficial de Microsoft en la que se puede consultar información acerca de los productos de Microsoft Fecha de consulta: 11/Octubre/2018
- [7]URL:https://www.danysoft.com/free/SQL_Server_2012_Gives_You_More_Advanced.pdf Pagina oficial de Danysoft en la que se puede consultar información sobre SQL Server Fecha de consulta: 11/Octubre/2018
- [8] URL: <https://sqlserver4b.weebly.com/ventajas-y-desventajas.html> Página Weebly en la que se puede consultar información sobre SQL Server Fecha de consulta: 11/Octubre/2018
- [9] URL: https://www.ecured.cu/Arquitectura_Cliente_Servidor Página EcuRed es una enciclopedia en la cual se puede consultar cualquier tipo de información Fecha de consulta: 7/Noviembre/2018



Universidad Politécnica de Puebla
Ingeniería en Informática

Laura Zacatzontetl Hernández
Daniel Paniagua González
Rebeca Rodríguez Huesca

Este documento se distribuye para los términos de la
Licencia 2.5 Creative Commons (CC-BC-NC-ND 2.5 MX)