

**UNIVERSIDAD INTERNACIONAL DE LAS AMÉRICAS  
ESCUELA DE INGENIERÍA INFORMÁTICA**

**SEMINARIO DE GRADUACIÓN  
PRÁCTICA PROFESIONAL DIRIGIDA  
TESIS DE GRADUACIÓN**

Para optar por el grado de Bachillerato en  
Ingeniería en Sistemas de Información

**PROTOTIPO FUNCIONAL PARA LA GESTIÓN DE RECURSOS  
HUMANOS DE LA ASOCIACIÓN DE DESARROLLO INTEGRAL DE SAN JUAN  
PURISCAL**

LUIS ROBERTO DUARTE VINDAS

**AUTOR**

MBD. OLMAN NÚÑEZ PERALTA

**TUTOR**

MAP. FABIÁN RODRÍGUEZ SIBAJA

**LECTOR**

**San José, Costa Rica**

**AGOSTO, 2020**

## **Dedicatoria**

Lo difícil no es pensar en elegir las personas a quien dedicarle la elaboración de este proyecto, si no, elegir las palabras correctas que serán plasmadas en este pequeño espacio. No suelo ser un hombre de muchas palabras en los momentos más importantes de mi vida; pero siempre trato de expresarme de alguna manera, aunque sea la más mínima posible.

Por esto quiero dedicarles este proyecto a mis padres, es por ellos que soy la calidad de persona hoy día. Por ellos estoy en este punto de mi vida. Sí, el más importante hasta el momento; por ellos este logro que finaliza tras varios años de esfuerzo. Hubo algunas caídas, pero, nunca me quedé atrás; aunque en algunos momentos sentía que no podía más. Siempre tuve como meta concluir mis estudios para agradecerles todo el esfuerzo que realizaron por mí y mis hermanos.

También dedicarles este logro a aquellos familiares que de alguna u otra manera me estuvieron apoyando y que siempre creyeron en mí, aunque me haya distanciado un poco. Espero aprecien este pequeño regalo que no es para mí, si no para Ustedes. pero me hace muy feliz y espero compartir esta celebración con todos reunidos como lo hacíamos antes.

## **Agradecimientos**

Primeramente, quiero agradecerles a todos los profesores que acogieron como estudiante y me brindaron el conocimiento que tengo hoy día, también hicieron que amara esta carrera más y más con cada materia que cursaba, si no fuera por ellos no estaría en esta etapa de mi vida profesional.

Gracias a aquellos, compañeros que me acompañaron durante esta travesía; esas salvadas que me hicieron no tengo como agradecerles, en especial a “Ninja” le debo mucho algunos ya les devolví el favor, así que quedamos a mano. Fueron años los que estuvimos compartiendo y más que compañeros, los considero amigos y me siento realmente afortunado, como dicen” El que gana un amigo, gana un tesoro”.

Por último, y no menos importante, a mi tutor, Olman, quien me estuvo guiando y aconsejando durante todo este proceso, aunque tengo que admitir que sufrí un poco en Base de Datos. Pero, realmente aprendí; el mejor profesor de Base de Datos que he tenido y he conocido.

Mis más sinceros agradecimientos a todos, de verdad, muchas gracias.

## CONTENIDO

<b>RESUMEN EJECUTIVO.....</b>	<b>1</b>
<b>CAPITULO I: INTRODUCCIÓN .....</b>	<b>2</b>
<b><i>Planteamiento del problema.....</i></b>	<b>2</b>
Descontento de los trabajadores por la gestión de horas extras. ....	2
Ausencia de un control de administración de la planilla.....	2
Deficiencia en la definición de tareas según los roles. ....	2
Carencia de un control de vacaciones para los empleados.....	3
Ausencia de un control disciplinario para los empleados. ....	3
Ausencia de un control para gestionar de los permisos de salida.....	3
Ausencia de un control para realizar los pagos a los empleados.....	3
<b><i>Objetivos .....</i></b>	<b>4</b>
Objetivo general .....	4
Objetivos específicos .....	4
<b><i>Justificación .....</i></b>	<b>5</b>
Viabilidad técnica .....	6
Viabilidad operativa.....	7
Viabilidad económica .....	7
Viabilidad legal.....	9
<b><i>Proyecciones.....</i></b>	<b>9</b>
Alcance funcional .....	10
Módulo de seguridad .....	10
Módulo de mantenimiento.....	10
Módulo de consultas.....	10
Módulo de reportes.....	11
Módulo de gestión de planilla.....	11
Módulo de gestión de vacaciones .....	11
Módulo de control de calidad de choferes. ....	11
Alcance tecnológico.....	12

Alcance metodológico.....	12
<b>CAPÍTULO II: MARCO REFERENCIAL.....</b>	<b>14</b>
<b>CAPÍTULO III: MARCO METODOLÓGICO .....</b>	<b>24</b>
<i>Enfoque cualitativo .....</i>	<i>24</i>
<i>Enfoque de la investigación.....</i>	<i>24</i>
<i>Investigación descriptiva .....</i>	<i>25</i>
<i>Método de la investigación .....</i>	<i>25</i>
<i>Fuente de información primaria .....</i>	<i>25</i>
<i>Fuente de información secundaria.....</i>	<i>25</i>
<i>Fuente de información terciaria.....</i>	<i>26</i>
<i>Variable conceptual.....</i>	<i>26</i>
<i>Variable operacional .....</i>	<i>26</i>
<i>Variable instrumental.....</i>	<i>26</i>
<i>Variables requeridas.....</i>	<i>26</i>
<i>Población.....</i>	<i>28</i>
<i>Muestra.....</i>	<i>29</i>
<i>Instrumentos de recolección de datos .....</i>	<i>30</i>
Entrevista.....	30
Observación.....	31
<i>Proceso para la recolección y análisis de datos .....</i>	<i>31</i>
<b>CAPITULO IV: ANÁLISIS DE RESULTADOS .....</b>	<b>32</b>
<i>Resultados de la entrevista.....</i>	<i>32</i>
<i>Resultados de la observación .....</i>	<i>34</i>
<i>Análisis de resultados .....</i>	<i>39</i>

<b>CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>41</b>
<i>Conclusiones .....</i>	<i>41</i>
<i>Recomendaciones .....</i>	<i>42</i>
<b>CAPÍTULO VI: PROPUESTA .....</b>	<b>44</b>
<i>Análisis .....</i>	<i>44</i>
Casos de uso .....	44
Análisis detallado del Software desarrollado .....	83
Análisis detallado del hardware requerido .....	86
Análisis detallado de las telecomunicaciones .....	87
<i>Diseño.....</i>	<i>87</i>
Arquitectura del sistema.....	87
Arquitectura del software .....	88
Diseño de Interfaces.....	89
Diseño de procesos .....	95
<i>Diagramas UML.....</i>	<i>111</i>
Diagrama de clases .....	111
Diseño de base de datos .....	112
Diccionario .....	113
<i>Programación.....</i>	<i>125</i>
Entradas y salidas .....	125
Procesos.....	129
Validaciones .....	130
Módulos.....	131
<i>Pruebas.....</i>	<i>132</i>
<b>ANEXOS .....</b>	<b>152</b>
<i>Anexo #1. Entrevista realizada .....</i>	<i>152</i>
<b>REFERENCIAS.....</b>	<b>153</b>

## Tablas

Tabla 1. Costos de desarrollo.....	8
Tabla 2. Costos de implementación.....	8
Tabla 3. Impuestos sobre la renta.....	23
Tabla 4. Definición de variables.....	26
Tabla 5. Caso de uso 01 – Inicio de sesión.....	46
Tabla 6. Caso de uso 02 - Nuevo empleado.....	47
Tabla 7. Caso de uso 03 - Actualizar empleado.....	48
Tabla 8. Caso de uso 04 -Recontratar empleado temporal.....	49
Tabla 9. Caso de uso 05 - Despedir empleado.....	50
Tabla 10. Caso de uso 06 - Nuevo permiso.....	51
Tabla 11. Caso de uso 07 - Ver permisos.....	52
Tabla 12. Caso de uso 08 - Aprobar o rechazar permiso.....	54
Tabla 13. Caso de uso 09 - Editar permiso.....	55
Tabla 14. Caso de uso 10 - Borrar permiso.....	56
Tabla 15. Caso de uso 11 - Perfil de usuario.....	58
Tabla 16. Caso de uso 12 - Editar información de contacto.....	58
Tabla 17. Caso de uso 13 - Nueva solicitud horas extras.....	60
Tabla 18. Caso de uso 14 - Ver horas extras.....	61
Tabla 19. Caso de uso 15 - Aprobar o rechazar horas extras.....	62
Tabla 20. Caso de uso 16 - Editar solicitud horas extras.....	63
Tabla 21. Caso de uso 17 - Borrar solicitud horas extras.....	65
Tabla 22. Caso de uso 18 - Nueva solicitud de vacaciones.....	66

Tabla 23. Caso de uso 19 - Ver solicitudes vacaciones .....	67
Tabla 24. Caso de uso 20 - Registrar voto .....	69
Tabla 25. Caso de uso 21 - Ver calidad servicio de choferes .....	70
Tabla 26. Caso de uso 22 - Amonestaciones .....	71
Tabla 27. Caso de uso 23 - Felicitaciones .....	72
Tabla 28. Caso de uso 24 - Cálculo salarial.....	74
Tabla 29. Caso de uso 25 - Aumentos salariales.....	75
Tabla 30. Caso de uso 26 - Retenciones salariales.....	76
Tabla 31. Caso de uso 27 - Incapacidades .....	78
Tabla 32. Caso de uso 28 - Viáticos .....	79
Tabla 33. Caso de uso 29 - Nueva tarea .....	80
Tabla 34. Caso de uso 30 - Ver tareas .....	81
Tabla 35. Matriz de requerimientos.....	85
Tabla 36. Hardware utilizado para el desarrollo .....	86
Tabla 37. Servicios para puesta en producción.....	86
Tabla 38. Diccionario para asignaciones choferes .....	113
Tabla 39. Diccionario para asistencias .....	113
Tabla 40. Diccionario para aumentos salariales.....	114
Tabla 41. Diccionario para cantones .....	114
Tabla 42. Diccionario para contratos empleados temporales .....	114
Tabla 43. Diccionario para control calidad.....	115
Tabla 44. Diccionario para despidos .....	115
Tabla 45. Diccionario para direcciones .....	116
Tabla 46. Diccionario para distritos .....	116

Tabla 47. Diccionario para empleados .....	117
Tabla 48. Diccionario para empleados temporales .....	118
Tabla 49. Diccionario para estados .....	118
Tabla 50. Diccionario para horarios choferes .....	118
Tabla 51. Diccionario para horas extras .....	119
Tabla 52. Diccionario para incapacidades .....	119
Tabla 53. Diccionario para permisos .....	120
Tabla 54. Diccionario para provincias.....	120
Tabla 55. Diccionario para retenciones salariales .....	121
Tabla 56. Diccionario para salarios .....	121
Tabla 57. Diccionario para tareas.....	122
Tabla 58. Diccionario para teléfonos.....	122
Tabla 59. Diccionario para tipo empleados .....	122
Tabla 60. Diccionario para tipo teléfonos.....	123
Tabla 61. Diccionario para vacaciones .....	123
Tabla 62. Diccionario para vacaciones disponibles .....	124
Tabla 63. Caso de prueba 01 .....	132
Tabla 64. Caso de prueba 02 .....	132
Tabla 65. Caso de prueba 03 .....	133
Tabla 66. Caso de prueba 04 .....	133
Tabla 67. Caso de prueba 05 .....	133
Tabla 68. Caso de prueba 06.....	134
Tabla 69. Caso de prueba 07.....	134
Tabla 70. Caso de prueba 08.....	134

Tabla 71. Caso de prueba 09.....	135
Tabla 72. Caso de prueba 10.....	135
Tabla 73. Caso de prueba 11.....	136
Tabla 74. Caso de prueba 12.....	136
Tabla 75. Caso de prueba 13.....	136
Tabla 76. Caso de prueba 14.....	137
Tabla 77. Caso de prueba 15.....	137
Tabla 78. Caso de prueba 16.....	138
Tabla 79. Caso de prueba 17.....	138
Tabla 80. Caso de prueba 18.....	139
Tabla 81. Caso de prueba 19.....	139
Tabla 82. Caso de prueba 20.....	139
Tabla 83. Caso de prueba 21.....	140
Tabla 84. Caso de prueba 22.....	140
Tabla 85. Caso de prueba 23.....	140
Tabla 86. Caso de prueba 24.....	141
Tabla 87. Caso de prueba 25.....	141
Tabla 88. Caso de prueba 26.....	142
Tabla 89. Caso de prueba 27.....	142
Tabla 90. Caso de prueba 28.....	142
Tabla 91. Caso de prueba 29.....	143
Tabla 92. Caso de prueba 30.....	143
Tabla 93. Caso de prueba 31.....	144
Tabla 94. Caso de prueba 32.....	144

Tabla 95. Caso de prueba 33 .....	145
Tabla 96. Caso de prueba 34 .....	145
Tabla 97. Caso de prueba 35 .....	145
Tabla 98. Caso de prueba 36 .....	146
Tabla 99. Caso de prueba 37 .....	146
Tabla 100. Caso de prueba 38 .....	146
Tabla 101. Caso de prueba 39 .....	147
Tabla 102. Caso de prueba 40 .....	147
Tabla 103. Caso de prueba 41 .....	148
Tabla 104. Caso de prueba 42 .....	148
Tabla 105. Caso de prueba 43 .....	148
Tabla 106. Caso de prueba 44 .....	149
Tabla 107. Caso de prueba 45 .....	149
Tabla 108. Caso de prueba 46 .....	149
Tabla 109. Caso de prueba 47 .....	150
Tabla 110. Caso de prueba 48 .....	150

## Figuras

Figura 1. Metodología en cascada. ....	15
Figura 2. Arquitectura cliente servidor. ....	18
Figura 3. Cargas sociales .....	22
Figura 4. Fases o etapas del enfoque cualitativo. ....	24
Figura 5. Fórmula para calcular la muestra. ....	29
Figura 6. Proceso cálculo planilla .....	35
Figura 7. Proceso control disciplinario .....	36
Figura 8. Proceso asignación tareas. ....	37
Figura 9. Proceso vacaciones .....	38
Figura 10. Diagrama de casos de uso .....	45
Figura 11. Arquitectura del sistema. ....	87
Figura 12. Arquitectura de software .....	88
Figura 13. Interfaz para el inicio de sesión .....	89
Figura 14. Interfaz para registrar nuevo empleado .....	90
Figura 15. Interfaz para los permisos solicitados .....	91
Figura 16. Interfaz para las horas extras solicitadas .....	91
Figura 17. Interfaz para consultar pagos. ....	92
Figura 18. Interfaz para nueva tarea .....	93
Figura 19. Interfaz para solicitar vacaciones .....	93
Figura 20. Interfaz para nueva amonestación .....	94
Figura 21. Proceso inicio de sesión .....	95
Figura 22. Proceso nuevo empleado .....	96
Figura 23. Proceso nuevo permiso .....	97

Figura 24. Proceso ver permisos .....	98
Figura 25. Proceso nueva hora extra .....	99
Figura 26. Proceso ver horas extras.....	100
Figura 27. Proceso aumento salarial.....	101
Figura 28. Proceso retención salarial.....	102
Figura 29. Proceso incapacidad.....	103
Figura 30. Proceso viáticos .....	104
Figura 31. Proceso calculo salarial.....	105
Figura 32. Proceso amonestaciones.....	106
Figura 33. Proceso felicitaciones.....	107
Figura 34. Proceso nueva tarea .....	108
Figura 35. Proceso ver tareas .....	109
Figura 36. Proceso solicitar vacaciones.....	110
Figura 37. Diagrama de clases .....	111
Figura 38. Diagrama de base de datos.....	112
Figura 39. Entrada vacaciones .....	125
Figura 40. Entrada permiso .....	126
Figura 41. Salida salario .....	127
Figura 42. Salida tarea .....	128
Figura 43. Proceso de configuración del servidor.....	129
Figura 44. Validación de la sesión .....	130
Figura 45. Módulo tareas.....	131

## **Resumen ejecutivo**

El presente proyecto llamado Prototipo Funcional para la Gestión de Recursos Humanos de la Asociación de Desarrollo Integral de San Juan Puriscal, tiene como objetivo desarrollar un prototipo funcional que ayude a solucionar un conjunto de problemáticas que se presentan en la empresa, estas en relación a la gestión de sus recursos humanos.

En el primer apartado de este documento se muestra el planteamiento del problema y los diferentes objetivos que ayudarán a solucionar los problemas descritos, la justificación y proyecciones para la ejecución del proyecto.

Por otro lado, para el desarrollo del proyecto, se realizaron cuatro distintas fases las cuales poseen un orden y relación entre sí, estas son análisis, diseño, programación y pruebas. En la primera fase se muestra todo el análisis previamente realizado, en este se encuentran las diferentes fuentes de información que se utilizaron, el análisis de viabilidades, proyecciones y resultados obtenidos en las entrevistas y la observación aplicados en la empresa. En la segunda fase, la cual corresponde a la de diseño, se detalla el funcionamiento del sistema mediante la elaboración de distintos diagramas, procesos y arquitecturas las cuales son relevantes para la siguiente fase. En la fase de programación se desarrollaron diferentes módulos que en conjunto forman un sistema el cual integra los procesos de recursos humanos existentes en la empresa. Posteriormente, en la fase final se muestran las distintas pruebas realizadas al prototipo, así como los resultados obtenidos de estas con la finalidad de comprobar un correcto funcionamiento.

Finalmente se da por finalizado el proyecto en el apartado de conclusiones abarcando todo lo acontecido en este documento y dando el cierre de los objetivos que se plantearon en un inicio. Asimismo, se mencionan las recomendaciones con el fin de suministrar nuevas mejoras para las futuras actualizaciones en el prototipo funcional.

## **CAPITULO I: INTRODUCCIÓN**

### **Planteamiento del problema**

El siguiente apartado presenta los problemas encontrados. Actualmente la Asociación de Desarrollo Integral de San Juan Puriscal presenta un mal manejo de su información sobre los recursos humanos, dicha información se encuentra gestionada en hojas de cálculo de Excel, estas hojas contemplan datos de cálculos para pagos de salarios, pagos de horas extras e información de los empleados

Por otra parte, en hojas de papel, se cuenta con información sobre tareas que debe realizar cada empleado según su rol asignado. A continuación, se enumeran los problemas detectados.

### **Descontento de los trabajadores por la gestión de horas extras.**

Los trabajadores presentan descontento ya que no se cuenta con un registro adecuado para confirmar las horas extras laboradas, lo que causa que los trabajadores en ocasiones soliciten el pago inmediatamente.

Esta situación provoca un descontrol sobre el pago total ya que el encargado de realizar los cálculos en ocasiones no conoce con exactitud cuantas horas realizó el trabajador.

### **Ausencia de un control de administración de la planilla.**

Disgusto por parte de los secretarios ya que dependen del tesorero para realizar el pago de salarios a los empleados pues este maneja todos los cálculos en un archivo de Excel lo cual ha llevado a problemas de gestión y comprensión de la información por parte de los trabajadores ya que no existe una definición clara del uso de este archivo.

### **Deficiencia en la definición de tareas según los roles.**

Los trabajadores se sienten disgustados al no existir un organigrama que especifique las tareas que deben realizar según el cargo que desempeñan en la empresa. Este problema afecta principalmente a los choferes.

**Carencia de un control de vacaciones para los empleados.**

Los secretarios administrativos cuentan con una pequeña boleta en donde apuntan la cantidad de días que los empleados solicitan vacaciones y luego se guarda en el archivo de empleados. Esto hace que sea difícil controlar cuantas vacaciones tiene cada empleado, además se presenta el riesgo de confundir las boletas.

**Ausencia de un control disciplinario para los empleados.**

Actualmente los administrativos no pueden gestionar adecuadamente las amonestaciones y felicitaciones para cada empleado por lo que no pueden generar reportes sobre estos y tampoco pueden catalogarlos debido a que esto se gestiona en un cuaderno indicando solamente si el empleado cuenta con alguna amonestación o felicitación.

**Ausencia de un control para gestionar los permisos de salida.**

Descontento de los secretarios administrativos ya que tienen que llamar al empleado para comprobar que el motivo por el cual el empleado se ausentó se debe a un permiso de salida. Esto se debe a que los empleados solicitan informalmente al encargado de aprobar los permisos y no se registra adecuadamente.

**Ausencia de un control para realizar los pagos a los empleados.**

Falta de un control para la gestión de la información en relación a los pagos, esto debido a que los datos de los trabajadores se encuentran en un único archivo lo cual provoca disgusto y confusión a los secretarios al momento de realizar o gestionar pagos salariales.

## **Objetivos**

### **Objetivo general**

Desarrollar un prototipo funcional para la gestión de recursos humanos de la Asociación de Desarrollo Integral de San Juan Puriscal.

### **Objetivos específicos**

- Analizar los requerimientos necesarios que serán implementados en el prototipo funcional.
- Diseñar el prototipo funcional.
- Desarrollar la programación del prototipo funcional para que este opere adecuadamente.
- Realizar las pruebas necesarias que comprueben que el prototipo presenta la menor cantidad de errores posibles.

## **Justificación**

Desde su fundación, ya hace unos 47 años de ello, la Asociación de Desarrollo Integral de San Juan Puriscal no ha implementado ningún tipo de sistema informático el cual les facilite las actividades sobre su recurso humano. Hoy día, la tecnología se ha convertido en un elemento fundamental, las empresas deben de ser rápidas y eficientes con la utilización de todos sus recursos, es por esta razón que la tecnología llega para resolver problemas y adaptarse a las necesidades.

El motivo principal de este proyecto, es desarrollar un prototipo funcional el cual brinde un apoyo tecnológico que le permita a la empresa resolver los principales problemas sobre sus recursos humanos, esto con el fin de acelerar sus procesos y permitir un mejor crecimiento.

La incorporación de este prototipo funcional por parte de la empresa, reflejará una mejora notable en la obtención de información sobre sus empleados, optimizando los procesos correspondientes, mejorando la calidad y garantizando que esta se encuentre actualizada en todo momento.

Por otro lado, brindará mayor tranquilidad y conformidad para aquellos empleados que interactúan mayormente con los procesos de recurso humano al saber que el prototipo ejecuta los cálculos necesarios para realizar los pagos regulares, así como los pagos de horas extras laboradas, entre otros. Adicionalmente, se espera dar seguridad al tener que pasar por un filtro de autenticación que determinará quien tiene permitido acceder y que restricciones posee para hacer uso de este.

A continuación, se da a conocer los distintos aspectos que se deben considerar para comprobar la viabilidad del prototipo que se pretende desarrollar.

### **Viabilidad técnica**

Es importante aclarar que la empresa no cuenta con ningún tipo de servidor el cual le permita alojar la base de datos y la aplicación, dicho esto, la implementación de este prototipo hará uso de servicios en la nube como lo es *Heroku*, este ofrece gran cantidad de herramientas y es realmente flexible conforme el sistema necesite escalar. Actualmente la empresa cuenta con varias computadoras de escritorio y una portátil en donde hacen uso de herramientas ofimáticas para realizar sus labores correspondientes. Los requerimientos de hardware que estas computadoras necesitan, realmente no son relevantes para el uso de este prototipo. No obstante, el único requerimiento que estas computadoras necesitan cumplir, es poseer un navegador web; por lo general este ya viene preinstalado.

Por otra parte, el computador en el cual se desarrolle el prototipo necesita cumplir los siguientes requerimientos mínimos para que este pueda ser completado sin ningún impedimento.

- Procesador Intel Core i3 a una frecuencia de 2GHz.
- 4 GB de memoria RAM.
- Tarjeta gráfica NVIDIA.
- Resolución de pantalla de unos 1024x768.

Estos requerimientos se establecieron basándose en los que las herramientas *Visual Studio Code* y *MySQL Workbench* especifican.

