PORTADA

**INDICE**

Introducción……………………………………………………………………………………. 3

Descripción del problema…………………………………………………………………….. 4

Solución del problema………………………………………………………………………… 5

Análisis de resultados…………………………………………………………………………. 6

Bitácora de responsables…………………………………………………………………….. 7

Conclusiones.………………………………………………………………………………….. 8

Bibliografía……………………………………………………………………………………… 9

Anexos………………………………………………………………………………………….. 9

**INTRODUCCIÓN**

A continuación se muestra la problemática como su forma en que se realizo la aplicación web utilizando diferentes tecnologías como HTML, CSS, Javascript, PHP y hojas de Excel y además del complemento de PHPSpreadSheet para poder conectarnos fácilmente a la base de datos (Excel).

Se crearon diferentes secciones indispensables para cada acción necesaria y se agregó un login y registro al inicio para que solo usuarios registrados pudiesen entrar al sistema.

Se creo un diseño simple con iconos para que fuera fácil de usar, demás de ciertas validaciones indispensables para su correcto funcionamientos pero aún así fueron saltas otras cuantas por falta de tiempo.

Este proyecto no fue completado como se debía por falta de tiempo pero se avanzó lo suficiente para su funcionamiento.

**DESCRIPCIÓN DEL PROBLEMA**

El presente proyecto consiste en crear una App Web para el manejo de planillas de

trabajadores, planilla que puede ser ajustable o adaptable a cualquier empresa que requiera

o desee implementarla. Esta utilizará como base de datos una hoja de cálculo que bien

puede ser Excel, Calc o la que el equipo de desarrollo considere más oportuna y adecuada,

además de la hoja de cálculo, dado que se trata de una App Web, se debe utilizar HTML5,

CSS3 y PHP v5.6 como versión mínima a utilizar, recomendándose utilizar la v7.2, junto

con la Library PhpSpreadsheet o bien puede utilizar la library PhpExcel, en conjunto con

las herramientas de desarrollo VisualStudio Code y Xampp v3.2.2 como administrador de

servicios.

La App debe contemplar que el pago de los trabajadores es por horas,

individualizado para paga trabajador según el puesto que desempeñe, lo que quiere decir

que debe de establecer un monto distinto a pagar por hora a cada asalariado. El pago es

semanal, diariamente de lunes a sábado debe de registrarse las hora trabajadas, laborando

un máximo de 48 horas semanales según lo establece el actual código de trabajo, si el

empleado trabaja más de las 48 horas semanales quiere decir que este está trabajando

tiempo extra, tiempo que deberá pagarse en tiempo y medio, por último y una vez calculado

el salario bruto deben contemplarse las deducciones clásicas del 10.1% de la CCSS y el 2%

por Asoc. Solidarista para calcular el neto.

**SOLUCIÓN DEL PROBLEMA**

Para resolver la problemática primero se tuvo que analizar el problema y sacar todo aquello que se tendría que hacer, todos los requerimientos que eran necesario y después habría que saber conectar correctamente a la base de datos, que en este caso fue con hojas de excel.

Una vez que se tenían los requerimientos se fue buscando soluciones para resolver la problemática, entonces se continuo poco a poco creando las interfaces en HTML y CSS después de ello había que hacer que funcionará todo así que se continuo primero creando la clase employees y después la clase users (para el inicio de sesión) con ello se fueron creando métodos para leer el archivo o crearlo si no existía, después así se fueron creando los métodos necesarios, estos son algunos:

- registerEmployee: para registrar un nuevo empleado

- calculate: para calcular los totales del salario, horas, etc.

- saveOperations: para guardar los cálculos en la hoja de excel

- createSheetOfWeek: para crear una nueva hoja de excel (que sería una nueva semana)

- fillSheet: método necesario para pasar la información de la hoja anterior a la nueva

- readFile: para leer el archivo

- updateHours: especial para actualizar las horas del día

- showTableHoursDay: para mostrar nombre, apellido y una caja de texto para registrar horas del día.

- showTable: para mostrar la tabla con el nombre, apellidos y los días a trabajar (mostrar las horas).

- showSalaryTable: para mostrar más datos.

- getLastWeek: para mostrar el nombre de la hoja actual

- dayOfWeek: muestra el día de la semana de hoy

- getDay: obtiene la fecha actual

Con esos métodos se llegó a una solución parcial porque se requería más tiempo para resolver todo lo que se pidió.

**ANÁLISIS DE RESULTADOS**

Los resultados fueron los esperados, se logro hacer registros para resolver las problemáticas como registrar usuario, empleados o actualizar horas, además de mostrar los datos necesarios en tablas e internamente hacer las operaciones necesarias.

**BITÁCORA DE RESPONSABILIDADES**

**CONCLUSIONES**

Por medio de este proyecto se lograron adquirir diferentes conocimientos en programación web, además de aprender a usar la librería de PHPSpreadsheet el cual sirve para facilitar la conexión de php con hojas de excel, ayudó a reforzar también conocimiento y a tener motivación para seguir haciendo app webs.

BIBLIOGRAFÍA

<https://phpspreadsheet.readthedocs.io/en/develop/>

<https://phpoffice.github.io/PhpSpreadsheet/master/>

<https://www.w3schools.com/>

<http://php.net/manual/es/>

ANEXOS