Ya que el entorno en el que se ejecutará la aplicación es web, es obligatorio tener una conexión a internet para su correcto funcionamiento. Toda página o aplicación web, necesita de un servidor en el cual estar alojado, en otras palabras, es donde los archivos que hacen parte del prototipo se encuentran a la espera de ser requeridos por el usuario, esto permite hacer uso del prototipo de manera remota desde cualquier dispositivo que posea internet y un navegador web. El servidor del servicio que se utilice debe de cumplir ciertas características mínimas para un correcto funcionamiento; debe poseer entre 512 MB y 1 GB de almacenamiento, y 512 MB de memoria RAM.

Por otra parte, Heroku posee la capacidad de ejecutar MySQL sin ningún problema mediante la utilización de lo que ellos llaman “Add-ons”, estos son diferentes herramientas y servicios que permiten desarrollar, ampliar y operar la aplicación mayor facilidad. Es importante tener en cuenta la región en la que se encuentre la base de datos, a medida que se aleja, la latencia sube por lo que las consultas hacia esta son más lentas. La región requerida será en el este de Estados Unidos y un almacenamiento 1 GB.

Vistos los puntos anteriores, se afirma que el computador donde se llevará a cabo el desarrollo del prototipo, y las capacidades con las que cuenta el servicio en la nube para implementar el mismo, cumplen los requisitos satisfactoriamente.

### **Viabilidad operativa**

Actualmente los empleados se encuentran laborando en la empresa, específicamente los administrativos, quienes harán uso más constante del prototipo, cuentan con conocimientos sobre el manejo de computadoras, programas ofimáticos e internet por lo que no hace falta de un entrenamiento para poder utilizar el prototipo.

Por otra parte, la forma en que se realizan las tareas no cambiará drásticamente haciendo que estas sean totalmente diferentes provocando que sea necesario realizar un entrenamiento, sino que estas se facilitarán automatizando gran parte de estas. Esto no quiere decir que se tenga que realizar una reducción de personal; la meta es agilizar las tareas y mejorar la eficiencia generando un ambiente más agradable para los empleados.

### **Viabilidad económica**

Gran parte del software y servicios necesarios para la elaboración de este prototipo como lo son *Visual Studio Code* y *MySQL Workbench*, cuentan con licencias gratuitas por lo que no agregarán ningún costo. Por otro lado, en cuanto al motor de base de datos *MySQL*, el cual también es gratuito, es necesario de una pequeña inversión mensual en el servicio llamado *Heroku*. Este servicio cuenta con un plan gratuito, pero debido a que el almacenamiento para este es tan solo de 5 MB; se decide utilizar el plan de 1 GB el cual se ajusta con las necesidades de la empresa.

Los costos de desarrollo están basados en el costo por día del desarrollador según el Ministerio de Trabajo y Seguridad Social al año 2020. Por otro lado, costos para adquisición de hardware o mobiliario se toman en cuenta ya que la empresa cuenta con estos.

A continuación, se muestran los costos de desarrollo e implementación más detalladamente.

Tabla 1. Costos de desarrollo.

Etapa	Cantidad de días	Costo por día	Costo total
Análisis	6	₡ 13,872.70	₡ 83,236.2
Diseño	6	₡ 13,872.70	₡ 83,236.2
Desarrollo	50	₡ 13,872.70	₡ 693,635
Pruebas	10	₡ 13,872.70	₡ 138,727
Total	72		₡ 998.834,4

Fuente: Elaboración propia.

Tabla 2. Costos de implementación.

Nombre	Costo mensual	Costo anual
Visual Studio Code	₡ 0	₡ 0
MySQL Workbench	₡ 0	₡ 0
Servicio para la aplicación	₡ 0	₡ 0
Servicio para base de datos	₡ 5,658.73	₡ 67,904.82
Total	₡ 5,658.73	₡ 67,904.82

Fuente: Elaboración propia.

## **Viabilidad legal**

Es de suma importancia tener en cuenta la existencia de leyes que protejan tanto al usuario como al desarrollador, por lo que se tomara en práctica aspectos legales como los que se citan a continuación.

- Ley de Protección de la Persona Frente al Tratamiento de sus Datos Personales (N°8968). La información que se obtenga y utilice en el caso de que el prototipo se implemente, será uso exclusivo del mismo y no se compartirá con terceros. Es importante aclarar que, durante el desarrollo de este proyecto, no se utilizarán datos reales.
- Ley de Derechos de Autor y Derechos Conexos (N°6683). Se mencionarán las respectivas citas bibliográficas y se hará referencia al final de este documento. Por otra parte, la empresa en donde se desarrollará este proyecto, no tiene permitido la distribución del prototipo sin el consentimiento del desarrollador y este será utilizado únicamente para los fines con el que fue creado.

Con lo anterior mencionado, se afirma con total certeza que el presente proyecto es legalmente viable para ser elaborado.

## **Proyecciones**

El objetivo principal de elaborar este proyecto, es desarrollar un prototipo funcional que sea eficiente para la empresa, esperando un efecto positivo en las labores que esta realiza sobre su recurso humano.

En este apartado se detallarán los diferentes alcances del proyecto para así delimitar hasta qué punto del desarrollo se pretende llegar. Entre los alcances establecidos, se exponen los siguientes.

## **Alcance funcional**

El prototipo funcional consta de varios módulos los cuales ayudarán a solventar los problemas que la empresa está afrontando actualmente. A continuación, se detalla cada uno de ellos.

### **Módulo de seguridad**

Este módulo permitirá la modificación, búsqueda y eliminación de aquellos usuarios que hagan uso del sistema. Como parte de las medidas de seguridad, este módulo incluirá un manejo controlado por roles para así limitar el acceso a los diferentes módulos del prototipo.

### **Módulo de mantenimiento**

**Empleados:** Gestionará la información personal relacionada a los empleados como el respectivo salario, pagos de horas extra, diferentes directrices, etc.

**Empleados temporales:** Esta sección será la encargada de controlar la información de los empleados temporales tal como las fechas en que trabajó y el cargo que este ejerció.

**Permisos de salida:** Tendrá la función de encargarse de controlar los permisos con o sin goce de salario, actualizar la información respectiva o cancelar el permiso en caso de no necesitarse.

**Amonestaciones y felicitaciones de empleados:** Permitirá controlar la información de conducta por parte de los empleados. Se podrá crear, clasificar y visualizar las amonestaciones y felicitaciones asignadas a los empleados.

### **Módulo de consultas**

**Información de los trabajadores:** En esta consulta se puede obtener datos relacionados a los trabajadores tales como el salario, la fecha de ingreso de la empresa y la información personal del empleado, entre otros.

**Trabajadores disponibles:** Se puede obtener todos aquellos trabajadores que se encuentran laborando en la empresa (incluidos los temporales).

**Vacaciones disponibles:** Se puede obtener la información relacionada a la cantidad de días vacacionales para cada empleado.

### **Módulo de reportes**

**Pago de salario:** En este reporte se mostrará la información del pago del salario detallando todas las deducciones y horas extras que se le hicieron al empleado.

**Control disciplinario:** Se mostrará la información de la cantidad y el detalle de las amonestaciones y felicitaciones que se le han realizado a cada empleado.

**Control de roles y tareas:** Se mostrarán y definirán las diferentes labores que desempeña el trabajador según el rol que cumpla en la empresa.

### **Módulo de gestión de planilla**

Este módulo permitirá gestionar el pago de la planilla, así como también pagar a los trabajadores las horas extras, viáticos y aplicar las deducciones necesarias a los salarios de los empleados de forma automática. También se llevará un registro de los salarios pagados para cada empleado. En el caso de los choferes es necesario hacerlo tanto para ruta normal como para las rutas turísticas (servicio de transporte público y transporte especial).

### **Módulo de gestión de vacaciones**

En este módulo se encargará de gestionar la información de las vacaciones de los empleados, tales como la fecha de salida y de ingreso, la cantidad disponible de vacaciones que tiene cada empleado, cantidad de vacaciones que ha utilizado hasta la fecha y estados en los que se encuentran dichas vacaciones (si se encuentran aprobadas o rechazadas por parte del supervisor).

### **Módulo de control de calidad de choferes.**

Este módulo se encarga de otorgar rutas a los trabajadores con base al número de recomendaciones obtenidas por parte de los turistas. De esta manera se premian las conductas apropiadas, así como conocer la opinión de los clientes.

### **Alcance tecnológico**

Para el desarrollo de este prototipo, se pretende que el entorno donde este opere sea web, esto quiere decir, que para que se pueda utilizar, es necesario tener un explorador web instalado en una computadora o cualquier otra plataforma que lo permita.

El desarrollo de la aplicación se llevará a cabo con las siguientes tecnologías y herramientas:

- JavaScript como lenguaje de programación seleccionado, tanto para el lado del cliente, como para el del servidor.
- MySQL versión 8.0.18 como motor de base de datos.
- Visual Studio Code versión 1.41.1 como herramienta de desarrollo. Esta herramienta es solamente un editor de texto, la razón por la cual fue seleccionado, es que no consume muchos recursos del computador y cuenta con gran cantidad de extensiones que facilitan el desarrollo.
- MySQL Workbench versión 8.0.19 (interfaz gráfica de usuario) como gestor de base de datos.

En cuanto a las características que posee el computador en donde se desarrollará el prototipo funcional, este cuenta con las siguientes:

- Computador portátil modelo Acer de la serie Aspire.
- Procesador Intel Core i5-5200U a una frecuencia de 2.20GHz.
- Tarjeta gráfica NVIDIA GeForce 940M.
- 8 GB de memoria RAM DDR3.
- 240 GB de almacenamiento en estado sólido.

### **Alcance metodológico**

Durante la elaboración del prototipo funcional, se trabajará bajo el modelo en cascada el cual consta de varias fases las cuales se comentan a continuación.

Como primera etapa, es fundamental tener en cuenta cuales son los problemas que se deben solucionar, esto es conocido como análisis de requerimientos. Consiste en realizar un

análisis de las necesidades del cliente para así determinar las características que ayudarán como guía para determinar las diferentes funcionalidades del software de acuerdo al cliente u organización donde se pretende realizar.

Una vez obtenidos estos requerimientos se puede proceder con la segunda etapa la cual consiste en diseñar los diagramas necesarios que permitan describir el comportamiento teórico del software, así como la interacción entre los diferentes usuarios y el sistema, cumpliendo con los requerimientos plasmados en la etapa anterior.

Ya definidas las primeras etapas, se procede con el diseño del prototipo, esta etapa corresponde en realizar el aspecto que tendrá el sistema, flujo entre las diferentes partes del sistema y dar a conocer si cumplirá con los requisitos mencionados al inicio.

En la fase de desarrollo, es donde se hará uso del lenguaje de programación y la base de datos, se construye el prototipo basándose en los diseños anteriores, tanto en los de usabilidad como en los de funcionalidad e interactividad.

Por último, pero no menos importante, la etapa de pruebas, consiste en procurar el perfecto funcionamiento del sistema minimizando al máximo los posibles errores que pueden ocurrir una vez el sistema sea entregado al usuario final.

Es importante destacar que cada fase puede tener un nombre distinto dependiendo de la fuente a la cual se consulte esta metodología, así como lo que se debe realizar en cada fase; por otra parte, no se llevarán a cabo los siguientes puntos:

- Elaboración de manuales, tanto de usuario como de procedimientos y técnico.
- Entrenamiento o capacitación a usuarios técnicos.
- Implementación del prototipo.

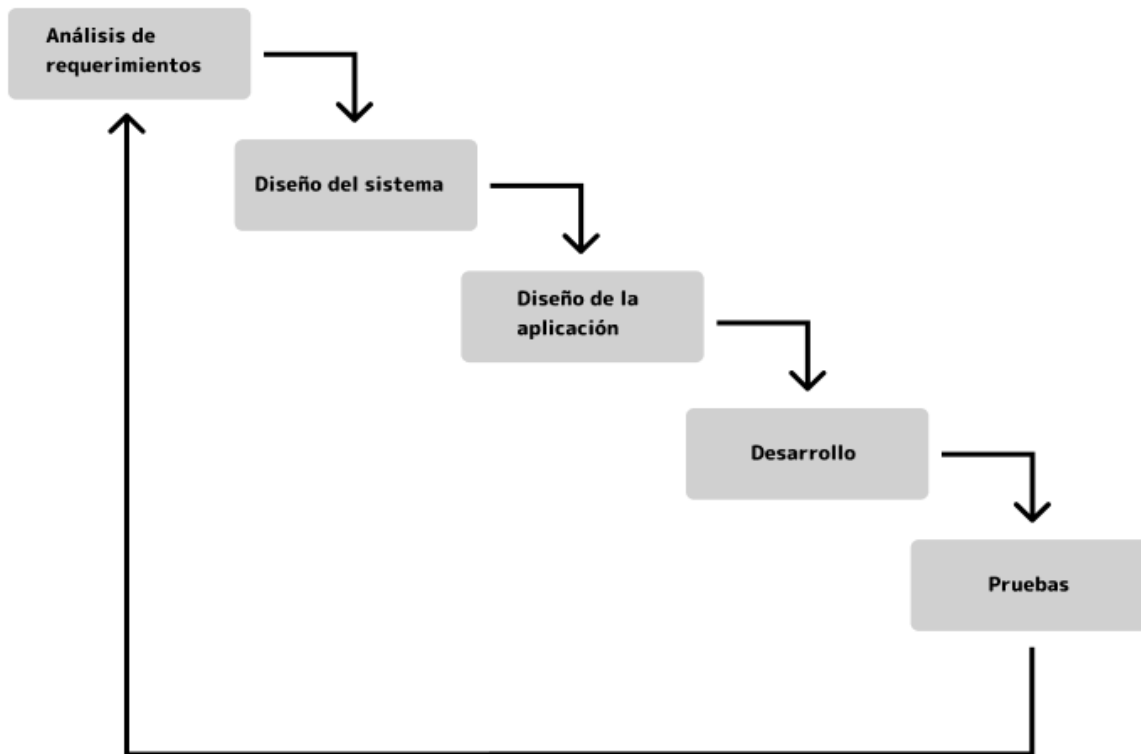
## CAPÍTULO II: MARCO REFERENCIAL

Para aquellos que no están muy familiarizados con conceptos sobre tecnología, específicamente sobre informática, resulta un poco confuso el entenderlos, es por ello que este apartado tiene como fin definir aquellos conceptos que se han utilizado anteriormente y los que se utilizarán con forme se avance en el proyecto facilitando su entendimiento.

Los **sistemas informáticos** tiene como propósito facilitar la recolección, administración, procesamiento, el almacenamiento y distribución de la información para satisfacer las necesidades de la organización, es un conjunto de partes interrelacionadas como lo indica (Alegsa, 2018) “Un sistema informático es un conjunto de partes o recursos formados por el hardware, software y las personas que lo emplean, que se relacionan entre sí para almacenar y procesar información con un objetivo en común.” (párr. 6). El objetivo principal del sistema, es facilitar las tareas diarias haciendo que estas sean más eficientes. Anteriormente las tareas eran realizadas de forma manual, pero gracias a la implementación de estos sistemas, ahora estas se han automatizado.

Es importante a la hora de realizar un sistema, tener en cuenta que existen diferentes **metodologías** para el desarrollo de software las cuales ayudan a estructurar, planificar y controlar este proceso. Anteriormente se mencionó que se implementará el modelo de cascado, o ciclo de vida como se conoce comúnmente. En relación con los modelos en cascada, (Pacha, 2015) indica que “es el enfoque metodológico que ordena rigurosamente las etapas del proceso para el desarrollo de software, de tal forma que el inicio de cada etapa debe esperar al término de la etapa anterior.” (párr. 1). Se puede entender también como un método de desarrollo lineal y secuencial que consta de varias fases definidas por diferentes tareas y objetivos. En la figura No. 1 se puede apreciar las fases de componen esta metodología.

Figura 1. Metodología en cascada.



Fuente: Elaboración propia

Para crear un sistema o programa informático, es necesario saber utilizar un **lenguaje de programación**. Este lenguaje informático es utilizado por los programadores para desarrollar programas que se componen de un conjunto de instrucciones que posteriormente una computadora se encarga de ejecutar de una manera especial en la que los seres humanos no entendemos, ampliando la definición anterior (Ayala, 2017) indica que “Un lenguaje de programación es un lenguaje diseñado para describir el conjunto de acciones consecutivas que un equipo debe ejecutar. Por lo tanto, un lenguaje de programación es un modo práctico para que los seres humanos puedan dar instrucciones a un equipo”. (párr. 1).

Existen varios **entornos para desarrollar un programa o sistema**. En el caso de este prototipo, el entorno en el cual se desarrollará será web, por lo tanto, una aplicación web es un programa informático que utiliza navegadores web y tecnologías web para realizar tareas a través de internet, (Alegsa, 2018) indica que “Una aplicación web es cualquier aplicación que es accedida vía web por una red como internet o una intranet”. (párr. 1).

Además, “el término también se utiliza para designar aquellos programas informáticos que son ejecutados en el entorno del navegador (por ejemplo, un applet de Java) o codificado con algún lenguaje soportado por el navegador (como JavaScript, combinado con HTML); confiándose en el navegador web para que reproduzca (renderice) la aplicación”. (párr. 2). En palabras simples, el proceso de renderización de una página web consiste en acciones que el navegador realiza, procediendo este a analizar y procesar el HTML y demás para finalmente visualizar dicha página.

Para este prototipo, se optará por la utilización del lenguaje de programación llamado **JavaScript** el cual es considerado una de las primeras opciones para el desarrollo web ya que cuenta con gran cantidad de bibliotecas, es rápido y no necesita ser compilado para ser ejecutado, pero si de un intérprete, en este caso un navegador web. Al respecto, (Pelhon, 2019) agrega lo siguiente:

JavaScript es uno de los lenguajes de programación más populares para el desarrollo web, ejecutado del lado del cliente (esta expresión significa que los scripts o programas son ejecutados en el navegador del usuario), y funciona como una extensión del código HTML. (párr. 7).

Otro aspecto muy importante es el uso de **Type Script** para el desarrollo de este prototipo funcional. Este es un super conjunto o super set, como se conoce comúnmente, de JavaScript y sus diferentes especificaciones este se compila a JavaScript puro y es considerado como un lenguaje de programación basado en dicho lenguaje. Como menciona (Hernandez, 2018):

Type Script es un super set de JavaScript. Decimos que una tecnología es un super set de un lenguaje de programación, cuando puede ejecutar programas de la tecnología, Type Script en este caso, y del lenguaje del que es el super set, JavaScript en este mismo ejemplo. En resumen, esto significa que los programas de JavaScript son programas válidos de Type Script, a pesar de que Type Script sea otro lenguaje de programación.

Adicionalmente, para elaborar adecuadamente una aplicación web, es necesario utilizar **HTML** y **CSS** junto con JavaScript. HTML es un lenguaje de marcado de hipertexto por sus siglas en inglés “Hyper Text Markup Language”. y cuyo propósito es describir la

estructura básica de la página o aplicación web y organizar la forma en la que esta mostrará su contenido. Por otra parte, CSS, hojas de estilos en cascada por sus siglas en inglés “Cascading Style Sheets” que no es un lenguaje no programación sino un lenguaje que especifica como el contenido establecido con HTML en la página o aplicación web, es presentado al usuario, es decir, que aspecto tendrá todo el contenido para que este sea visualmente más agradable y organizado.

HTML y CSS por sí solos no aportan mucha funcionalidad y dinamismo a la aplicación desde el lado del cliente, es decir, desde el navegador, por esta razón que se hará uso de **VueJS**, según el creador (You, 2017) “Vue es un *framework* progresivo para construir interfaces de usuario. A diferencia de otros *frameworks* monolíticos, Vue está diseñado desde cero para ser utilizado incrementalmente.” (párr. 1) Para entender un poco mejor que es Vue, es necesario entender dos palabras que menciona Evan You. Primeramente, Vue es un *framework*, esto quiere decir que es un conjunto de herramientas para desarrollo que trabajan en un proyecto completo bajo ciertas reglas, en palabras más simples, el *framework* define la forma en que se debe desarrollar el proyecto.

Por otra parte, Vue es progresivo en este contexto significa que puede ir creciendo junto con el proyecto, es decir, Vue no necesariamente es para proyectos pequeños o para proyectos grandes, se puede comenzar desde cosas muy simples hasta avanzar a las más complejas, esto permite que se integre en un proyecto existente de forma incremental según los requisitos para así construir interfaces de usuario adaptables y sofisticadas.

Por ende, podemos entender a Vue como un conjunto de herramientas y funciones que permiten desarrollar páginas y aplicaciones web de una manera más simple, agrega más dinamismo a estas permitiendo que la obtención de información desde la base de datos de forma dinámica.

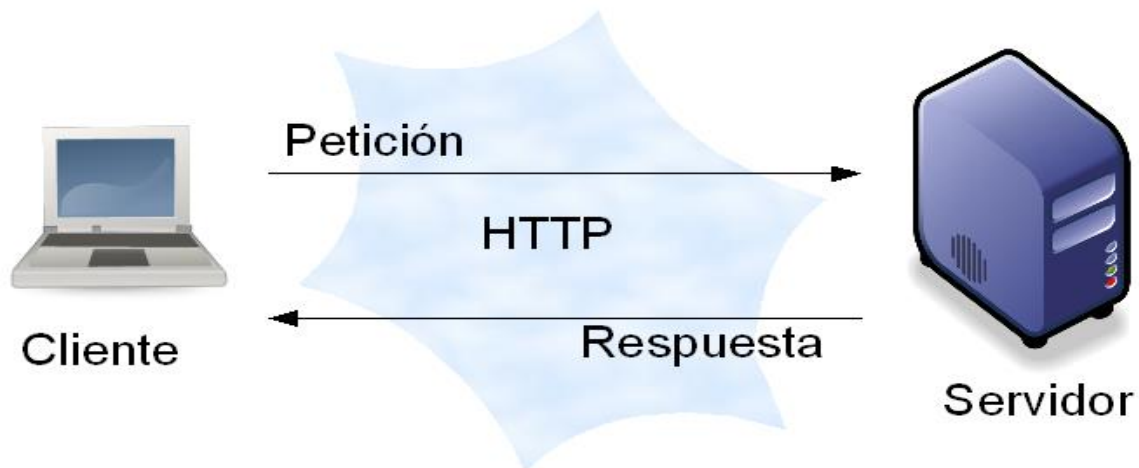
Con el continuo crecimiento y apoyo con el que cuenta JavaScript, ha permitido que este lenguaje se adapte muy bien a diferentes **arquitecturas de software**, es decir, la forma en que se estructura y el funcionamiento e interacción que tendrán las partes del software, como explica (Apiumhub, 2017):

Es un conjunto de padrones [sic] a través de los cuales se define como tendrá que ser el código fuente. Marca unas pautas, objetivos y restricciones teniendo en cuenta los

posibles riesgos a los que se puede enfrentar el desarrollo y cómo superarlo. También es necesario tener presente como quieres que interactúen las diferentes partes del sistema. (párr. 1).

Considerando lo anterior, el prototipo estará basado en la arquitectura cliente servidor donde se define al servidor como la parte encargada de recibir la solicitud por parte del cliente, también es el que procesa la información y la entrega al cliente a través de una API REST o interfaz de programación de aplicaciones por sus siglas en inglés “Application Programming Interface” (se definirá más adelante), que cuenta con la capacidad de enviar y consultar información. Por otra parte, el cliente es el usuario activo que mediante el uso de un navegador web, hará uso de la aplicación, solicitando información al servidor para que este posteriormente la retorne. Dicha arquitectura se puede apreciar en la figura No. 2

Figura 2. Arquitectura cliente servidor.



Fuente [https://2.bp.blogspot.com/-](https://2.bp.blogspot.com/-KspM4fIVaqc/T1Fv9iShNJI/AAAAAAAAADY/EqRMTZjmxk0/s1600/cliente-servidor-http.png)

[KspM4fIVaqc/T1Fv9iShNJI/AAAAAAAAADY/EqRMTZjmxk0/s1600/cliente-servidor-http.png](https://2.bp.blogspot.com/-KspM4fIVaqc/T1Fv9iShNJI/AAAAAAAAADY/EqRMTZjmxk0/s1600/cliente-servidor-http.png)

JavaScript no es capaz de ejecutarse en un servidor, otros lenguajes cumplen este propósito sin ningún problema, por esta razón, es necesario utilizar **Node.js** lo que permite construir toda la lógica necesaria para el cliente y el servidor con un solo lenguaje de programación. Citando a (Cabana, 2017) “Node.js es un entorno JavaScript que nos permite

ejecutar en el servidor, de manera asíncrona, con una arquitectura orientada a eventos y basado en el motor V8 de Google.” (párr. 1). El motor V8 de Google permite ejecutar código JavaScript fuera de un navegador web, es esta tecnología la que hace posible que se ejecute código en un servidor, el motor traduce el código JavaScript directamente a código máquina, es decir, código que las computadoras pueden entender.

Anteriormente se mencionó que la comunicación entre el cliente y un servidor se realiza mediante una **API REST**. En primera instancia es necesario entender dos conceptos muy importantes, como indica (Torres, 2018) “Una **API** es un conjunto de procesos, métodos y subrutinas que cumplen una o muchas funciones con el fin de ser utilizadas por otro software”. (párr. 4).

“**REST** deriva su nombre de **Representational State Transfer**. Describe un estilo particular de creación de **API** diseñado para ser liviano, fácil de usar y adecuado para servicios de gran volumen”. (Torres, 2018, párr. 17),

Adicional a esto, se puede decir que una API REST se compone de dos partes, la interfaz de programación de aplicaciones o API por sus siglas en inglés. Esta interfaz permite que las aplicaciones se conecten para compartir información entre sí, por otra parte, REST es una arquitectura para las APIs, se conecta vía web con el protocolo HTTP y permite acceder a la información con identificadores únicos.

Todo el tráfico de información que este prototipo realice necesita ser almacenado en algún lugar, es por ello que el uso de una **base de datos** es de suma importancia debido a que no existe tecnología sin datos y sería imposible crear este prototipo sin una base de datos. Estas bases permiten almacenar gran cantidad de información y se encuentra organizada de tal forma que su acceso es mucho más sencillo. Al respecto, (Vásquez, 2018) menciona que:

Una base de datos es un conjunto de información almacenada, relacionada y estructurada, es una información, que puede ser mostrada, cambiada o eliminada posteriormente mediante usuarios que tienen privilegios para realizar ciertas funciones a la base de datos. Las bases de datos están compuestas por tablas, las tablas están compuestas por cerdas, filas y columnas, que es donde se almacena la información. (párr. 1).

En otras palabras, se puede entender a las bases de datos como un conjunto de información almacenada y consultada sistemáticamente. Estas bases de datos a su vez necesitan un sistema gestor de base de datos el cual es el software encargado de gestionar base de datos. Con estos sistemas se puede crear la estructura de la base de datos, escribir y consultar información, asegurar la integridad de los datos y controlar el acceso a esta información, es decir, definir los permisos de quien puede y quien no puede visualizar o acceder a esta información.

Existen diferentes tipos de modelos para las bases de datos, para este proyecto, se procederá a utilizar el **modelo relacional** el cual es el más común hoy en día, (Charte, 2015) lo define en su libro de la siguiente manera: “El modelo relacional de bases de datos está basado en la teoría de conjuntos habitual en el algebra matemática, siendo SQL el lenguaje utilizado para componer las expresiones que operan sobre dichos conjuntos”. (p.31). Este modelo ordena los datos en tablas, cada tabla se compone de columnas y filas y relaciones entre ellas. Estas relaciones son representadas por el modelo y pueden ser relaciones de uno a uno, uno a muchos y muchos a mucho. El objetivo del modelo relacional, como su nombre lo indica, es crear relaciones entre la información que se encuentra en la base de datos.

Para utilizar una base de datos es necesario tener conocimiento de un lenguaje especial, dependiendo de la base de datos que se utilice este lenguaje puede cambiar, normalmente *SQL* se toma como primera opción y para el presente proyecto no será la excepción. *SQL* o *Structured Query Language* cuyo significado corresponde a “Lenguaje de Consulta Estructurado”, permite crear comandos para obtener, modificar o eliminar el contenido que se encuentra en la base de datos.

Entre las ventajas que posee este lenguaje de consultas, se encuentran los **procedimientos almacenados**, estos son un conjunto de comandos SQL que pueden almacenarse en la base de datos para posteriormente ser llamados y ejecutados, (Charte, 2015) explica que “Un procedimiento almacenado es una secuencia de sentencias a las que se asigna un nombre, de tal manera que es posible ejecutarlas usando dicho nombre como si de cualquier otra orden SQL se tratase”, (p.253).

Los procedimientos almacenados son muy útiles cuando se desea aumentar la seguridad, estos tienen que recibir parámetros específicos y cantidades exactas o de lo

contrario fallan, son un buen método para evitar a ataques, también cuando las consultas resultan ser muy complejas, es más viable crear un procedimiento almacenado y al ser ejecutados en un único lote de código, reduce significativamente el tráfico de red entre el cliente y el servidor.

Todos los conceptos definidos anteriormente sobre tecnología corresponden a el funcionamiento interno bajo el que trabajará el prototipo funcional de **recursos humanos**, este se encarga de gestionar la información del personal que se encuentra laborando en la empresa, por esta razón (García I. , 2017) define el recurso humano como “el concepto utilizado para nombrar al departamento que hay en las empresas para realizar labores de selección, contratación, formación y el empleo de las personas que son necesarias para integrarse en la plantilla y conseguir de esta manera los objetivos fijados”. (párr. 1). Es importante tener en cuenta que los recursos humanos son indispensables para cualquier empresa, ayuda a las personas que trabajan en la empresa a seguir desarrollándose, formándose y creciendo, además, permite contratar mejores trabajadores y asignarlos a cada puesto disponible.

Uno de los apartados principales para este proyecto como para la administración del recurso humano es el cálculo de **planilla o nómina**, esto consiste en un “registro financiero que realiza la empresa para llevar los salarios de sus empleados, las bonificaciones y las deducciones”. (Riquelme, 2017, párr. 4).

Es decir, la planilla o nómina es un instrumento administrativo importante que permite calcular el salario de los empleados de manera ordenada para proceder con el pago y llevar un control más preciso de la situación financiera de la empresa.

Al realizar los cálculos salariales para cada empleado hay que tener en cuenta que se tiene que aplicar las **cargas sociales**, estas son deducciones que se aplican a todas las personas que empiezan a laborar en el país, esto con el fin de contribuir a la Caja Costarricense del Seguro Social.

<b>Caja Costarricense de Seguro Social</b>			
<b>Concepto</b>	<b>Patrono</b>	<b>Trabajador</b>	<b>Porcentaje</b>
<b>SEM</b>	9.25%	5.50%	14.75%
<b>IVM</b>	5.25%	4.00%	9.25%
<b>TOTAL CCSS</b>	<b>14.50%</b>	<b>9.50%</b>	<b>24.00%</b>
<b>Recaudación Otras Instituciones</b>			
<b>Cuota Patronal Banco Popular</b>			0.25%
<b>Asignaciones Familiares</b>			5.00%
<b>IMAS</b>			0.50%
<b>INA</b>			1.50%
<b>TOTAL OTRAS INSTITUCIONES</b>			<b>7.25%</b>
<b>Ley de Protección al Trabajador (LPT)</b>			
<b>Aporte Patrono Banco Popular</b>			0.25%
<b>Fondo de Capitalización Laboral</b>			3.00%
<b>Fondo de Pensiones Complementarias</b>			0.50%
<b>Aporte Trabajador Banco Popular</b>			1.00%
<b>INS</b>			1.00%
<b>TOTAL LPT</b>			<b>5.75%</b>
<b>TOTAL</b>			<b>37.00%</b>

Figura 3. Cargas sociales

Fuente: Caja Costarricense de Seguro Social

En la figura anterior, no todos los campos afectan directamente al trabajador, algunos de estos deben ser cotizados por el patrono como es el caso de la sección de Recaudación de otras Instituciones que incluye la cuota patronal al Banco Popular, asignaciones familiares, IMAS e INA. En la sección de Ley de Protección al Trabajador, se aplican todos los campos al patrono a excepción del Aporte trabajador Banco Popular, este como su nombre lo indica, corresponde al trabajador.

SEM y IVM son siglas para Salario por concepto de Enfermedad y Maternidad y por concepto de Invalidez y Muerte, respectivamente.

Adicionalmente a estos cargos, los empleados que laboren en este país deben pagar un impuesto cuando sus ingresos salariales superan ciertas cantidades ya establecidas. En la siguiente tabla se muestran las cantidades y los porcentajes a pagar sobre su ingreso bruto.

Tabla 3. Impuestos sobre la renta

Ingreso bruto mensual	Tasa
Hasta <b>₡840.000,00</b>	No sujeto
Sobre el exceso de <b>₡840.000</b> hasta <b>₡1.233.000</b>	10%
Sobre el exceso de <b>₡1.233.000</b> hasta <b>₡2.163.000</b>	15%
Sobre el exceso de <b>₡2.163.000</b> hasta <b>₡4.325.000</b>	20%
Sobre el exceso de <b>₡4.325.000</b>	25%

Fuente: Recopilado de Ministerio de Hacienda

Por otro lado, las vacaciones son un tema importante que hay que considerar, como indica el (Ministerio de Trabajo y Seguridad Social , 2020) “todo trabajador tiene derecho a vacaciones anuales remuneradas, cuyo mínimo se fija en dos semanas por cada cincuenta semanas de labores continuas, al servicio de un mismo patrono”. (pág. 48).

Los sistemas de recursos humanos son de suma importancia debido a que cumplen una labor muy importante dentro de cualquier empresa, en estos sistemas se trabaja almacenando y manipulando información de los empleados, que a su vez representan un valioso activo para la empresa. Sin restarle importancia, también se encuentra el trato al cliente, al brindar servicios de transporte público y privado, entre otros, es importante considerar un acceso rápido a la información para una mejor y rápida toma de decisiones como puede ser el detectar y aplicar conductas inadecuadas a los trabajadores responsables.

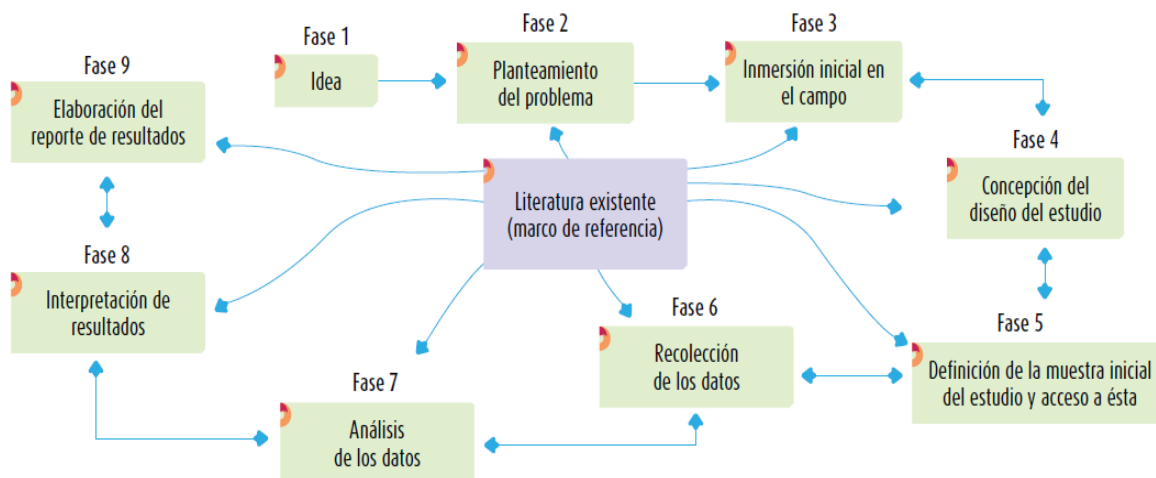
## CAPÍTULO III: MARCO METODOLÓGICO

En el presente capítulo se mostrará la metodología que se utilizará para la realización de este proyecto con el propósito de definir como se obtendrán los datos y como se llevará a cabo la investigación. Existen varios métodos para enfocar la investigación, pero solo se seleccionará el que mejor se ajuste a la investigación y permita completarla satisfactoriamente.

### Enfoque cualitativo

Este enfoque recolecta y analiza datos sin medición numérica para descubrir preguntas de investigación durante el proceso interpretación. (León, 2017) señala que “Se basa en el análisis no estadístico de datos para luego formular propuestas de interpretación, es un enfoque más subjetivo y amplio del estudio de problemáticas”. (párr. 5).

Figura 4. Fases o etapas del enfoque cualitativo.



Fuente: <https://4.bp.blogspot.com/->

KNBenJEMzQ8/Wd8ZSwmGRHI/AAAAAAAAAEc/KIiQl7\_4JyA\_4s4UXa5cI2Q6mevo9sHKACEwYBhg  
L/s1600/enfoque%2Bcualitativo.png

### Enfoque de la investigación

El enfoque seleccionado para este proyecto corresponde al cualitativo con el cual se pretende realizar entrevistas al personal administrativo y visitas a la instalación de la empresa para observar que procesos realizan y como los llevan a cabo. Al utilizar métodos de

recolección de información, permite una incursión a la experiencia que poseen los participantes seleccionados.

### **Investigación descriptiva**

Este tipo de investigación busca obtener las características del tema a investigar, definirlo, seleccionar la técnica de recolección de datos y fuentes para consultar, como indica (Universia, 2017) “La investigación descriptiva es la que se utiliza, tal como el nombre lo dice, para describir la realidad de situaciones, eventos, personas, grupos o comunidades que se estén abordando y que se pretenda analizar”. (párr. 8).

### **Método de la investigación**

Para efectos de este proyecto, se seleccionará el tipo de investigación descriptiva debido a que entre sus características se encuentra la capacidad de recolectar información con diferentes técnicas sobre los diversos conceptos relacionados a la investigación que se está llevando a cabo; dicho esto, se pretende observar y detallar los procedimientos de la empresa sobre recursos humanos para obtener una visión más clara y no perder de vista las problemáticas de la empresa.

### **Fuente de información primaria**

Según (Maranto & González, 2015) en su investigación sobre fuentes de investigación “Este tipo de fuentes contienen información original es decir son de primera mano, son el resultado de ideas, conceptos, teorías y resultados de investigaciones. Contienen información directa antes de ser interpretada, o evaluado por otra persona”. (p. 3).

Para esta fuente se utilizarán entrevistas para el personal administrativo de la Asociación de Desarrollo Integral de San Juan Puriscal, así como observación con el propósito de conocer el modo en que realizan los procesos.

### **Fuente de información secundaria**

Este tipo de fuentes son las que ya han procesado información de una fuente primaria. El proceso de esta información se pudo dar por una interpretación, un análisis, así como la extracción y reorganización de la información de la fuente primaria. (Maranto & González, 2015, pág. 3).

Aquí se consultarán libros de texto y páginas web acerca de bases de datos (específicamente sobre MySQL), programación en el lenguaje JavaScript y recursos humanos.

### **Fuente de información terciaria**

Este tipo de fuente envía al investigador a fuentes de información primarias o secundarias ya que suelen ser fuentes de referencias. (Maranto & González, 2015) indican que “Este tipo de fuentes son las que recopilan fuentes de información primarias o secundarias. Estas fuentes son utilizadas para buscar datos o para obtener una idea general sobre algún tema”. (p. 3).

### **Variable conceptual**

Según (Arcia, 2010) en su investigación “Son definiciones de diccionario, de libro especializado y describen la esencia o las características reales de un objeto o fenómeno”. (párr. 5). Es decir, ayuda a entender la definición del concepto “técnico” relacionado a la variable.

### **Variable operacional**

Su objetivo es obtener la mayor cantidad de información posible de la variable seleccionada, como indica (Arcia, 2010) “Se trata de describir cuidadosamente los criterios de evaluación y medición de la información que se recoja”. (párr. 8).

### **Variable instrumental**

Según (Arcia, 2010) “Aquí se aclara como se estudiará la variable que se acaba de definir, los medios o instrumentos para recoger la información”. (párr. 6). Es decir, corresponde a la herramienta que se utilizará para obtener la información.

### **Variables requeridas**

Tabla 4. Definición de variables

Objetivo específico	Variable	Variable conceptual	Variable operacional	Variable instrumental
---------------------	----------	---------------------	----------------------	-----------------------

Definir los principales requerimientos del prototipo	Requerimientos del prototipo	Según (Alegsa, 2016) “Los requerimientos son declaraciones que identifican atributos, capacidades, características y/o cualidades que necesita cumplir un sistema (o un sistema de software) para que tenga valor y utilidad para el usuario.” (párr. 2)	Reuniones con el personal administrativo de la Asociación de Desarrollo	Encuesta y observaciones
Diseñar la interfaz del prototipo	Interfaz gráfica de usuario (aspecto visual) para el prototipo	Es la elaboración de las distintas pantallas mediante las cuales permite a una persona utilizar el prototipo funcional	Realización del diseño gráfico para el prototipo funcional con herramientas especializadas para este propósito	Figma

Programar el prototipo en base a los requerimientos establecidos	Prototipo funcional	Prototipo funcional en entorno web para los recursos humanos de la Asociación de Desarrollo	Escritura del código de programación y elaboración de la base de datos con las herramientas correspondientes	Visual Studio Code y MySQL Workbench
Realizar las pruebas necesarias para comprobar el funcionamiento del prototipo	Pruebas para el prototipo funcional	Consiste en verificar el comportamiento del prototipo para detectar y corregir los errores con el fin de que el funcionamiento de este sea más estable	Elaboración de pruebas necesarias para corroborar un correcto funcionamiento	Visual Studio Code Insomnia

Fuente: Elaboración propia

### **Población**

Según (González, 2015) “En términos estadísticos, población es un conjunto finito o infinito de personas, animales o cosas que presentan características comunes, sobre los cuales se quiere efectuar un estudio determinado”. (párr. 3). Esto quiere decir que se utilizará un conjunto de personas con características similares las cuales serán objeto de estudio. Para este proyecto, la población que se pretende estudiar corresponde a los participantes que conforman la Asociación de Desarrollo Integral de San Juan Puriscal componiéndose de alrededor de doce personas, incluyendo choferes, secretarios y junta directiva.

## Muestra

Por otra parte, la muestra se puede entender por un subconjunto que representa una parte de la población que se pretende estudiar con el fin de recolectar datos. Existe una fórmula la cual permite calcular la muestra, en la siguiente figura se puede apreciar esta con mayor exactitud.

Figura 5. Fórmula para calcular la muestra.

$$n = \frac{K^2 p q N}{E^2 (N-1) + K^2 p q}$$

Fuente: <https://i1.wp.com/www.estudiosmercado.com/wp-content/uploads/2015/06/F%C3%B3rmula-tama%C3%B1o-muestral-poblaci%C3%B3n-finita.png>

El significado de las variables utilizadas en la figura anterior corresponde a:

- n = tamaño de la muestra.
- N = tamaño de la población.
- K= nivel de confianza.
- p = proporción esperada.
- q = probabilidad de fracaso.
- E = precisión (margen de error)

En la siguiente tabla, se muestra los valores de **K** recomendados según el nivel de confianza deseado.

Tabla 5. Niveles de confianza.

Nivel de K	1.28	1.44	1.65	1.96	2.58
Nivel de confianza deseado	80%	85%	90%	95%	99%

Fuente: Elaboración propia.

A continuación, se procede a aplicar la fórmula, para ello hay que considerar los siguientes valores:

- $N = 12$
- $K = 95\%$  cuyo equivalente corresponde a 1.96
- $p = 50\%$  cuyo equivalente corresponde a 0.5
- $q = 0.5$
- $E = 0.05$

$$n = \frac{(1.96)^2(0.5)(0.5)(12)}{(0.05)^2(12 - 1) + (1.96)^2(0.5)(0.5)}$$

Al realizar la fórmula, se obtiene como resultado para la muestra un total de 11,66, se tomará como muestra a 12 personas ya que no se puede obtener fracciones de individuos.

### **Instrumentos de recolección de datos**

Estos instrumentos consisten diferentes técnicas que permiten al investigador realizar una mejor recopilación de la información con un objetivo específico. Para obtener la información apropiada, es importante conocer las características de estas técnicas y enfocarse en los problemas que presenta la empresa.

A continuación, se presentan los métodos que serán utilizados para la recolección de información.

### **Entrevista**

Según (Alvarez-Gayou Jurgenson, 2003) “En la investigación cualitativa, la entrevista busca entender el mundo desde la perspectiva del entrevistado, y desmenuzar los significados de sus experiencias”. (p. 109). Esta técnica es una forma de interacción social. Su propósito es establecer una conversación con el entrevistado permitiendo obtener información más completa sobre la empresa y sus necesidades.

## **Observación**

Permite reforzar la información obtenida durante la entrevista, se conocen mejor los procesos realizados en la empresa valorando los mismos para posteriormente ser incluidos en el prototipo. Al respecto, (Mdpstranieri, 2015) menciona que:

Es una técnica que consiste en examinar atentamente el fenómeno, hecho o caso, y en la cual el investigador realiza su inspección y estudio tal como son o tienen lugar espontáneamente, mediante el empleo de sus propios sentidos, con o sin ayuda de aparatos técnicos, toma información y la registra para su posterior análisis. (párr. 5)

Normalmente la comunicación en esta técnica es mínima, el investigador procura estar alerta a los sucesos que van ocurriendo, interpretándolos con el fin de obtener un conocimiento más completo de lo que observa.

## **Proceso para la recolección y análisis de datos**

Para la parte del proceso de recolección y análisis de la información, se llevará a cabo una visita a la Asociación de Desarrollo Integral de San Juan Puriscal, donde se realizará una entrevista a los funcionarios con el fin de profundizar en la problemática actual, sus procesos y posibles solicitudes en cuanto a la realización del prototipo, esta entrevista será aplicada a parte de la junta directiva y secretarios administrativos. La información recopilada pasara por un proceso de análisis donde se seleccionará lo más relevante y se aplicaran diferentes técnicas para así interpretar conclusiones con base en los resultados obtenidos.

Posteriormente, se realizará una segunda visita, en esta ocasión el objetivo es espectral las actividades llevadas a cabo por los secretarios administrativos ya que ellos poseen mayor interacción con los procesos de la organización. Esto permitirá reforzar la información obtenida durante la entrevista.

Toda la información analizada será mostrada por medio de diagramas para evidenciar los resultados obtenidos.

## CAPITULO IV: ANÁLISIS DE RESULTADOS

A continuación, se expondrán y analizarán los resultados obtenidos mediante los instrumentos de recolección de información establecidos en el capítulo anterior. El presente apartado pretende detallar la situación actual de la empresa para una mejor comprensión sobre el desarrollo del prototipo, además, permitirá apoyar la investigación generando un punto de vista más claro y amplio.

### **Resultados de la entrevista**

La entrevista realizada a los empleados y junta directiva de la empresa inicia con la **primera pregunta** de la misma y cuyo objetivo es conocer si han adquirido algún sistema informático de Recursos Humanos, lo cual afirman que desde su fundación la empresa no ha contado con ningún tipo de sistema similar y que todos los procesos son realizados de manera manual. También comentaron que utilizan programas ofimáticos para facilitar y agilizar un poco los procesos actuales.

Por otra parte, miembros de la junta directiva comentaron que actualmente cuentan con un gran interés en implementar un sistema de este tipo que permita facilitar las tareas frecuentes relacionadas al recurso humano, agregan también que esta iniciativa no ha sido tomada muy en serio debido a la inversión que representa un sistema de esta escala.

La **segunda pregunta** consiste en conocer cuales procesos de Recursos Humanos realizan y cuales consideran que presentan una dificultad elevada, esto con el objetivo de determinar cuáles necesitan mayor análisis.

Los entrevistados comentan que aplican los procesos comúnmente conocidos, como lo son el cálculo de la planilla para poder realizar pagos, gestión de roles, vacaciones, horas extras e información imprescindible de los empleados, entre otras cosas básicas.

Por otra parte, mencionaron que ellos no sienten que los procesos sean muy difíciles, pero si muy poco eficientes y enfatizaron mucho en el proceso para calcular la planilla. Expresaron que ese proceso en específico es muy dependiente de una sola persona y que genera muchos atrasos para proceder a realizar los pagos a los empleados.

El objetivo de la **tercera pregunta** consiste en comprender el propósito que desean para el módulo de control de calidad de choferes.

Indican que dicho módulo lo consideran importante debido a que su principal fuente de ingresos son los servicios de transporte público y turístico, y en los cuales los choferes interactúan con gran cantidad de personas. Por este motivo, las pequeñas encuestas realizadas para calificar el servicio que brindan los choferes, son de suma importancia para que no disminuya el tráfico de usuarios. Buscan que este módulo permita gestionar las conductas positivas y negativas, además, dependiendo de la calificación obtenida en la encuesta, asignar una ruta o servicio en específico e incluso que lleve un registro de la calidad del servicio para posteriormente otorgar algún incentivo para aquellos choferes que mantengan una buena calificación durante un periodo de tiempo.

El objetivo de la **cuarta pregunta** consiste en conocer la funcionalidad que esperan del módulo de gestión de planilla.

Comentan que, mediante la automatización de este proceso, ellos esperan que el cálculo de planilla pueda realizarse por más de una persona para así eliminar los atrasos en el pago de los salarios. Agregan también que, debe de contar con un registro el cual les permita calcular el aguinaldo, además, para los choferes se necesita incluir el cálculo del servicio turístico.

La **quinta pregunta** tiene como objetivo identificar los rubros que utilizan para el cálculo de planilla.

Mencionan que horas extras, deducciones por ley, incapacidades, jornadas de trabajo y aguinaldos, son algunos de ellos. Además, expresan que constantemente deben actualizar información relacionada al pago salarial que se encuentra en el expediente, y que, si en algún rubro no llegan a coincidir estos datos con los que se encuentran en el archivo de Excel, genera retrasados y por ende molestias.

Por otra parte, comenta el encargado del cálculo de planilla que es común que el pago de horas extras se solicite inmediatamente por algunos empleados porque no se registran adecuadamente dichas horas, esto genera un descontrol en el pago total ya que es necesario verificar con exactitud la cantidad de horas extras.

Por último, la **pregunta seis**, tiene como objetivo conocer los requerimientos que solicitaría para un sistema de información, ya que es la primera vez obtendrían uno.

Definen un sistema que facilite las tareas que actualmente hacen, siendo estas lo más automatizadas posibles y que resuelvan satisfactoriamente los problemas mencionados anteriormente. Mencionan también un sistema seguro y eficiente, que el cálculo de planilla sea más simple y exacto, que incluya toda la información de los empleados y que el sistema sea sencillo de utilizar.

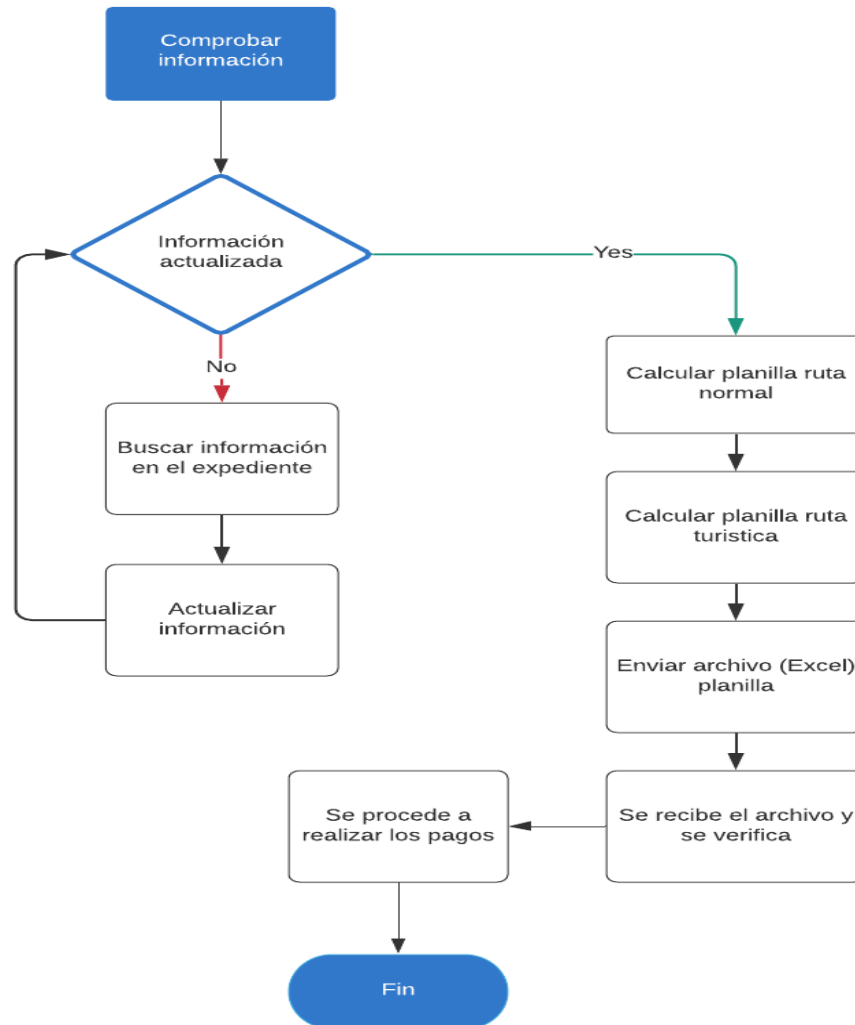
### **Resultados de la observación**

En el siguiente apartado se mostrará la información obtenida durante la etapa de observación mediante la utilización de diagramas, esto con el propósito de simplificar, comprobar y demostrar los problemas mencionados anteriormente, así como conocer los procesos realizados por la empresa.

Para la realización de la observación, se procedió a incursionar en el ambiente de trabajo, tratando de evitar cualquier tipo de interferencia y documentando cada una de las acciones importantes realizadas por los empleados que a su vez son relevantes para la elaboración de este proyecto.

La siguiente figura muestra el flujo que sigue el encargado de realizar el cálculo de planilla.

Figura 6. Proceso cálculo planilla

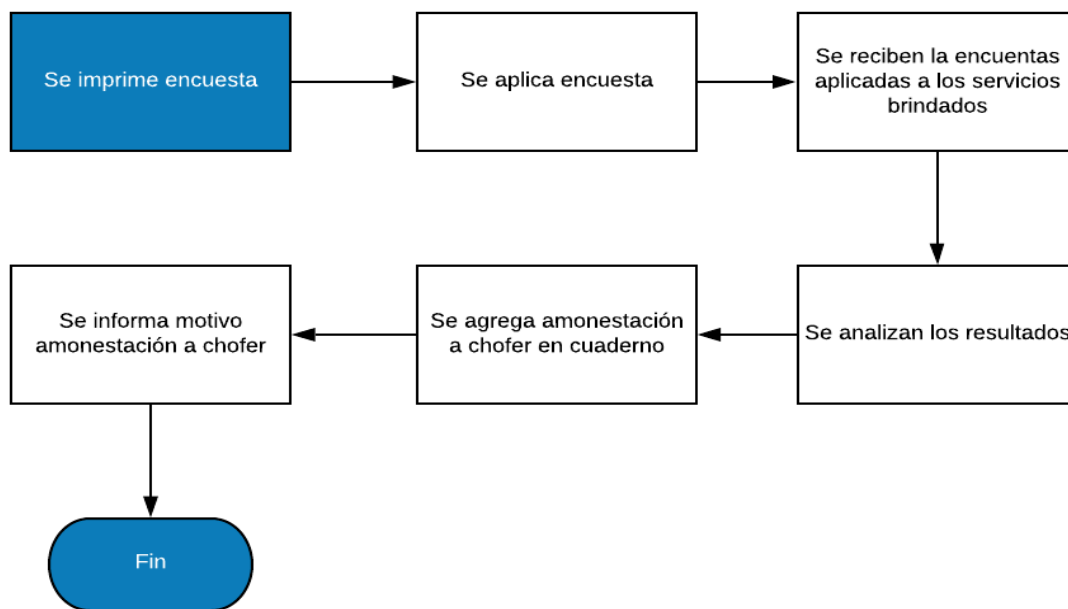


El encargado inicia comprobando si el empleado cuenta con algún aumento salarial, incentivo o retención salarial aplicado recientemente. Una vez confirmado esos datos, procede a utilizar una plantilla creada por el mismo para facilitar el cálculo, en esta comienza a ingresar datos como el nombre del empleado, total de horas laboradas y horas extras. En el caso de los empleados con el rol de chofer, se observa que realizan un procedimiento muy similar, en este, algunos datos cambian o no se hacen uso de ellos y el resultado obtenido es sumado al proceso principal realizado previamente para así obtener un único valor. Este

segundo cálculo lo trabajan junto al primero, pero sin llegar al colisionar el uno con el otro, aunque este sea relativamente más sencillo.

Una vez realizados los cálculos necesarios, se envía el archivo Excel a los secretarios administrativos quienes se encargan de realizar los pagos, en primera instancia verifican que los montos a pagar sean exactos y finalmente proceden a depositar el dinero a las cuentas bancarias de los empleados.

Figura 7. Proceso control disciplinario

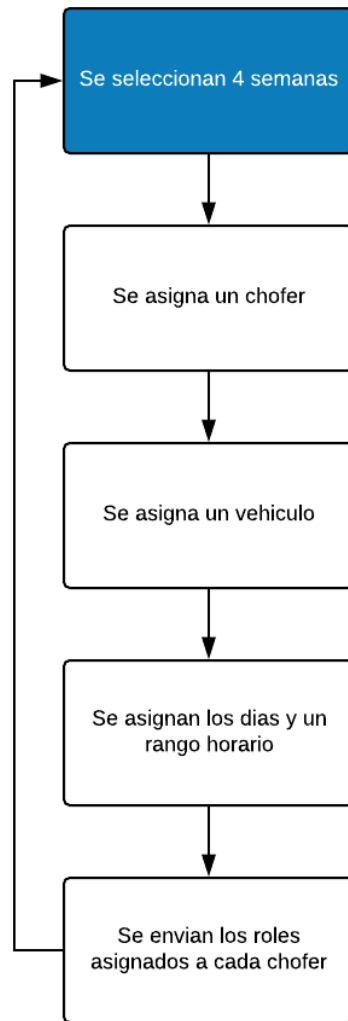


Se logra observar que este no es un proceso que realicen con mucha frecuencia para el servicio de transporte público, pero si en el caso del servicio turístico. Normalmente organizan un viaje de este tipo al mes y entre las acciones que realizan se encuentra el generar y aplicar una pequeña encuesta a los clientes o usuarios con el fin de recibir la mayor cantidad de retroalimentación posible. Esta encuesta en realidad son un par de preguntas.

Por otra parte, se logra apreciar que al recibir las encuestas proceden a analizarlas; en este análisis no se destaca nada especial, únicamente categorizan y contabilizan las respuestas obtenidas con el fin de determinar si se brindó un buen servicio o no, y que aspectos deben mejorarse. Si se concluye que el servicio fue de mala calidad se procede a amonestar al chofer

encargado del respectivo viaje, apuntando esta amonestación en un cuaderno y posteriormente se le informa el motivo de esta.

Figura 8. Proceso asignación tareas

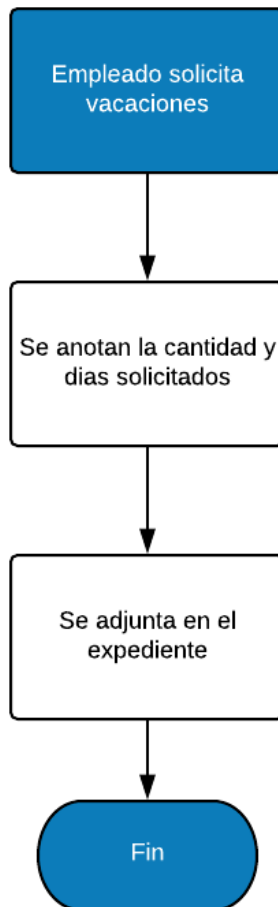


A primera vista se observa que utilizan una plantilla en Excel elaborada por ellos mismos, en esta plantilla se logra apreciar divisiones, cada división representa una semana, para un total de cuatro divisiones, se da por hecho que esta labor debe realizarse cada mes.

Adicionalmente, se observan los nombres de los empleados, en este caso los choferes. A cada chofer se le asigna un autobús, el tipo de servicio que debe brindar y el horario que este debe cumplir, finalmente se imprime esta plantilla y se le otorga a cada uno de los empleados.

Comparando los documentos de hace un mes atrás y el actual, se logra apreciar que el único rubro que presenta una variación es el nombre del empleado, los demás rubros se mantienen exactamente igual.

Figura 9. Proceso vacaciones



Este proceso es el más sencillo de todos. Primeramente, cuando el empleado solicita vacaciones se accede al expediente en busca de un documento, en este documento se van contabilizando la cantidad de días vacacionales que el empleado va acumulando, la cantidad de días solicitados y que días los utilizará. Finalmente, este documento se almacena nuevamente en el expediente y se genera un comprobante para posteriormente ser aprobado y firmado por un encargado.

## **Análisis de resultados**

Gracias a la aplicación de la entrevista y la observación se logra identificar diferentes puntos. A continuación, se enumeran estos:

Con la información recopilada de estos dos métodos aplicados, se puede confirmar que la empresa cuenta con conocimientos básicos y necesarios para el uso de una computadora y dispositivo móvil capaz de hacer una conexión a internet. Gracias a esto, se verifica que la factibilidad operativa del prototipo funcional se cumple sin ningún problema.

Por otra parte, se confirma la aprobación por parte de los empleados y de la organización en general, están conscientes que hacer el cambio a la digitalización y más específicamente a la relacionada con los recursos humanos, es un gran cambio, pero demuestran una actitud positiva hacia el mismo.

La entrevista aplicada confirma que el proceso para el cálculo de planilla es tedioso y genera atrasos en los pagos de los empleados, dichos empleados aseguran que es normal no recibir los pagos en las fechas correspondiente. Es importante tener en consideración esta problemática para que el disgusto de los empleados no aumente y genere problemas a futuro.

Relacionado al método de observación, más precisamente en el proceso para el cálculo de la planilla, si identifican varios aspectos. Primeramente, que a los empleados con rol de chofer se les calcula dos salarios distintos, este debido a que ellos brindan servicios especiales los cuales pueden ser viajes turísticos y transporte de estudiantes u otro tipo de personas. Para el cálculo de este servicio, el salario que ganan por hora en la mayoría de casos es un poco menor, también las horas extras no son contabilizadas y se aplican las deducciones por ley necesarias. Otro aspecto es que el cálculo realizado para el servicio especial es sumado al cálculo normal, es decir que se obtiene un único resultado el cual es el monto que se le pagará al empleado.

En el control disciplinario, lo que se busca conocer es si el chofer asignado para determinado viaje de servicio especial brindó un buen servicio a los clientes, en simples palabras, que tan bien fue el trato del chofer con los clientes. Si se reciben comentarios negativos se procurará asignarle este tipo de servicio con menos frecuencia. Todo este proceso se interpreta como un sistema de puntaje o calificación en donde los clientes le asignan una calificación y/o un comentario al chofer, posteriormente se calcula un promedio

el cual será otorgado al chofer. El rango utilizado más común es de cero a cinco. La empresa procura que el servicio brindado sea el mejor posible por lo que con una calificación de 3,5 hacia arriba es el apto para que los choferes tengan más oportunidades de que se les asigne un viaje de servicio especial.

Los módulos propuestos para la elaboración del prototipo funcional responden positivamente a las necesidades que establecidas por la empresa. Cada uno de estos módulos se encarga de una necesidad específica,}; sin embargo, cada uno de ellos trabaja de manera conjunta con los demás para así realizar los procesos de manera más controlada y con un correcto funcionamiento como es característico de un sistema de información.

Por último, el software desarrollado permitirá una mayor rapidez y efectividad en los procesos que realiza la empresa en comparación a como estos se efectúan actualmente, reflejando así cambios positivos es la gestión de sus recursos humanos.

## CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES

### Conclusiones

Al inicio de este proyecto, se planteó un objetivo general y una serie de objetivos específicos con el fin de generar una guía que ayude a la elaboración del presente documento. Con la finalización del prototipo funcional y tomando en cuenta un resultado satisfactorio en el desarrollo de este, se presentan las conclusiones para cada uno de los objetivos específicos.

Con relación al análisis de requerimientos, se puede concluir que para el desarrollo de este prototipo funcional se recolectaron y analizaron los debidos requerimientos satisfactoriamente gracias a las reuniones con los diferentes miembros de la junta directiva en donde se llevó acabo la elaboración de este prototipo. Asimismo, se hizo uso de técnicas como entrevistas y la observación, siendo esta última indispensable para conocer el entorno de trabajo lo cual permitió comprender y confirmar los problemas descritos por el personal de la empresa.

Luego de recopilar toda la información necesaria en la etapa de análisis, esta fue procesada con el fin de extraer lo más relevante lo cual fue tomado en cuenta para la elaboración de este prototipo, en base a lo obtenido; se logró realizar los diferentes diseños que deben ser incluidos para todo proyecto de desarrollo de software, entre ellos los diferentes diagramas UML que facilitaron la etapa de programación para el prototipo, el diseño de las arquitecturas, tanto del sistema como del software, diseño de la base de datos la cual almacena toda información de la empresa, y por último el diseño de las interfaces de usuario las cuales presentan un aspecto simple, limpio y ordenado; además de ser amigable lo que hace entender al usuario que es fácil de utilizar.

Por otra parte, se concluye que el desarrollo de la programación del prototipo funcional cumple con cada uno de los diseños propuestos en la etapa anterior, satisfaciendo así los requerimientos planteados. Además, se logró desarrollar un sistema el cual cuenta con los diferentes módulos definidos al inicio de este documento y una base de datos con información persistente de la cual hace uso la aplicación elaborada, dando así solución a las problemáticas plantadas en un inicio.

Al final de todo desarrollo que utilice un lenguaje de programación, es necesario la realización de pruebas para corroborar un óptimo funcionamiento en el software elaborado, estas pruebas tienen como objetivo minimizar lo más posible aquellos problemas que puedan llegar a ocurrir durante el uso del sistema. La serie de pruebas aplicadas a las diferentes funcionalidades del prototipo permitió encontrar fallos que posteriormente fueron corregidos. Una vez corregidos estos fallos, se garantiza que cada uno de los módulos cuenta con un correcto funcionamiento, permitiendo así una implementación sin ningún inconveniente de por medio.

Finalmente, dicho todo lo anterior, se da por concluido el objetivo general, el cual consiste en desarrollar un prototipo funcional para la gestión de recursos humanos en la Asociación de Desarrollo Integral de San Juan Puriscal y dando por completados todos los objetivos planteados.

### **Recomendaciones**

A continuación, se presentan algunas recomendaciones como parte de posibles mejoras que se pueden incorporar como nuevas funcionalidades para futuras versiones.

Se recomienda la implementación de web sockets en un plazo menor a un mes y deberá realizarse por el desarrollador de este prototipo u otro con los conocimientos necesarios tanto en el lenguaje como la tecnología a implementar. Esto con el fin de eliminar la brecha de tiempo entre en las diferentes solicitudes que realizan los empleados como las horas extras y permisos, y el tiempo en que toma a un administrador ver la solicitud y proceder a rechazarla o aceptarla. Esto permitirá que el sistema notifique a los administradores en tiempo real cuando un empleado realice una nueva solicitud, realizando una acción sobre esta en un minuto o menos. Adicionalmente, esta funcionalidad puede ser extendida a otros clientes como lo puede ser una aplicación móvil o de escritorio en caso de crearlos en un futuro.

También se recomienda implementar gráficos, esta labor es necesario de 10 días de desarrollo y de un programador con conocimientos en JavaScript, Node.js y MySQL. Estos gráficos permitirán visualizar información como los empleados con mayor cantidad de permisos solicitados, empleados a los que se les han asignado más amonestaciones y

felicitaciones, entre otros datos relevantes. Integrando estos gráficos con la recomendación anterior permite conocer estos datos de manera más precisa, rápida y accesible.

Por otro lado, se recomienda la creación de manuales, tanto el manual de usuario el cual permite una mejor comprensión sobre el funcionamiento del prototipo, así como el manual técnico para facilitar las labores de mantenimiento a nivel de código fuente. Para la elaboración de estos manuales se requerirá de un plazo de 4 días además de ser realizado por el desarrollador del prototipo u otra persona con los conocimientos necesarios sobre el lenguaje utilizado y el entorno de ejecución para este.

Se recomienda llevar la base de datos a los servicios de Amazon Web Services en un plazo no mayor a 3 días, esta labor deberá ser realizada por una persona con conocimientos en administración de servidores. La razón de esto se debe a que el servicio Heroku presenta costos elevados en relación a las características que ofrece.

Se recomienda la adquisición de un certificado SSL, así como la implementación del mismo en un periodo de un día y por una persona con conocimientos en administración de servidores o servicios en la nube, esto con el fin de aumentar la seguridad cifrando todo el tráfico entre los diferentes usuarios que establecen una conexión con la aplicación.

También, se recomienda la implementación de diferentes parámetros que permitan modificar los topes en las cargas sociales e impuesto de renta en un periodo no mayor a cuatro días y bajo la responsabilidad del autor del prototipo funcional. Esto con el propósito de no generar dependencia con un desarrollador a la hora de modificar dichos valores.

Por último, es recomendable la implementación completa de este sistema en un plazo aproximado de 2 días y por una persona con conocimientos en la utilización de la plataforma Heroku. Asimismo, se ofrece 24 horas como valor agregado por parte del creador de este prototipo para asegurar un correcto funcionamiento en dicha implementación. Esto con el objetivo solventar la necesidad proveniente de la Asociación de Desarrollo Integral de San Juan Puriscal.

## **CAPÍTULO VI: PROPUESTA**

En este capítulo se presentarán los detalles correspondientes al prototipo desarrollado para la empresa creando así la solución para los problemas expuestos en capítulos anteriores. Casos de uso, diagramas, diferentes diseños que abarcan las arquitecturas de software y sistema, el modelo de base de datos, la programación y pruebas realizadas, entre otros aspectos importantes. Todo lo anterior permitirá una mejor comprensión del comportamiento del prototipo al explicar más detalladamente su funcionamiento.

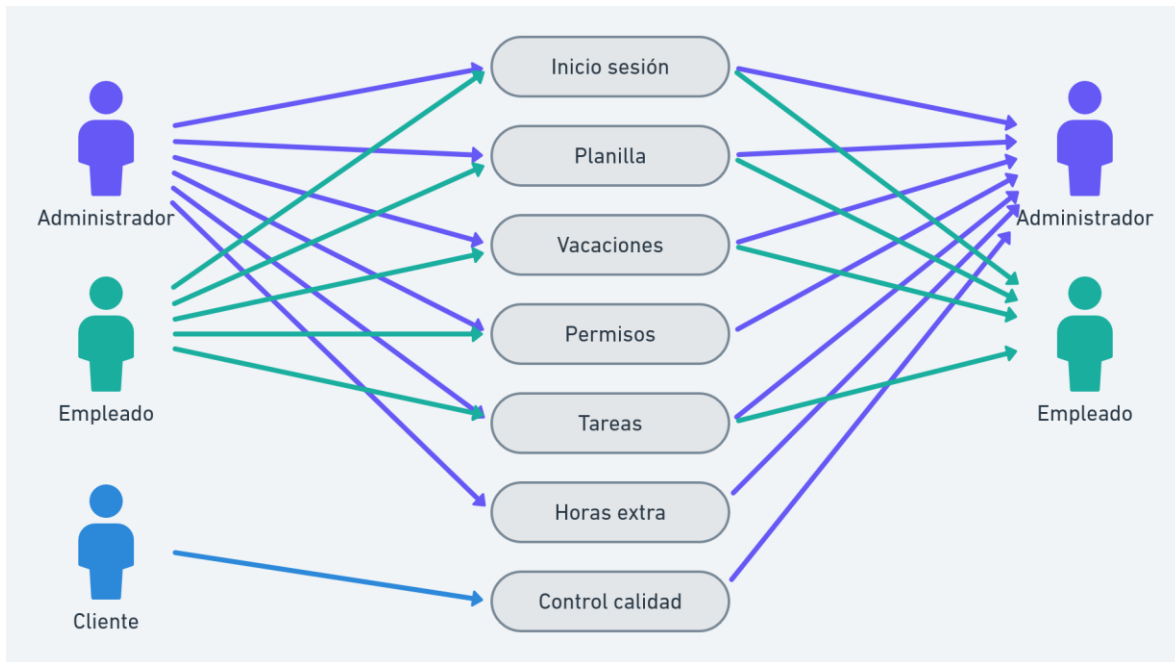
### **Análisis**

Como se mencionó anteriormente en la metodología de desarrollo, en este caso la metodología de ciclo de vida, el análisis es su primera etapa y consiste en conocer las necesidades y analizar, valga la redundancia, los problemas presentes en la empresa. Se realizó una reunión con algunos miembros de la junta directiva de la Asociación de Desarrollo Integral de San Juan Puriscal para conocer y detallar los requerimientos con el fin de satisfacer sus necesidades.

### **Casos de uso**

A continuación, se muestran los casos de uso, estos hacen referencia a las actividades y procesos presentes en el prototipo. Como se muestra en la figura # se muestran los diferentes requerimientos establecidos gracias a la etapa de análisis. Además, de los actores quienes interactúan con estos por medio del prototipo, esto genera una perspectiva más clara del actor y sus diferentes acciones dentro del sistema.

Figura 10. Diagrama de casos de uso



Fuente: Elaboración propia.

Los actores en cuestión, representan a los diferentes miembros de la junta directiva, en este caso el administrador, los secretarios y choferes que serán el empleado. Por último el cliente; aquellos que realizan los viajes turísticos con la empresa e interactúan con el sistema brindando retroalimentación sobre el servicio que se les brindó.

Tabla 5. Caso de uso 01 – Inicio de sesión

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 01	<b>Nombre del caso de uso:</b> Inicio de sesión
<b>Fecha de elaboración:</b>	20/05/2020
<b>Descripción caso de uso:</b>	Permite a los empleados y administradores acceder al sistema por medio de sus credenciales.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador y trabajador.
<b>Precondiciones:</b>	Tener conexión a internet.
<b>Flujo básico del caso de uso</b>	
Inicia cuando un actor desea acceder al sistema, este requiere iniciar sesión utilizando sus credenciales las cuales corresponden a un correo electrónico y una contraseña en los respectivos campos habilitados, consecutivamente, el actor presiona o da clic en el botón de iniciar sesión, el sistema comprueba los datos ingresados y permite ingresar al sistema.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. Si el usuario a ingresado al sistema previamente, el sub flujo No. 1: Sesión iniciada es ejecutado.</li> <li>3. El sistema muestra la página de inicio de sesión.</li> <li>4. El usuario ingresa sus credenciales las cuales son el correo electrónico y contraseña</li> <li>5. El sistema valida las credenciales ingresadas y permite el ingreso al sistema.</li> </ol>	
<b>Sub Flujos</b>	
<b>Sub Flujo No. 1:</b> Sesión iniciada	El sistema detecta si el usuario ha iniciado sesión previamente o intenta acceder a la página de inicio de sesión, automáticamente se redirecciona a la página principal de sistema.
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si las credenciales son incorrectas, el sistema no permitirá el ingreso al sistema.

<b>Flujo Alternativo No. 2</b>	Si el correo electrónico no cumple el formato, el sistema deberá advertir al usuario.
<b>Flujo Alternativo No. 3</b>	Si la contraseña no cumple el mínimo de caracteres, el sistema deberá advertir al usuario.
<b>Flujo Alternativo No. 4</b>	Si el usuario no digita nada en los campos requerido, el sistema deberá advertirle.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
Una vez iniciada la sesión, el usuario puede hacer uso de las funcionalidades del sistema.	

Fuente: Elaboración propia.

Tabla 6. Caso de uso 02 - Nuevo empleado

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 02	<b>Nombre del caso de uso:</b> Nuevo empleado
<b>Fecha de elaboración:</b>	20/05/2020
<b>Descripción caso de uso:</b>	Permite registrar un nuevo empleado en el sistema.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador.
<b>Precondiciones:</b>	Tener conexión a internet, haber ingresado en el sistema y poseer el rol de administrador.
<b>Flujo básico del caso de uso</b>	
El administrador ingresa a la sección o pantalla de Nuevo empleado, seguidamente completa el formulario que aparece en pantalla y da clic en el botón Agregar nuevo empleado, finalmente el empleado es agregado al sistema.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario selecciona la opción "Nuevo empleado" en el menú.</li> <li>3. El sistema carga un formulario.</li> <li>4. El usuario completa el formulario con la información del empleado.</li> <li>5. El usuario presiona el botón "Agregar empleado".</li> </ol>	

6. El sistema registra el nuevo empleado.	
<b>Sub Flujos</b>	
<b>Sub Flujo No. 1:</b> Empleado temporal	Si el nuevo empleado a agregar es de tipo temporal, el sistema habilita dos campos nuevos en el formulario.
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrara una advertencia.
<b>Flujo Alterno No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema registra el nuevo empleado.	

Fuente: Elaboración propia.

Tabla 7. Caso de uso 03 - Actualizar empleado

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 03	<b>Nombre del caso de uso:</b> Actualizar empleado
<b>Fecha de elaboración:</b>	20/05/2020
<b>Descripción caso de uso:</b>	Permite actualizar la información de un empleado.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador.
<b>Precondiciones:</b>	Tener conexión a internet, haber ingresado en el sistema y poseer el rol de administrador.
<b>Flujo básico del caso de uso</b>	
El administrador ingresa a la sección o pantalla Ver empleados, seguidamente se mostrará una pantalla con todos los empleados activos en el sistema, el administrador presiona en Ver más información del empleado, luego se mostrará una ventana con los campos necesarios para	

actualizar la información, finalmente el administrador completa los campos y presión el botón Actualizar.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario selección la opción “Actualizar empleado” en el menú.</li> <li>3. El sistema carga un formulario.</li> <li>4. El usuario completa el formulario con la nueva información del empleado.</li> <li>5. El usuario presiona el botón “Actualizar información”.</li> <li>6. El sistema registra el nuevo empleado.</li> </ol>	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrara una advertencia.
<b>Flujo Alterno No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema actualiza la información del empleado.	

Fuente: Elaboración propia.

*Tabla 8. Caso de uso 04 -Recontratar empleado temporal*

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 04	<b>Nombre del caso de uso:</b> Recontratar empleado temporal
<b>Fecha de elaboración:</b>	20/05/2020
<b>Descripción caso de uso:</b>	Permite recontratar un empleado temporal.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador.

<b>Precondiciones:</b>	Tener conexión a internet, haber ingresado en el sistema y poseer el rol de administrador.
<b>Flujo básico del caso de uso</b>	
El administrador ingresa a la sección de recontractar empleado temporal y el sistema le muestra un formulario en el cual ingresará la fecha de salida del empleado y una descripción de por qué ha sido recontractado.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario selección la opción “Recontractar empleado temporal” en el menú.</li> <li>3. El sistema carga un formulario.</li> <li>4. El usuario completa el formulario con la información necesaria.</li> <li>5. El usuario presiona el botón “Recontractar empleado”.</li> <li>6. El sistema recontracta al empleado temporal.</li> </ol>	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrará una advertencia.
<b>Flujo Alterno No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema actualiza la información del empleado.	

Fuente: Elaboración propia.

Tabla 9. Caso de uso 05 - Despedir empleado

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 05	<b>Nombre del caso de uso:</b> Despedir empleado
<b>Fecha de elaboración:</b>	20/05/2020
<b>Descripción caso de uso:</b>	Permite al administrador despedir un empleado.

<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador.
<b>Precondiciones:</b>	Tener conexión a internet, haber ingresado en el sistema y poseer el rol de administrador.
<b>Flujo básico del caso de uso</b>	
Estando el administrador en la sección o pantalla que muestra la información del empleado, presiona el botón que dice Despedir, seguidamente se mostrará una ventana solicitando la razón del despido, por último el administrador presiona el botón Confirmar despido.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario selección la opción “Despedir empleado” en el menú.</li> <li>3. El sistema solicita el ingresar la razón del despido.</li> <li>4. El usuario ingresa el motivo de despido.</li> <li>5. El usuario presiona el botón “Despedir empleado”.</li> <li>6. El sistema despide al empleado.</li> </ol>	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si el campo habilitado no cumple con el formato correcto, el sistema mostrará una advertencia.
<b>Flujo Alterno No. 2</b>	Si el campo habilitado se encuentra vacío, el sistema mostrará una advertencia.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema despide el empleado y lo traslada a la sección de ex empleados.	

Fuente: Elaboración propia.

Tabla 10. Caso de uso 06 - Nuevo permiso

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 06	<b>Nombre del caso de uso:</b> Nuevo permiso
<b>Fecha de elaboración:</b>	25/05/2020

<b>Descripción caso de uso:</b>	Permite a los empleados solicitar un permiso.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Empleado.
<b>Precondiciones:</b>	Tener conexión a internet, haber ingresado en el sistema.
<b>Flujo básico del caso de uso</b>	
El empleado ingresa a la sección o pantalla Solicitar nuevo permiso, se cargara un formulario que será necesario que el empleado complete, posteriormente deberá presionar el botón Solicitar.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario selección la opción “Nuevo permiso” en el menú.</li> <li>3. El sistema carga un formulario en la pantalla.</li> <li>4. El usuario completa el formulario.</li> <li>5. El usuario presiona el botón “Solicitar permiso”.</li> <li>6. El sistema registra la solicitud del nuevo permiso.</li> </ol>	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrará una advertencia.
<b>Flujo Alterno No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema registra la solicitud del empleado en el sistema como pendiente de aprobación.	

Fuente: Elaboración propia.

Tabla 11. Caso de uso 07 - Ver permisos

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 07	<b>Nombre del caso de uso:</b> Ver los permisos.

<b>Fecha de elaboración:</b>	25/05/2020
<b>Descripción caso de uso:</b>	Permite al administrador y empleado ver todos los permisos solicitados.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador y empleado.
<b>Precondiciones:</b>	Tener conexión a internet, haber ingresado en el sistema.
<b>Flujo básico del caso de uso</b>	
El empleado o administrador ingresan a la sección o pantalla Ver permisos, en el caso del administrador se cargan todos los permisos de los empleados y en el caso del empleado, se cargan únicamente sus permisos.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario selección la opción “Ver permisos” en el menú.</li> <li>3. El sistema carga todos los permisos, el sub flujo No. 1: Permisos por rol es ejecutado.</li> </ol>	
<b>Sub Flujos</b>	
<b>Sub Flujo No. 1:</b> Permisos por rol	Si el usuario ingresado en el sistema posee un rol de administrador, el sistema cargar los permisos de todos los empleados. En caso contrario, el sistema carga únicamente los permisos del empleado.
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si los empleados o el empleado no han solicitado permisos, el sistema mostrará una imagen o texto indicando que no existen permisos.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema carga todos los permisos con éxito dependiente el rol del usuario.	

Fuente: Elaboración propia.

Tabla 12. Caso de uso 08 - Aprobar o rechazar permiso

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 08	<b>Nombre del caso de uso:</b> Aprobar o rechazar permiso
<b>Fecha de elaboración:</b>	25/05/2020
<b>Descripción caso de uso:</b>	Permite al administrador aceptar o rechazar la solicitud de permiso realizada por un empleado.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador.
<b>Precondiciones:</b>	Tener conexión a internet, haber ingresado en el sistema y poseer el rol de administrador.
<b>Flujo básico del caso de uso</b>	
El administrador ingresa a la pantalla Ver permisos, una vez cargados los permisos, este elije cual permiso aceptar o rechazar y automáticamente pasan a tener un estado aprobado o rechazado.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario selecciona la opción “Ver permisos” en el menú.</li> <li>3. El sistema carga los permisos.</li> <li>4. El usuario abre un permiso.</li> <li>5. Si el usuario presiona el botón “Aceptar permiso”, el sub flujo No. 1: Aceptar permiso es ejecutado.</li> <li>6. Si el usuario presiona el botón “Rechazar permiso”, el sub flujo No. 2: Rechazar permiso es ejecutado.</li> <li>7. El sistema actualiza el permiso solicitado por el empleado.</li> </ol>	
<b>Sub Flujos</b>	
<b>Sub Flujo No. 1:</b> Aceptar permiso	El sistema actualiza automáticamente el estado del permiso a aprobado.
<b>Sub Flujo No. 1:</b> Rechazar permiso	El sistema actualiza automáticamente el estado del permiso a rechazado.
<b>Flujos Alternos</b>	

<b>Flujo Alternativo No. 1</b>	Si el sistema no puede actualizar el permiso, se mostrará una alerta indicando que no se pudo actualizar.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condicionales</b>	
El sistema actualiza el estado del permiso con éxito.	

Fuente: Elaboración propia.

Tabla 13. Caso de uso 09 - Editar permiso

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 09	<b>Nombre del caso de uso:</b> Editar permiso
<b>Fecha de elaboración:</b>	25/05/2020
<b>Descripción caso de uso:</b>	Permite al empleado actualizar la solicitud realizada anteriormente.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Empleado.
<b>Precondiciones:</b>	Tener conexión a internet, haber ingresado en el sistema.
<b>Flujo básico del caso de uso</b>	
El empleado ingresa a la pantalla Ver permisos, selecciona un permiso y presiona el botón Editar, posteriormente se muestra una pantalla conteniendo un formulario, el empleado completa el formulario y presiona el botón Aceptar. El permiso editado pasa automáticamente a tener un estado pendiente de aprobación.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario selecciona la opción “Ver permisos” en el menú.</li> <li>3. El sistema carga los permisos del empleado.</li> <li>4. El usuario abre un permiso.</li> <li>5. El sistema carga un formulario con los campos necesarios para actualizar el permiso.</li> <li>6. El usuario completa el formulario.</li> <li>7. El usuario presiona el botón “Confirmar”.</li> </ol>	

8. El sistema actualiza el permiso y cambia el estado pendiente.	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrara una advertencia indicando que los datos ingresados no cumplen el formato necesario.
<b>Flujo Alterno No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia y solicita al usuario que complete los campos requeridos.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema actualiza el estado del permiso con éxito.	

Fuente: Elaboración propia.

Tabla 14. Caso de uso 10 - Borrar permiso

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 10	<b>Nombre del caso de uso:</b> Borrar permiso
<b>Fecha de elaboración:</b>	25/05/2020
<b>Descripción caso de uso:</b>	Permite al administrador y empleado borrar un permiso.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador y empleado.
<b>Precondiciones:</b>	Tener conexión a internet, haber ingresado en el sistema.
<b>Flujo básico del caso de uso</b>	

El actor involucrado ingresa a la pantalla Ver permisos, elige el o los permisos que desea eliminar y presiona el botón Eliminar, si el actor es un empleado, el permiso es eliminado únicamente para él, de caso contrario, el o los permisos son eliminados para ambos actores.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario selecciona la opción “Ver permisos” en el menú.</li> <li>3. El sistema carga los permisos dependiendo del rol.</li> <li>4. El usuario abre un permiso.</li> <li>5. El usuario presiona el botón “Borrar permiso”, el sub flujo No. 1: Borrar permiso es ejecutado</li> </ol>	
<b>Sub Flujos</b>	
<b>Sub Flujo No. 1:</b> Borrar permiso	El sistema detecta que usuario está borrando el permiso, si es un empleado, el permiso será borrado únicamente para él, en caso contrario, el permiso será borrado tanto para el empleado como para el administrador.
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrara una advertencia indicando que los datos ingresados no cumplen el formato necesario.
<b>Flujo Alterno No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia y solicita al usuario que complete los campos requeridos.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema borra el permiso con éxito dependiendo del rol que posea el usuario.	

Fuente: Elaboración propia.

Tabla 15. Caso de uso 11 - Perfil de usuario

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 11	<b>Nombre del caso de uso:</b> Perfil de usuario
<b>Fecha de elaboración:</b>	27/05/2020
<b>Descripción caso de uso:</b>	Permite a los empleados ver su información.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Empleado.
<b>Precondiciones:</b>	Tener conexión a internet, haber ingresado en el sistema.
<b>Flujo básico del caso de uso</b>	
El empleado ingresa a la pantalla Perfil éste carga toda la información necesaria que está relacionada al empleado.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario selecciona la opción “Ver perfil” en el menú.</li> <li>3. El sistema carga toda la información relacionada al empleado.</li> </ol>	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si el sistema no puede cargar la información del empleado, se mostrará una imagen o texto indicando que no se pudo obtener la información.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema obtiene y muestra la información del usuario con éxito.	

Fuente: Elaboración propia.

Tabla 16. Caso de uso 12 - Editar información de contacto

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 12	<b>Nombre del caso de uso:</b> Editar información de contacto.
<b>Fecha de elaboración:</b>	27/05/2020
<b>Descripción caso de uso:</b>	Permite al empleado actualizar su información de contacto.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Empleado.
<b>Precondiciones:</b>	Tener conexión a Internet, haber ingresado en el sistema.
<b>Flujo básico del caso de uso</b>	
El empleado ingresa a la pantalla Perfil, se carga toda la información relacionada a él, el empleado presiona el botón Actualizar información de contacto, se carga una pantalla con un formulario, el empleado completa el formulario y presiona el botón Actualizar.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario selecciona la opción “Ver perfil” en el menú.</li> <li>3. El sistema carga la información relacionada al empleado.</li> <li>4. El usuario presiona el botón “Editar información”.</li> <li>5. El sistema abre un formulario con los campos necesarios para actualizar la información de contacto.</li> <li>6. El usuario completa el formulario.</li> <li>7. El usuario presiona el botón “Guardar cambios”.</li> <li>8. El sistema actualiza la información de contacto.</li> </ol>	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrara una advertencia indicando que los datos ingresados no cumplen el formato necesario.

<b>Flujo Alterno No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia y solicita al usuario que complete los campos requeridos.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema actualiza la información de contacto relacionada con el empleado de manera exitosa.	

Fuente: Elaboración propia.

Tabla 17. Caso de uso 13 - Nueva solicitud horas extras

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 13	<b>Nombre del caso de uso:</b> Nueva solicitud horas extras
<b>Fecha de elaboración:</b>	29/05/2020
<b>Descripción caso de uso:</b>	Permite al empleado crear una nueva solicitud de horas extras.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Empleado.
<b>Precondiciones:</b>	Tener conexión a Internet, haber ingresado en el sistema.
<b>Flujo básico del caso de uso</b>	
El empleado ingresa a la sección nueva, horas extras, el sistema carga un formulario el cual completa el empleado, finalmente el empleado presión el botón Solicitar horas extra.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario selecciona la opción “Ver solicitud” en el menú de horas extras.</li> <li>3. El sistema carga un formulario con los campos necesarios para realizar la solicitud.</li> <li>4. El usuario completa el formulario.</li> <li>5. El usuario presiona el botón “Realizar solicitud”.</li> <li>6. El sistema registra la solicitud creada por el empleado.</li> </ol>	
<b>Sub Flujos</b>	

<b>Flujos Alternos</b>	
<b>Flujo Alternativo No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrara una advertencia indicando que los datos ingresados no cumplen el formato necesario.
<b>Flujo Alternativo No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia y solicita al usuario que complete los campos requeridos.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema actualiza el estado del permiso con éxito.	

Fuente: Elaboración propia.

Tabla 18. Caso de uso 14 - Ver horas extras

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 14	<b>Nombre del caso de uso:</b> Ver horas extras.
<b>Fecha de elaboración:</b>	29/05/2020
<b>Descripción caso de uso:</b>	Permite al administrador y empleado ver todas las horas extras solicitadas.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador y empleado.
<b>Precondiciones:</b>	Tener conexión a internet, haber ingresado en el sistema.
<b>Flujo básico del caso de uso</b>	
El actor ingresa a la sección Ver horas extras, en caso de que el actor sea un administrador se mostraran todas las solicitudes de horas extras, de caso contrario solo se mostraran las solicitudes correspondientes para cada empleado.	
<b>Detalle paso a paso del flujo básico</b>	

<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario selección la opción “Ver permisos” en el menú.</li> <li>3. El sistema carga todos los permisos, el sub flujo No. 1: Horas extras por rol es ejecutado.</li> </ol>	
<b>Sub Flujos</b>	
<b>Sub Flujo No. 1:</b> Horas extras por rol	Si el usuario ingresado en el sistema posee un rol de administrador, el sistema carga todas las horas extras solicitadas por los empleados. En caso contrario, el sistema carga únicamente las horas extras que solicito el empleado.
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si los empleados o el empleado no han solicitado permisos, el sistema mostrará una imagen o texto indicando que no existen permisos.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema carga todas las solicitudes de horas extras con éxito dependiente el rol del usuario.	

Fuente: Elaboración propia.

Tabla 19. Caso de uso 15 - Aprobar o rechazar horas extras

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 15	<b>Nombre del caso de uso:</b> Aprobar o rechazar horas extras
<b>Fecha de elaboración:</b>	29/05/2020
<b>Descripción caso de uso:</b>	Permite al administrador aceptar o rechazar las solicitudes de horas extras realizadas por los empleados.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador.
<b>Precondiciones:</b>	Tener conexión a internet, haber ingresado en el sistema y poseer el rol de administrador.

<b>Flujo básico del caso de uso</b>	
El administrador ingresa a la sección Ver horas extras, en esta se cargan todas las solicitudes de horas extras, posteriormente el elije aquellas que desea aceptar o rechazar, automáticamente estas pasan a tener un estado aprobado o rechazado.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario selecciona la opción “Ver solicitudes” en el menú de horas extras.</li> <li>3. El sistema carga las solicitudes de horas extras.</li> <li>4. El usuario abre una solicitud.</li> <li>5. Si el usuario presiona el botón “Aceptar solicitud”, el sub flujo No. 1: Aceptar solicitud es ejecutado.</li> <li>6. Si el usuario presiona el botón “Rechazar solicitud”, el sub flujo No. 2: Rechazar permiso es ejecutado.</li> <li>7. El sistema actualiza las horas extras solicitadas por el empleado.</li> </ol>	
<b>Sub Flujos</b>	
<b>Sub Flujo No. 1:</b> Aceptar permiso	El sistema actualiza automáticamente el estado de la solicitud a aprobado.
<b>Sub Flujo No. 1:</b> Rechazar permiso	El sistema actualiza automáticamente el estado de la solicitud a rechazado.
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si el sistema no puede actualizar la solicitud de horas extras, se mostrará una alerta indicando que no se pudo actualizar.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema actualiza el estado de la solicitud de horas extras con éxito.	

Fuente: Elaboración propia.

Tabla 20. Caso de uso 16 - Editar solicitud horas extras

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>
---

<b>Número de caso de uso:</b> 16	<b>Nombre del caso de uso:</b> Editar solicitud horas extras
<b>Fecha de elaboración:</b>	29/05/2020
<b>Descripción caso de uso:</b>	Permite al empleado editar una solicitud de horas extras realizada anteriormente.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Empleado.
<b>Precondiciones:</b>	Tener conexión a internet, haber ingresado en el sistema.
<b>Flujo básico del caso de uso</b>	
El empleado ingresa a la sección Ver horas extras, todas las solicitudes realizadas por este empleado son cargadas, selecciona la solicitud que desea editar y presiona el botón Editar, se muestra una pantalla con un formulario el cual debe completar, una vez completado el formulario presiona el botón Aceptar. La solicitud de horas extras editada pasa automáticamente a tener un estado pendiente de aprobación.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario selecciona la opción “Ver solicitudes” en el menú de horas extras.</li> <li>3. El sistema carga las solicitudes de horas extras del empleado.</li> <li>4. El usuario abre una solicitud.</li> <li>5. El sistema carga un formulario con los campos necesarios para actualizar la solicitud.</li> <li>6. El usuario completa el formulario.</li> <li>7. El usuario presiona el botón “Confirmar”.</li> <li>8. El sistema actualiza la solicitud y cambia el estado pendiente.</li> </ol>	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrara una advertencia indicando que los datos ingresados no cumplen el formato necesario.

<b>Flujo Alternativo No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia y solicita al usuario que complete los campos requeridos.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema actualiza el estado de la solicitud de horas extras con éxito.	

Fuente: Elaboración propia.

Tabla 21. Caso de uso 17 - Borrar solicitud horas extras

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 17	<b>Nombre del caso de uso:</b> Borrar solicitud horas extras
<b>Fecha de elaboración:</b>	29/05/2020
<b>Descripción caso de uso:</b>	Permite al administrador y empleado borrar un permiso.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador y empleado.
<b>Precondiciones:</b>	Tener conexión a internet, haber ingresado en el sistema.
<b>Flujo básico del caso de uso</b>	
El actor ingresa a la sección Ver horas extras, selecciona aquellas que desea eliminar y presiona el botón Eliminar, en caso de que el actor sea un empleado, la solicitud será eliminada únicamente para él, de caso contrario, se elimina para ambos actores.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario selecciona la opción “Ver solicitudes” en el menú de horas extras.</li> <li>3. El sistema carga las solicitudes dependiendo del rol.</li> <li>4. El usuario abre una solicitud.</li> <li>5. El usuario presiona el botón “Borrar solicitud”, el sub flujo No. 1: Borrar solicitud es ejecutado</li> </ol>	

<b>Sub Flujos</b>	
<b>Sub Flujo No. 1:</b> Borrar solicitud	El sistema detecta que usuario está borrando la solicitud, si es un empleado, la solicitud será borrada únicamente para él, en caso contrario, la solicitud será borrada tanto para el empleado como para el administrador.
<b>Flujos Alternos</b>	
<b>Flujo Alternativo No. 1</b>	Si el sistema no puede borrar la solicitud del empleado, se mostrará una imagen o texto indicando que no se pudo borrar dicha solicitud.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema borra la solicitud de horas extras con éxito dependiendo del rol que posea el usuario.	

Fuente: Elaboración propia.

Tabla 22. Caso de uso 18 - Nueva solicitud de vacaciones

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 18	<b>Nombre del caso de uso:</b> Nueva solicitud de vacaciones
<b>Fecha de elaboración:</b>	31/05/2020
<b>Descripción caso de uso:</b>	Permite al empleado crear una nueva solicitud de vacaciones.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Empleado.
<b>Precondiciones:</b>	Tener conexión a internet, haber ingresado en el sistema.
<b>Flujo básico del caso de uso</b>	
El empleado ingresa a la pantalla Nueva solicitud de vacaciones, se carga un formulario el cual debe completar, una vez completado presiona el botón Solicitar vacaciones.	
<b>Detalle paso a paso del flujo básico</b>	
1. El usuario ingresa al sistema.	

<ol style="list-style-type: none"> <li>2. El usuario selecciona la opción “Nueva solicitud” en el menú de vacaciones.</li> <li>3. El sistema carga un formulario con los campos necesarios para realizar la solicitud.</li> <li>4. El usuario completa el formulario.</li> <li>5. El usuario presiona el botón “Crear solicitud”.</li> <li>6. El sistema registra la solicitud.</li> </ol>	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	
<b>Flujo Alternativo No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrara una advertencia indicando que los datos ingresados no cumplen el formato necesario.
<b>Flujo Alternativo No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia y solicita al usuario que complete los campos requeridos.
<b>Flujo Alternativo No. 3</b>	El sistema comprueba si el empleado posee suficientes días disponibles para solicitar vacaciones, si excede la cantidad se mostrará una alerta indicando que no posee suficientes días.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema registra la solicitud de vacaciones con éxito.	

Fuente: Elaboración propia.

Tabla 23. Caso de uso 19 - Ver solicitudes vacaciones

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 19	<b>Nombre del caso de uso:</b> Ver solicitudes vacaciones

<b>Fecha de elaboración:</b>	31/05/2020
<b>Descripción caso de uso:</b>	Permite al administrador ver las solicitudes de vacaciones.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador.
<b>Precondiciones:</b>	Tener conexión a internet, haber ingresado en el sistema.
<b>Flujo básico del caso de uso</b>	
El administrador ingresa a la pantalla Ver solicitudes de vacaciones, se cargan todas las solicitudes de vacaciones que han realizado los empleados.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario selecciona la opción “Ver solicitudes” en el menú de vacaciones.</li> <li>3. El sistema carga las solicitudes de vacaciones.</li> </ol>	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrara una advertencia indicando que los datos ingresados no cumplen el formato necesario.
<b>Flujo Alterno No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia y solicita al usuario que complete los campos requeridos.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema registra la solicitud de vacaciones con éxito.	

Fuente: Elaboración propia.

Tabla 24. Caso de uso 20 - Registrar voto

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 20	<b>Nombre del caso de uso:</b> Registrar voto
<b>Fecha de elaboración:</b>	02/06/2020
<b>Descripción caso de uso:</b>	Permite a los clientes registrar un voto para un empleado con rol de chofer
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Cliente.
<b>Precondiciones:</b>	Tener conexión a internet.
<b>Flujo básico del caso de uso</b>	
El cliente ingresa a la pantalla habilitada para registrar su voto, esta carga un formulario el cual debe ser completado por el mismo, una vez completado, este presiona el botón Enviar formulario y el voto es registrado en el sistema.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El cliente ingresa a la página pública habilitada.</li> <li>2. El sistema carga un formulario con los campos necesarios para registrar la calificación.</li> <li>3. El cliente completa el formulario.</li> <li>4. El sistema registra la calificación.</li> <li>5. El sistema impide que el cliente vuelva a realizar la calificación durante cierto tiempo.</li> </ol>	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrara una advertencia indicando que los datos ingresados no cumplen el formato necesario.
<b>Flujo Alterno No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia y

	solicita al cliente que complete los campos requeridos.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema registra la calificación para el empleado chofer con éxito	

Fuente: Elaboración propia.

Tabla 25. Caso de uso 21 - Ver calidad servicio de choferes

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 21	<b>Nombre del caso de uso:</b> Ver calidad servicio de choferes
<b>Fecha de elaboración:</b>	16/06/2020
<b>Descripción caso de uso:</b>	Permite al administrador ver la calidad del servicio que brindan los choferes según los clientes
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador.
<b>Precondiciones:</b>	Tener conexión a internet.
<b>Flujo básico del caso de uso</b>	
El administrador ingresa a la página Calidad servicio de choferes y el sistema carga un ranking de choferes basándose en los formularios enviados por los clientes.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario da clic en la opción Calidad servicio choferes en el menú Control disciplinario.</li> <li>3. El sistema carga una tabla una tabla con los choferes y el puntaje asignado según las calificaciones que los clientes le otorgaron.</li> <li>4. El usuario selecciona uno empleado dando clic.</li> <li>5. El sistema carga otra tabla mostrando información sobre la opinión que tuvo cada cliente.</li> </ol>	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	

<b>Flujo Alterno No. 1</b>	Si ningún chofer no ha recibido ninguna calificación, el sistema mostrara un texto indicando que los choferes no poseen calificaciones.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema carga el ranking de choferes basándose en el servicio que brindaron.	

Fuente: Elaboración propia.

Tabla 26. Caso de uso 22 - Amonestaciones

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 22	<b>Nombre del caso de uso:</b> Amonestaciones
<b>Fecha de elaboración:</b>	16/06/2020
<b>Descripción caso de uso:</b>	Permite al administrador crear amonestaciones y al empleado ver aquellas que le han sido asignadas.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador y empleado.
<b>Precondiciones:</b>	Tener conexión a internet.
<b>Flujo básico del caso de uso</b>	
El actor ingresa a la pantalla de amonestaciones, dependiendo del actor se mostrará un tipo de pantalla diferente. Para un empleado el sistema carga sus amonestaciones y, por otro lado, a un administrador se le cargará un formulario y una tabla.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario da clic en la opción Amonestaciones en el menú Control disciplinario.</li> <li>3. Si el usuario es administrador, el sub flujo No. 1: Crear amonestación es ejecutado.</li> <li>4. Si el usuario es un empleado, el sub flujo No. 2: Ver amonestaciones asignadas es ejecutado.</li> </ol>	
<b>Sub Flujos</b>	

<b>Sub flujo No. 1:</b> Crear amonestación	El sistema carga un formulario el cual debe completar para registrar una nueva amonestación además de una tabla con todas las amonestaciones registradas. Cuando este completa el formulario y presiona el botón Asignar amonestación, esta se registra en el sistema y se carga en la tabla de amonestaciones.
<b>Sub flujo No. 2:</b> Ver amonestaciones asignadas	El sistema carga todas las amonestaciones que le han sido asignadas.,
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrara una advertencia indicando que los datos ingresados no cumplen el formato necesario.
<b>Flujo Alterno No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia y solicita al cliente que complete los campos requeridos.
<b>Flujo Alterno No. 3</b>	Si a un empleado no se la han asignado amonestaciones el sistema mostrará un texto indicando que no ha recibido ninguna amonestación.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema registra la amonestación o carga la misma de manera exitosa.	

Fuente: Elaboración propia.

Tabla 27. Caso de uso 23 - Felicitaciones

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>
---

<b>Número de caso de uso:</b> 23	<b>Nombre del caso de uso:</b> Felicitaciones
<b>Fecha de elaboración:</b>	18/06/2020
<b>Descripción caso de uso:</b>	Permite al administrador crear felicitaciones y al empleado ver aquellas que le han sido asignadas.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador y empleado.
<b>Precondiciones:</b>	Tener conexión a internet.
<b>Flujo básico del caso de uso</b>	
El actor ingresa a la pantalla de Felicitaciones, dependiendo del actor se mostrará un tipo de pantalla diferente. Para un empleado el sistema carga sus felicitaciones y, por otro lado, a un administrador se le cargará un formulario y una tabla.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario da clic en la opción Felicitaciones en el menú Control disciplinario.</li> <li>3. Si el usuario es administrador, el sub flujo No. 1: Crear felicitación es ejecutado.</li> <li>4. Si el usuario es un empleado, el sub flujo No. 2: Ver felicitaciones asignadas es ejecutado.</li> </ol>	
<b>Sub Flujos</b>	
<b>Sub flujo No. 1:</b> Crear felicitación	El sistema carga un formulario el cual debe completar para registrar una nueva felicitación además de una tabla con todas las felicitaciones registradas. Cuando este completa el formulario y presiona el botón Asignar felicitación, esta se registra en el sistema y se carga en la tabla de felicitaciones.
<b>Sub flujo No. 2:</b> Ver felicitaciones asignadas	El sistema carga todas las felicitaciones que le han sido asignadas.,
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrara una advertencia indicando que los datos ingresados no cumplen el formato necesario.

<b>Flujo Alterno No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia y solicita al cliente que complete los campos requeridos.
<b>Flujo Alterno No. 3</b>	Si a un empleado no se la han asignado ninguna felicitación el sistema mostrará un texto indicando que no ha recibido ninguna felicitación.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema registra la felicitación o carga la misma de manera exitosa.	

Fuente: Elaboración propia.

Tabla 28. Caso de uso 24 - Cálculo salarial

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 24	<b>Nombre del caso de uso:</b> Cálculo salarial
<b>Fecha de elaboración:</b>	19/06/2020
<b>Descripción caso de uso:</b>	Permite al administrador consultar el salario de los empleados.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador y empleado.
<b>Precondiciones:</b>	Tener conexión a internet. Ser administrador o secretario.
<b>Flujo básico del caso de uso</b>	
El actor ingresa a la pantalla de Cálculo salarial, se cargan los empleados y posteriormente se selecciona uno y presiona el botón Calcular salario.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario da clic en la opción Cálculo salarial en el menú Planilla.</li> <li>3. El sistema carga a todos los empleados y un campo para ingresar el mes.</li> <li>4. El usuario elije el mes y presiona el botón Consultar.</li> </ol>	

5. El sistema carga la información del empleado relacionada al salario.	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrara una advertencia indicando que los datos ingresados no cumplen el formato necesario.
<b>Flujo Alterno No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia y solicita al cliente que complete los campos requeridos.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema carga la información sobre el salario del empleado de manera exitosa.	

Fuente: Elaboración propia.

Tabla 29. Caso de uso 25 - Aumentos salariales

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 25	<b>Nombre del caso de uso:</b> Aumentos salariales
<b>Fecha de elaboración:</b>	20/06/2020
<b>Descripción caso de uso:</b>	Permite al administrador realizar aumentos salariales para los empleados.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador.
<b>Precondiciones:</b>	Tener conexión a internet.
<b>Flujo básico del caso de uso</b>	
El administrador ingresa a la página Aumentos salariales, el sistema carga un formulario y el administrador lo completa, finalmente presiona el botón Aplicar aumento y se registra.	
<b>Detalle paso a paso del flujo básico</b>	

<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario da clic en la opción Aumentos salariales en el menú Planilla.</li> <li>3. El sistema carga a todos los aumentos realizados y un formulario.</li> <li>4. El usuario completa el formulario.</li> <li>5. El usuario presiona el botón Aplicar aumento.</li> <li>6. El sistema registra el nuevo aumento salarial para el empleado seleccionado en el formulario.</li> <li>7. El sistema refleja ese nuevo registro en la tabla con los aumentos salariales realizados.</li> </ol>	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrara una advertencia indicando que los datos ingresados no cumplen el formato necesario.
<b>Flujo Alterno No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia y solicita al cliente que complete los campos requeridos.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema registra el aumento salarial de manera exitosa.	

Fuente: Elaboración propia.

Tabla 30. Caso de uso 26 - Retenciones salariales

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 26	<b>Nombre del caso de uso:</b> Retenciones salariales
<b>Fecha de elaboración:</b>	20/06/2020

<b>Descripción caso de uso:</b>	Permite al administrador realizar retenciones salariales para los empleados.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador.
<b>Precondiciones:</b>	Tener conexión a Internet.
<b>Flujo básico del caso de uso</b>	
El administrador ingresa a la página Retenciones salariales, el sistema carga un formulario y el administrador lo completa, finalmente presiona el botón Aplicar retención y este se registra.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario da clic en la opción Retenciones salariales en el menú Planilla.</li> <li>3. El sistema carga a todas las retenciones realizadas y un formulario para crear una nueva.</li> <li>4. El usuario completa el formulario.</li> <li>5. El usuario presiona el botón Aplicar retención.</li> <li>6. El sistema registra la nueva retención salarial para el empleado seleccionado en el formulario.</li> <li>7. El sistema refleja ese nuevo registro en la tabla con las retenciones salariales realizadas.</li> </ol>	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrará una advertencia indicando que los datos ingresados no cumplen el formato necesario.
<b>Flujo Alterno No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia y solicita al cliente que complete los campos requeridos.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema registra la nueva retención salarial para el empleado de manera exitosa.	

Fuente: Elaboración propia.

Tabla 31. Caso de uso 27 - Incapacidades

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 27	<b>Nombre del caso de uso:</b> Incapacidades
<b>Fecha de elaboración:</b>	21/06/2020
<b>Descripción caso de uso:</b>	Permite al administrador realizar incapacidades para los empleados.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador.
<b>Precondiciones:</b>	Tener conexión a internet.
<b>Flujo básico del caso de uso</b>	
El administrador ingresa a la página Incapacidades, el sistema carga un formulario y el administrador lo completa, finalmente presiona el botón Aplicar retención y este se registra.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario da clic en la opción Incapacidades en el menú Planilla.</li> <li>3. El sistema carga a todas las incapacidades registradas y un formulario para crear una nueva.</li> <li>4. El usuario completa el formulario.</li> <li>5. El usuario presiona el botón Confirmar.</li> <li>6. El sistema registra la nueva incapacidad para el empleado seleccionado en el formulario.</li> <li>7. El sistema refleja ese nuevo registro en la tabla con las incapacidades registradas.</li> </ol>	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrará una advertencia indicando que los datos ingresados no cumplen el formato necesario.
<b>Flujo Alterno No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia y

	solicita al cliente que complete los campos requeridos.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema registra la nueva incapacidad para el empleado de manera exitosa.	

Fuente: Elaboración propia.

Tabla 32. Caso de uso 28 - Viáticos

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 28	<b>Nombre del caso de uso:</b> Viáticos
<b>Fecha de elaboración:</b>	21/06/2020
<b>Descripción caso de uso:</b>	Permite al administrador asignar viáticos para los empleados.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador.
<b>Precondiciones:</b>	Tener conexión a internet.
<b>Flujo básico del caso de uso</b>	
El administrador ingresa a la página Viáticos, el sistema carga un formulario y el administrador lo completa, finalmente presiona el botón Otorgar viáticos y este se registra.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario da clic en la opción Viáticos en el menú Planilla.</li> <li>3. El sistema carga a todas los viáticos registrados y un formulario para crear uno nuevo.</li> <li>4. El usuario completa el formulario.</li> <li>5. El usuario presiona el botón Otorgar viáticos.</li> <li>6. El sistema registra los nuevos viáticos para el empleado seleccionado en el formulario.</li> <li>7. El sistema refleja ese nuevo registro en la tabla con los viáticos registrados.</li> </ol>	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	

<b>Flujo Alternativo No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrara una advertencia indicando que los datos ingresados no cumplen el formato necesario.
<b>Flujo Alternativo No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia y solicita al cliente que complete los campos requeridos.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema registra los nuevos viáticos para el empleado de manera exitosa.	

Fuente: Elaboración propia.

Tabla 33. Caso de uso 29 - Nueva tarea

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 29	<b>Nombre del caso de uso:</b> Nueva tarea
<b>Fecha de elaboración:</b>	23/06/2020
<b>Descripción caso de uso:</b>	Permite al administrador asignar una tarea a un empleado.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador.
<b>Precondiciones:</b>	Tener conexión a internet.
<b>Flujo básico del caso de uso</b>	
El administrador ingresa a la página Nueva tarea, el sistema carga un formulario y el administrador lo completa, finalmente presiona el botón Asignar tarea y esta se registra sin problemas.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario da clic en la opción Nueva tarea en el menú Tareas.</li> <li>3. El sistema carga un formulario para crear una nueva tarea.</li> <li>4. El usuario completa el formulario.</li> </ol>	

5. El usuario presiona el botón Asignar tarea.	
6. El sistema registra la nueva tarea para el empleado seleccionado en el formulario.	
<b>Sub Flujos</b>	
<b>Flujos Alternos</b>	
<b>Flujo Alternativo No. 1</b>	Si alguno de los campos del formulario no cumple con el formato correcto, el sistema mostrara una advertencia indicando que los datos ingresados no cumplen el formato necesario.
<b>Flujo Alternativo No. 2</b>	Si todos o alguno de los campos se encuentran vacíos, el sistema mostrará una advertencia y solicita al cliente que complete los campos requeridos.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema registra la nueva tarea para el empleado de manera exitosa.	

Fuente: Elaboración propia.

Tabla 34. Caso de uso 30 - Ver tareas

<b>Prototipo Funcional para la Gestión de Recursos Humanos en la Asociación de Desarrollo Integral de San Juan Puriscal</b>	
<b>Número de caso de uso:</b> 30	<b>Nombre del caso de uso:</b> Ver tareas
<b>Fecha de elaboración:</b>	29/06/2020
<b>Descripción caso de uso:</b>	Permite al administrador y al empleado ver las tareas.
<b>Autor caso de uso:</b>	Roberto Duarte
<b>Actores relacionados:</b>	Administrador y empleado.
<b>Precondiciones:</b>	Tener conexión a internet.
<b>Flujo básico del caso de uso</b>	
El actor ingresa a la página Ver tarea, en caso de ser administrador el sistema carga todas las tareas asignadas a los empleados, en caso contrario, el sistema carga aquellas tareas que le han sido	

asignadas, finalmente el administrador puede borrar la tarea y ambos actores pueden ver más información sobre esta.	
<b>Detalle paso a paso del flujo básico</b>	
<ol style="list-style-type: none"> <li>1. El usuario ingresa al sistema.</li> <li>2. El usuario da clic en la opción Ver tareas en el menú Tareas.</li> <li>3. El sistema carga todas las tareas dependiendo de si es administrador o empleado.</li> <li>4. Si el usuario presión el botón Mas información, el sub flujo No. 1: Mas información es ejecutado.</li> <li>5. Si el usuario presión el botón Eliminar tarea, el sub flujo No. 2: Eliminar tarea es ejecutado.</li> </ol>	
<b>Sub Flujos</b>	
<b>Sub Flujo No. 1:</b> Mas información	El sistema abre una ventana mostrando información más detallada sobre la tarea asignada.
<b>Sub Flujo No. 2:</b> Eliminar tarea	El sistema abre una ventana solicitando la confirmación del usuario para proceder a eliminar la tarea, en caso de que presione Confirmar, esta tarea es eliminada.
<b>Flujos Alternos</b>	
<b>Flujo Alterno No. 1</b>	Si no se han asignado tareas a los empleados, el sistema mostrará un texto indicando que no se han asignado tareas.
<b>Requerimientos especiales</b>	
No aplica.	
<b>Post-Condiciones</b>	
El sistema carga todas las tareas asignadas a empleados de manera exitosa.	

Fuente: Elaboración propia.

## **Análisis detallado del Software desarrollado**

En este apartado se enumeran los requerimientos específicos para el desarrollo del prototipo funcional, basándose en los problemas hallados y las necesidades planteadas por la empresa.

**Req. 1 Módulo de seguridad:** este módulo permitirá el ingreso al sistema utilizando el correo electrónico y una contraseña como credenciales, esta última debe incluir un mínimo de 10 caracteres alfanuméricos, además, deberá de restringir el acceso y uso de los diferentes módulos del sistema dependiendo del rol que se posea, el rol de administrador tendrá un acceso completo sin restricciones y, por otro lado, el rol de empleado tendrá acceso a módulos como el de vacaciones, permisos y control de calidad, pero únicamente podrán realizar acciones como leer o crear, esto dependerá del objetivo de cada módulo. El sistema únicamente tendrá una pantalla de inicio de sesión por lo que el registro de nuevos usuarios será controlado por aquellos que poseen el rol de administrador.

**Req. 2 Módulo de mantenimiento:** este módulo será responsable de crear nuevos trabajadores, ya sean permanentes o temporales. Ambos tipos de empleados poseen información como el nombre completo, cedula, teléfono, fecha de contratación, correo electrónico, salario por hora, cantidad de horas que trabajará por día y el cargo que desempeñará. El rol de administrador podrá gestionar todos los permisos de salida que solicitan los empleados, es decir, podrá visualizarlos, aprobarlos, rechazarlos y eliminarlos.

Por otro lado, el módulo será capaz de crear y asignar tareas a los empleados, para los empleados que posean un cargo de chofer, se les asignará una semana con datos preestablecidos como el horario que debe cumplir para cada día, el bus que deberá operar y el tipo de servicio que deberá realizar.

**Req. 3 Módulo de consultas:** por medio de este módulo se podrá consultar información más detallada y específica de los trabajadores disponibles en la empresa, ya sean permanentes o temporales, también, se podrá consultar la cantidad de días vacacionales disponibles que posee cada empleado. Este módulo se encontrará implícito en los demás módulos por lo que no es necesario la utilización de un menú específico en el que se realicen todas las consultas necesarias para obtener cierta información en concreto.

**Req. 4 Módulo de reportes:** este módulo de se hará cargo de generar un reporte detallado mensual de diferentes aspectos de como lo son los pagos de salario realizados, amonestaciones realizadas y tareas asignadas. Los reportes tendrán la opción de generar un archivo PDF con toda la información correspondiente. Entre los detalles incluidos en este archivo, se encuentran datos como el nombre completo del empleado, deducciones aplicadas y horas extras laboradas con sus respectivos montos, fecha del pago, salario bruto y neto. En cuanto a las amonestaciones y tareas, estas mostraran fechas en las que se aplicaron y el motivo por el que se aplicó.

**Req. 5 Módulo de planilla:** este módulo será responsable de calcular los pagos para cada empleado, deberá aplicar de forma automática las horas extras laboradas y deducciones necesarias, como por ejemplo las cargas sociales. También deberá de existir un registro de todos los cálculos realizados.

Por otro lado, el módulo deberá ser capaz de calcular el pago para los choferes de dos diferentes maneras, la primera normalmente no posee ningún cambio y es para el tipo de servicio público, la segunda presenta dos cambios, el primero se encuentra en las horas extras, estas no son aplicadas, el segundo es el monto ganado por hora, todo lo demás se mantiene normalmente. Esta segunda manera pertenece al tipo de servicio privado. De igual manera, los reportes generados deberán mostrar esta diferencia.

**Req. 6 Módulo de vacaciones:** el presente módulo permitirá a los empleados solicitar vacaciones y un administrador se encargará de aceptar o rechazar dicha solicitud. En el caso de los empleados, al ingresar al módulo se deberá mostrar la cantidad de vacaciones disponibles y al realizar la solicitud se deberá detallar la fecha de salida y fecha de ingreso o retorno. El sistema deberá ser capaz de limitar la cantidad de días solicitado por el empleado, es decir, no podrá solicitar vacaciones si estas son mayores a la cantidad de días disponibles.

**Req. 7 Módulo de control de calidad:** mediante este módulo se determinará si un empleado con cargo de chofer es más propenso a asignarle mayor cantidad de servicios turísticos, esto mediante un sistema de calificación o puntos en donde los clientes llenan un formulario otorgando una calificación o puntuación al empleado. El formulario se encontrará habilitado en una ruta específica del sistema, esta ruta o enlace es de acceso público por lo que no es necesario estar registrado e iniciar sesión en el sistema.

Además, este módulo mostrará información importante como la calificación asignada al chofer, nombre del cliente, comentario y la fecha en que realizó el formulario con el propósito de tener un mejor control.

La siguiente tabla muestra la relación entre los módulos establecidos en las proyecciones, y los requerimientos planteados por miembros de la junta directiva de la Asociación de Desarrollo Integral de San Juan Puriscal.

Tabla 35. Matriz de requerimientos

<b>Módulos</b>	<b>Requerimientos</b>
Seguridad	Req 1
Mantenimiento	Req 1, Req 2
Consultas	Req 1, Req 3
Reportes	Req 1, Req 4
Planilla	Req 1, Req 5
Vacaciones	Req 1, Req 6
Control de calidad	Req 1, Req 7

Fuente: Elaboración propia

### Análisis detallado del hardware requerido

A continuación, se muestra información detallada del hardware utilizado para el desarrollo de este prototipo funcional, así como los diferentes servicios necesarios para una futura implementación.

Tabla 36. Hardware utilizado para el desarrollo

Nombre	Especificaciones técnicas	Cantidad	Costo
Hacer Aspire E 15	Procesador Intel Core i5 Tarjeta gráfica NVIDIA GeForce 940M Memoria RAM: 8GB Almacenamiento: Disco estado sólido de 240GB	1	Alrededor de \$200.000

Fuente: Elaboración propia

Hoy día el costo de esta unidad portátil se encuentra devaluado debido a que la adquisición de la misma supera los cuatro años. También cabe aclarar que esta unidad recibió actualizaciones en su hardware como lo es el caso del almacenamiento el cual se cambió por un disco de estado sólido, este cambio fue realizado antes de iniciar con el desarrollo de este prototipo.

Tabla 37. Servicios para puesta en producción

Nombre	Especificaciones técnicas	Plataforma	Costo
Heroku	Almacenamiento de 512MB Memoria RAM de 1GB	Node.js	\$0
Heroku	Almacenamiento de 10GB o más.	MySQL	\$29,000 según el tipo de cambio al mes de julio del 2020

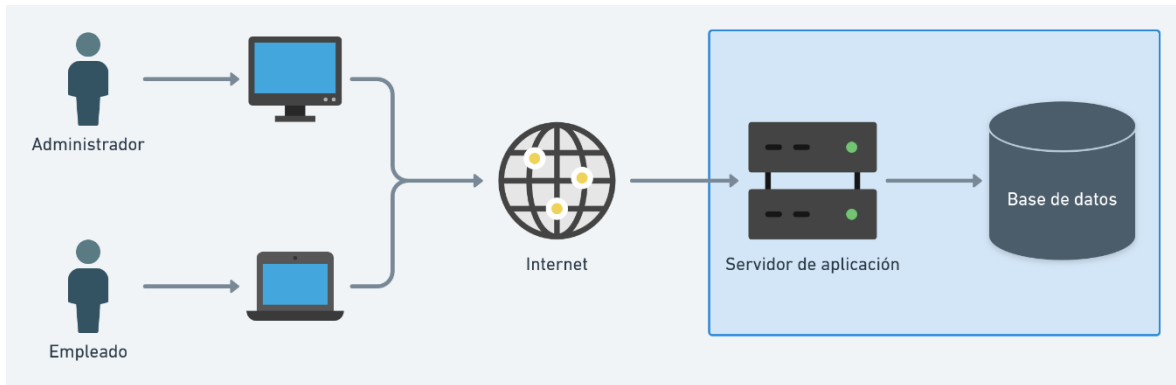
Fuente: Elaboración propia.

## Análisis detallado de las telecomunicaciones

Es necesario contar con una red LAN, ya sea en las instalaciones principales de la empresa o en los hogares de los empleados y administradores, en esta red necesitará un router el cual se conectará a internet para a través de este hacer uso de la aplicación.

## Diseño

En esta sección se definen las diferentes arquitecturas, diagramas e interfaces gráficas



de usuario. Este apartado corresponde a la segunda etapa de la metodología de ciclo de vida, se podrá obtener un punto de vista más claro sobre el funcionamiento del prototipo.

## Arquitectura del sistema

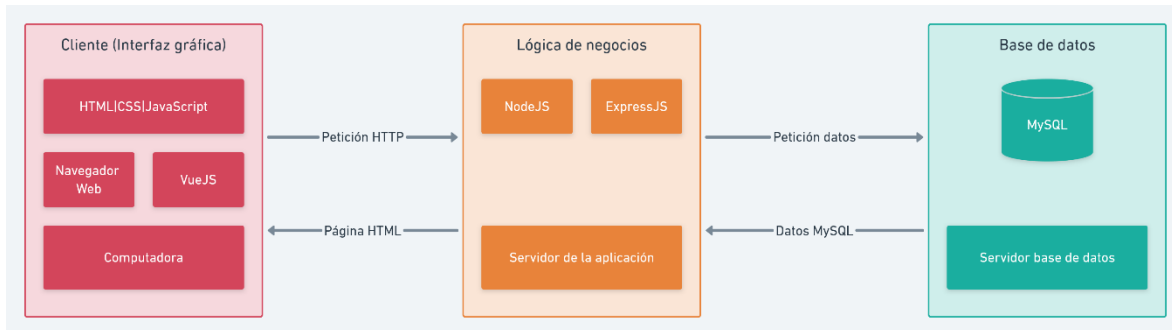
Figura 11. Arquitectura del sistema

Fuente: Elaboración propia.

La arquitectura presentada en la imagen anterior permite el acceso al sistema a los diferentes tipos de usuarios según el rol que estos desempeñan, estos necesitarán hacer uso de un computador el cual cuente con acceso a Internet, esto con el fin de poder ejecutar la aplicación la cual es responsable de realizar la comunicación con la base de datos la cual almacena toda la información de la empresa, y el usuario, el cual podrá acceder a esta.

## Arquitectura del software

Figura 12. Arquitectura de software



Fuente: Elaboración propia.

El usuario hace uso de la aplicación mediante un computador con acceso a internet, esta aplicación es ejecutada por medio de un navegador web el cual le muestra la interfaz gráfica que le permitirá hacer uso de las diferentes funcionalidades. Como se aprecia en la imagen, el cliente, es decir, la aplicación, se compone de las tecnologías estándares en el desarrollo web las cuales son HTML, CSS y JavaScript, para esta última se hace uso de la biblioteca Vue.js que permite construir interfaces gráficas más complejas, escalables y eficientes. Por último, dependiente del rol que posea el usuario se mostrarán mayor o menor cantidad de pantallas, limitando así ciertas funcionalidades.

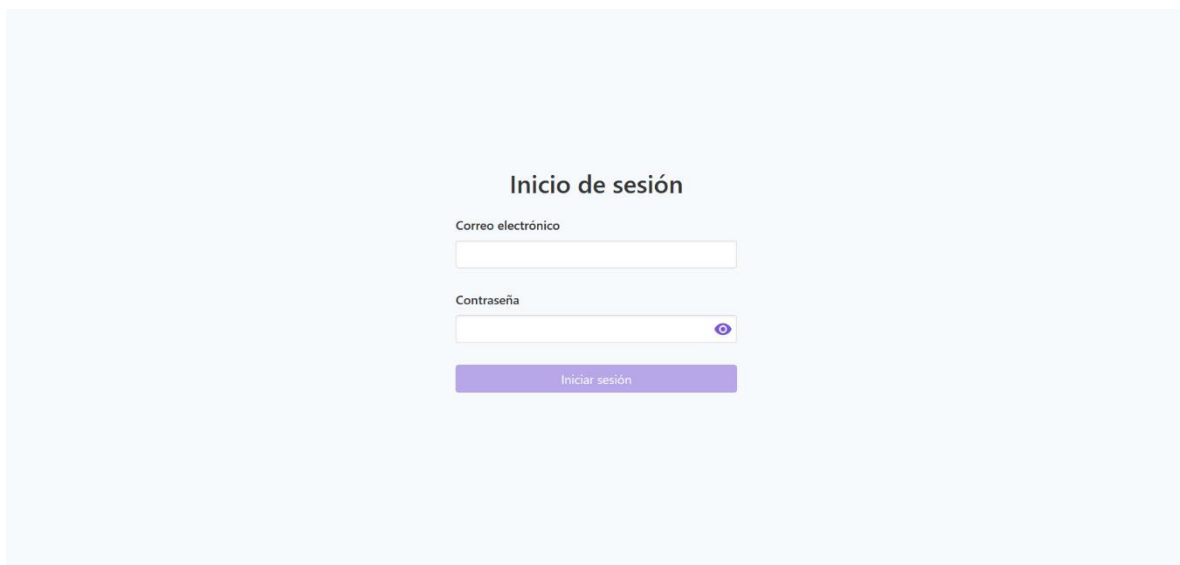
En cuanto al servidor de la aplicación que se encuentra en ejecución gracias a las capacidades con las que cuenta Node.js y Express, expone diferentes rutas en las que el cliente puede realizar solicitudes o peticiones HTTP que permiten conocer si un usuario ha iniciado sesión y el rol que este posee para así eliminar las funcionalidades del sistema. Claro está que esta capa lógica es la encargada de realizar peticiones a la base de datos y enviar estos resultados al cliente, en palabras más simples, establece una comunicación entre el cliente y la base de datos.

Por último, la capa de base datos es encargada de estructurar la información y almacenarla para que esta pueda ser accedida en un futuro mediante el flujo de peticiones o solicitudes que inicia el usuario por medio del cliente, es decir, el navegador web.

## Diseño de Interfaces

A continuación, se muestran los diseños de las diferentes pantallas con las que cuenta el prototipo funcional, las imágenes presentadas corresponden a la interfaz gráfica que utiliza el sistema desarrollado.

Figura 13. Interfaz para el inicio de sesión

The image shows a login interface titled "Inicio de sesión" centered on a light blue background. Below the title, there are two input fields: "Correo electrónico" and "Contraseña". The "Contraseña" field has a small eye icon on its right side. Below the input fields is a purple button labeled "Iniciar sesión".

Inicio de sesión

Correo electrónico

Contraseña

Iniciar sesión

Fuente: Elaboración propia.

Este formulario es el que se mostrará a cada empleado una vez intenten acceder al sistema, deberán ingresar un correo electrónico y contraseña validas, seguidamente deberán presionar el botón “Iniciar sesión” para poder ingresar.

Figura 14. Interfaz para registrar nuevo empleado

The image shows a web application interface for registering a new employee. At the top, there is a navigation bar with links for 'Inicio', 'Empleados', 'Permisos', 'Horas extra', 'Planilla', 'Control disciplinario', 'Tareas', and 'Vacaciones'. On the right, there are links for 'Perfil' and 'Cerrar sesión'.

The main content area is divided into two primary sections:

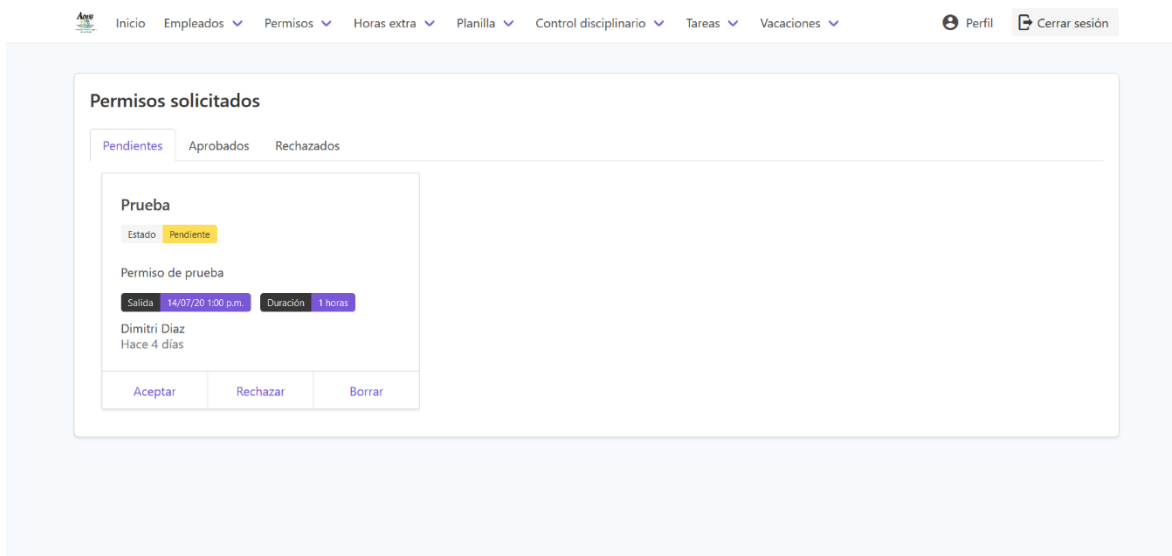
- Información básica:** This section is titled 'Completa los datos personales del nuevo empleado'. It contains several input fields:
  - Número de cédula:** A text input field.
  - Nombre:** A text input field.
  - Primer apellido:** A text input field.
  - Segundo apellido:** A text input field.
  - Fecha de nacimiento:** A date selection field with the placeholder text 'Seleccionar fecha...'.
  - Correo electrónico:** A text input field.
  - Contraseña:** A text input field.
  - Confirmar contraseña:** A text input field.
- Información de contacto:** This section is titled 'Completa los datos de domicilio e ingresa números de teléfono'. It includes:
  - Provincia:** A dropdown menu with the placeholder 'Selecciona uno'.
  - Cantón:** A dropdown menu with the placeholder 'Selecciona uno'.
  - Distrito:** A dropdown menu with the placeholder 'Selecciona uno'.
  - Dirección exacta:** A large text area for entering the address.
  - Números de teléfono:** A section with a 'Tipo' dropdown menu, a text input field, and an 'Agregar' button.
- Laboral:** This section is titled 'Completa los siguientes datos básicos importantes para la empresa'. It includes:
  - Tipo de empleado:** A dropdown menu with the placeholder 'Selecciona un tipo'.
  - Salario por hora:** A text input field with the value '0.00'.
  - Horas de trabajo por día:** A text input field.

At the bottom right of the form, there is a purple button labeled 'Guardar empleado'. A yellow note box is also present, stating: 'Nota: La contraseña debe de cumplir los siguientes parámetros: • Una longitud de 10 caracteres. • Únicamente debe contener letras y números. • Al menos una letra mayúscula y un número.'

Fuete: Elaboración propia.

Esta figura corresponde al formulario que un administrador requiere completar para registrar un nuevo empleado. Una vez completado, el usuario deberá presionar el botón “Guardar empleado” ubicado en la esquina inferior derecha, esto creará un nuevo registro del empleado y mostrará una pequeña notificación.

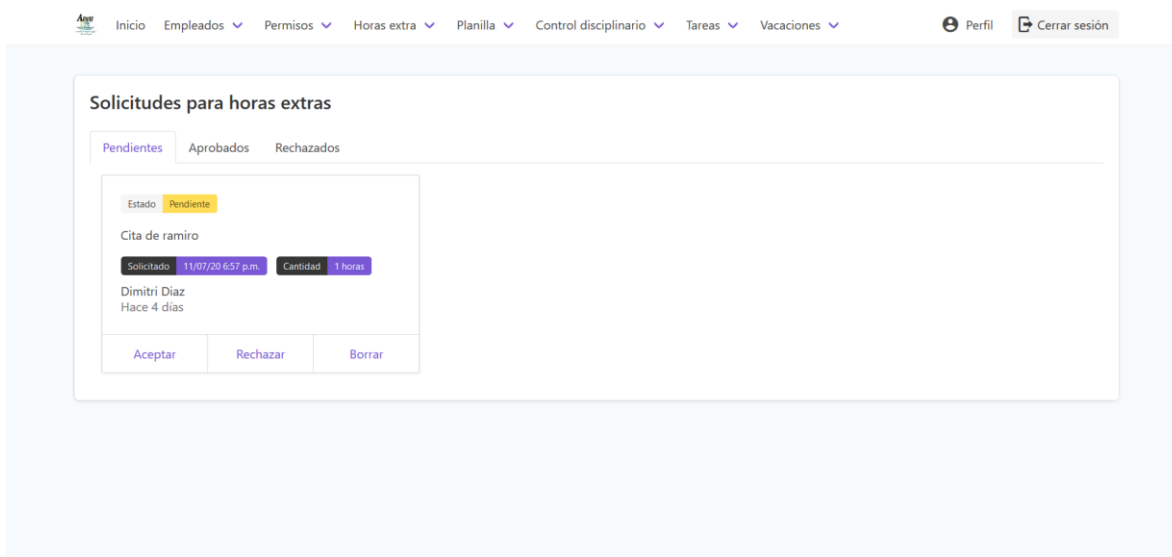
Figura 15. Interfaz para los permisos solicitados



Fuente: Elaboración propia.

Esta pantalla muestra todos los permisos que han sido solicitado por los empleados, estos son categorizados por pendientes, aprobado y rechazados. En la parte inferior de cada solicitud se encuentran botones que permiten aceptar, rechazar o borrar este, presionando uno de estos botones actualiza el permiso y muestra una pequeña notificación.

Figura 16. Interfaz para las horas extras solicitadas



Fuente: Elaboración propia.

Similar al diseño anterior, permite visualizar las solicitudes de horas extras que los empleados realizan, así mismo, se puede aceptar, rechazar o borrar dicha solicitud.

Figura 17. Interfaz para consultar pagos

**Consulta de pagos**

Selecciona el mes y a los empleados que quieres consultarles el pago.

Elije un mes

<input type="checkbox"/>	ID	Nombre empleado	Fecha de contrato	Tipo de empleado
<input type="checkbox"/>	1	Leonidas Quesada Jonhs	07/07/2020	Administrador
<input type="checkbox"/>	2	Dimitri Diaz Salas	23/02/2010	Chofer

Consultar

**Detalles**

El porcentaje de cargas sociales corresponde a un 10.5%.  
 Expande la fila para ver mas informacion.

Nothing here.

Fuente: Elaboración propia.

El diseño para la consulta de pagos se conforma de dos secciones, en la primera se muestran los empleados que se encuentra laborando en la empresa. Además, de un campo para ingresar el mes y el año; seleccionados estos datos y los empleados, se deberá presionar el botón “Consultar”, esto cargará la sección dos con los datos del salario para cada empleado seleccionado.

Figura 18. Interfaz para nueva tarea

**Nueva tarea**

Empleado  
Dimitri Diaz Salas

Titulo  
[ ]

Descripcion  
[ ]

Transporte

Tipo de servicio [ ] Vehiculo [ ]

Dias  
 Lunes  Martes  Miércoles  Jueves  Viernes  Sábado  Domingo

Asignar tarea

Fuente: Elaboración propia.

Los primeros tres campos del formulario se mostrarán para todos los tipos de empleados, en caso de que el empleado sea un chofer se habilitará un interruptor que dice “Transporte”. El cual permitirá asignar una tarea relacionada al servicio de transporte público o privado.

Figura 19. Interfaz para solicitar vacaciones

**Nueva solicitud de vacaciones**

DÍAS DISPONIBLES  
**0**

DÍAS SOLICITADOS  
**0**

DÍAS DISPONIBLES DESPUÉS DE LA SOLICITUD  
**0**

Calendar: Julio 2020  
 Selected date: 15

Solicitar vacaciones

Fuente: Elaboración propia.

Esta pantalla para solicitar vacaciones se compone de un calendario en el cual el empleado selecciona que días tomará para vacacionar. Los números a la izquierda cambiarán dinámicamente dependiendo de los días disponibles que tiene para vacacionar y la cantidad días que selecciona en el calendario.

Figura 20. Interfaz para nueva amonestación

Empleado	Motivo	Fecha ↓
<input type="text"/>		
Dimitri Diaz	Prueba. No se preocupe	11/07/2020

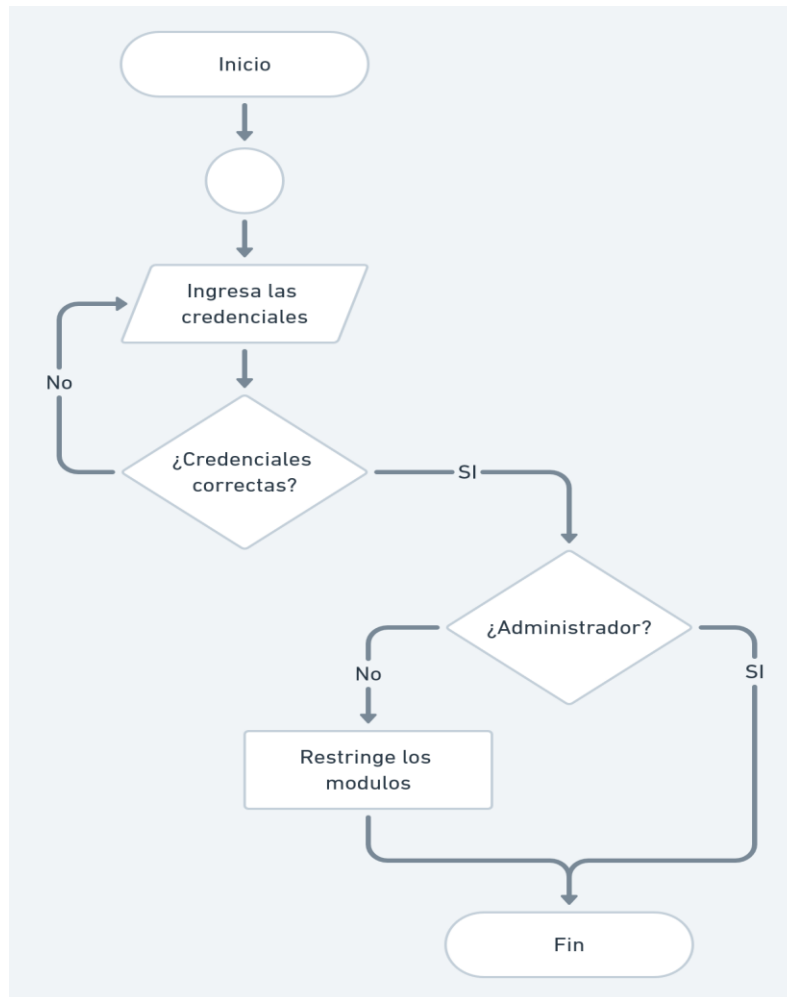
Fuente: Elaboración propia.

En esta pantalla se puede crear una nueva amonestación (sección a la izquierda) y ver las amonestaciones que se han registrado (sección a la derecha). Las amonestaciones pueden filtradas por el nombre del empleado.

## Diseño de procesos

Las figuras mostradas a continuación, muestran el flujo de los procesos que se debe seguir para entender el orden lógico a la hora de realizar las diferentes funcionalidades del prototipo funcional.

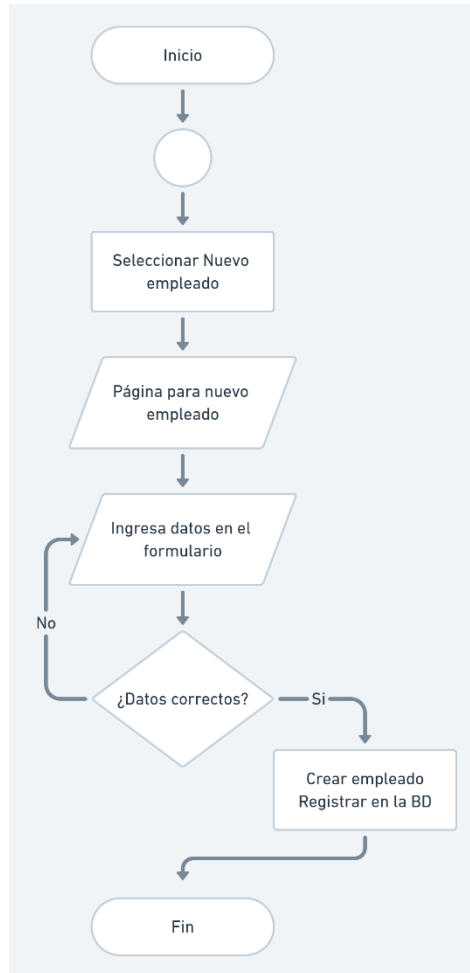
Figura 21. Proceso inicio de sesión



Fuente: Elaboración propia.

Este proceso requiere de credenciales correctas, es decir, estar registrado en el sistema, de lo contrario nunca se podrá acceder a este. Si el rol del usuario no es administrador el sistema restringe ciertos módulos del sistema.

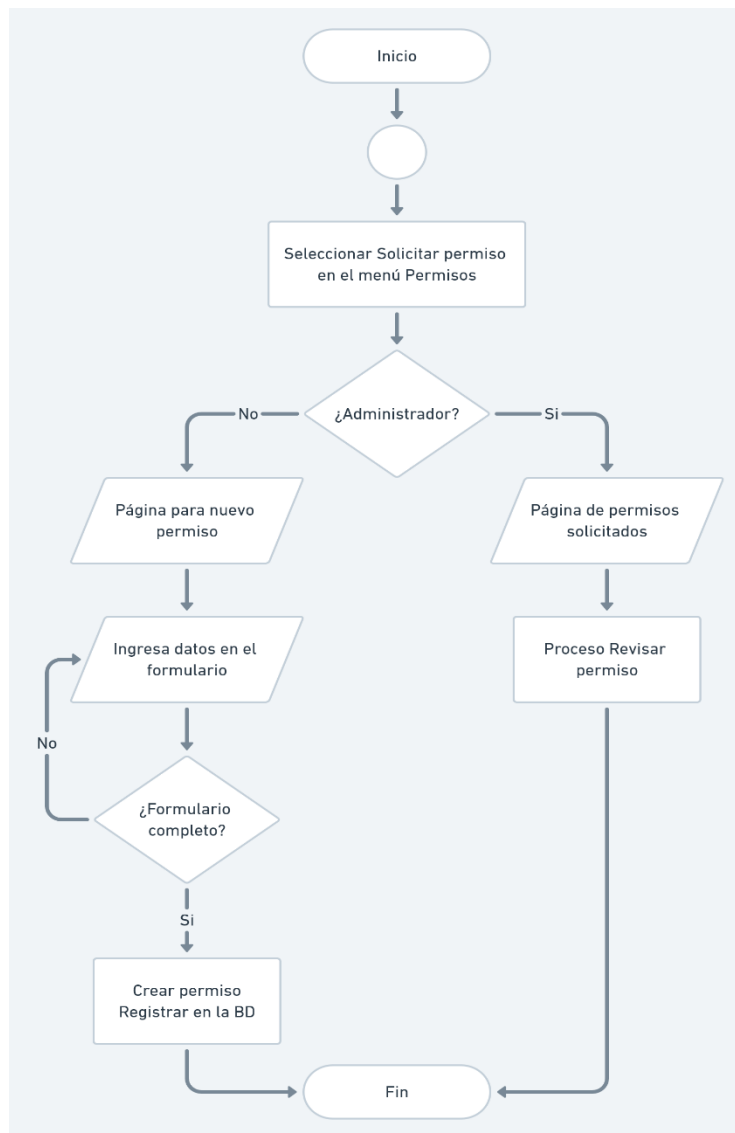
Figura 22. Proceso nuevo empleado



Fuente: Elaboración propia.

Este proceso requiere que los campos del formulario para registrar un nuevo empleado no cuenten con datos inválidos, únicamente será registrado en la base de datos si no se presenta ningún error.

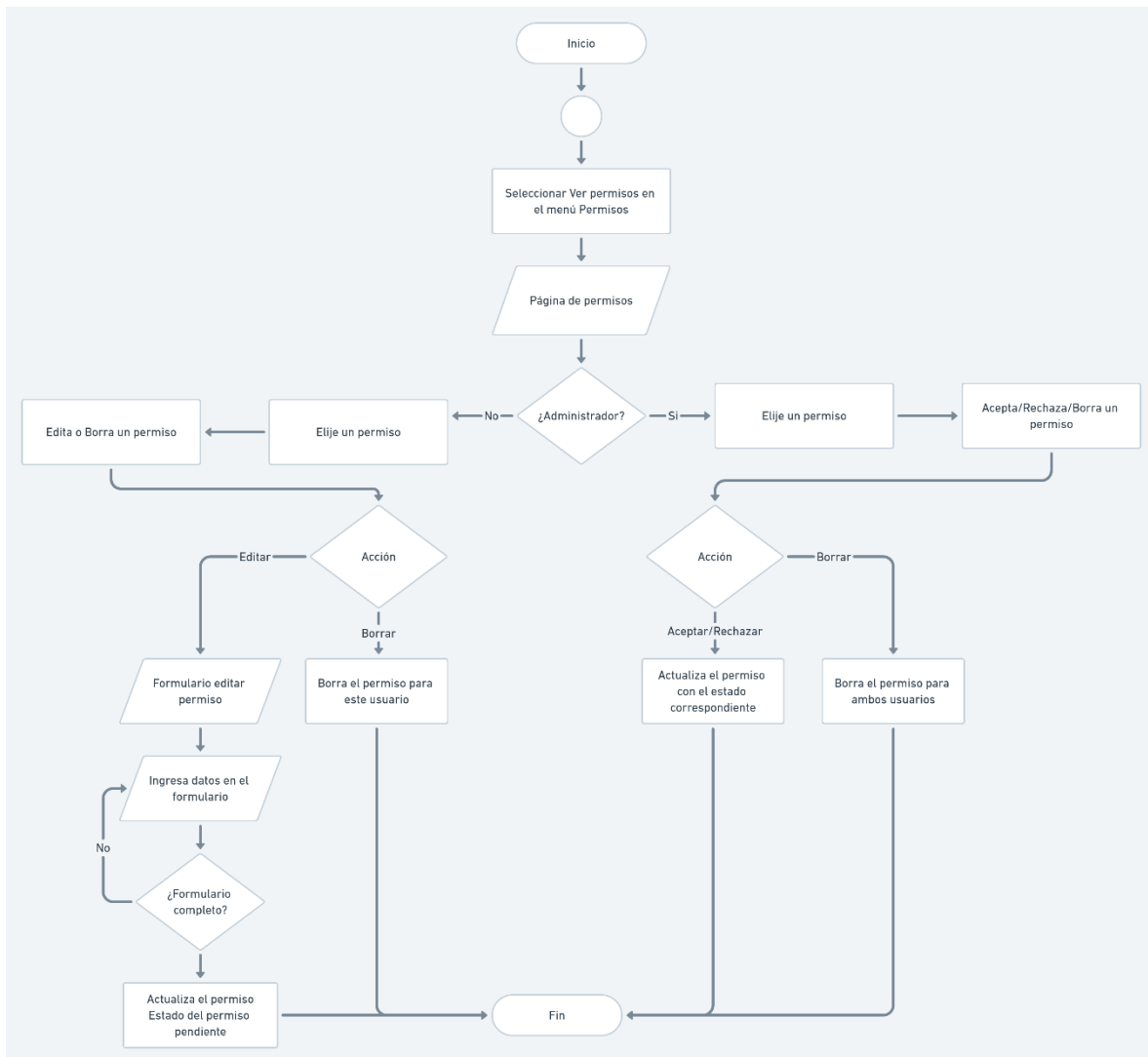
Figura 23. Proceso nuevo permiso



Fuente: Elaboración propia.

Para este proceso, los usuarios administradores son redirigidos a la página para ver los permisos, ejecutando el proceso para revisar un permiso. En caso de los empleados con un rol diferente deberán completar un formulario para registrar un nuevo permiso.

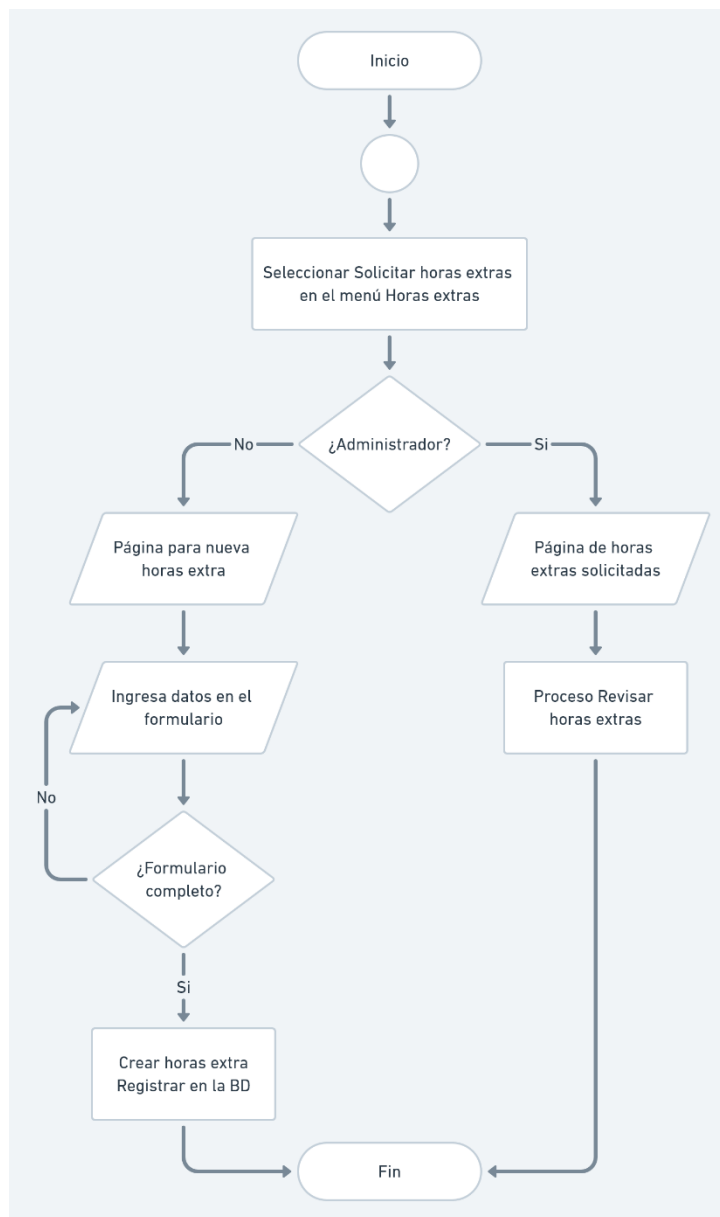
Figura 24. Proceso ver permisos



Fuente: Elaboración propia.

Los usuarios administradores elijen los permisos que desean aceptar, rechazar o borrar, en caso de desee borrar un permiso, este se borrará tanto para él como para el empleado que lo solicitó. En el caso de los empleados si editan una solicitud realizada, ya sea aceptada o rechazada, esta pasará automáticamente a pendiente, si borra ésta sólo será borrada para sí mismo.

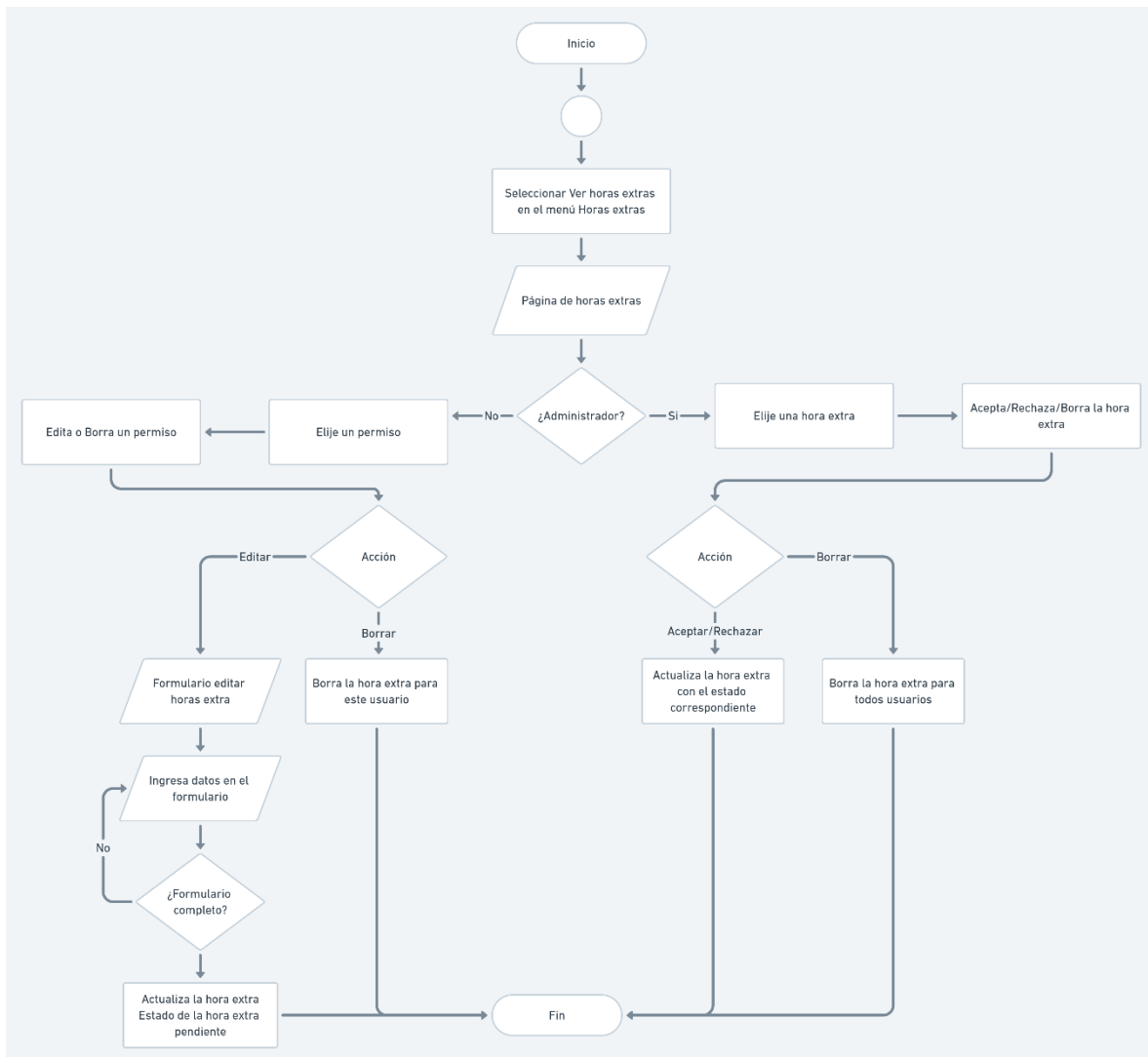
Figura 25. Proceso nueva hora extra



Fuente: Elaboración propia.

El empleado deberá completar un formulario para registrar una hora extra, mientras que el administrador será redireccionado a la página para revisar las solicitudes de horas extras.

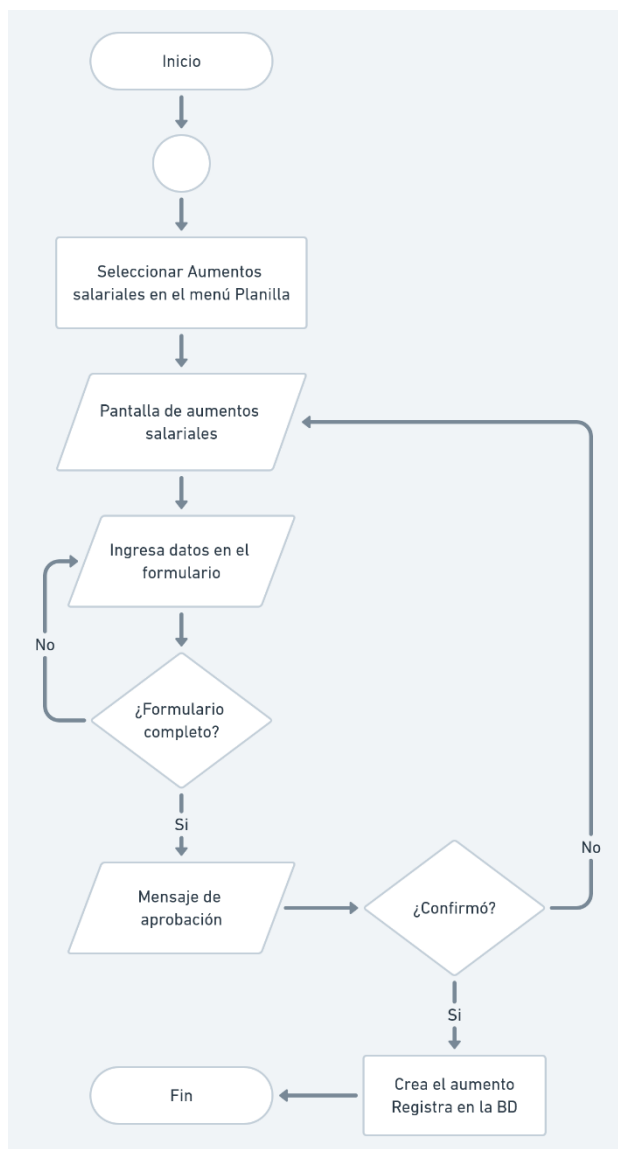
Figura 26. Proceso ver horas extras



Fuente: Elaboración propia.

Los usuarios administradores elijen las horas extras que desean aceptar, rechazar o borrar; en caso de desee borrar una solicitud, esta se borrará tanto para él como para el empleado que la solicitó. En el caso de los empleados, si editan una solicitud realizada, ya sea aceptada o rechazada, ésta pasará automáticamente a pendiente, si la borra sólo será borrada para sí mismo.

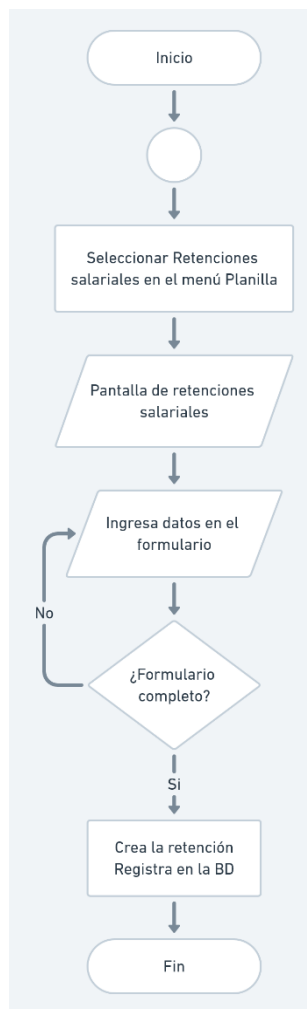
Figura 27. Proceso aumento salarial



Fuente: Elaboración propia.

El administrador deberá completar el formulario y confirmar el aumento para registrar este en la base de datos.

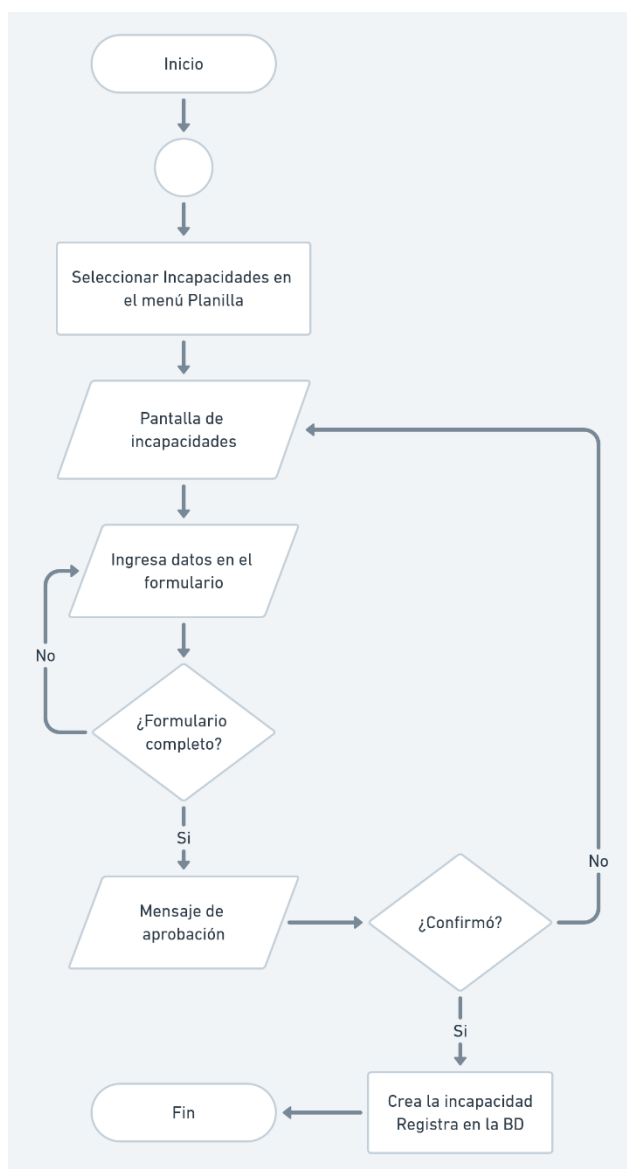
Figura 28. Proceso retención salarial



Fuente: Elaboración propia.

El administrador deberá ingresar a la página de retenciones y completar el formulario para registrar la nueva retención salarial.

Figura 29. Proceso incapacidad



Fuente: Elaboración propia.

El administrador deberá completar el formulario y confirmar la incapacidad para registrar dicha incapacidad en la base de datos.

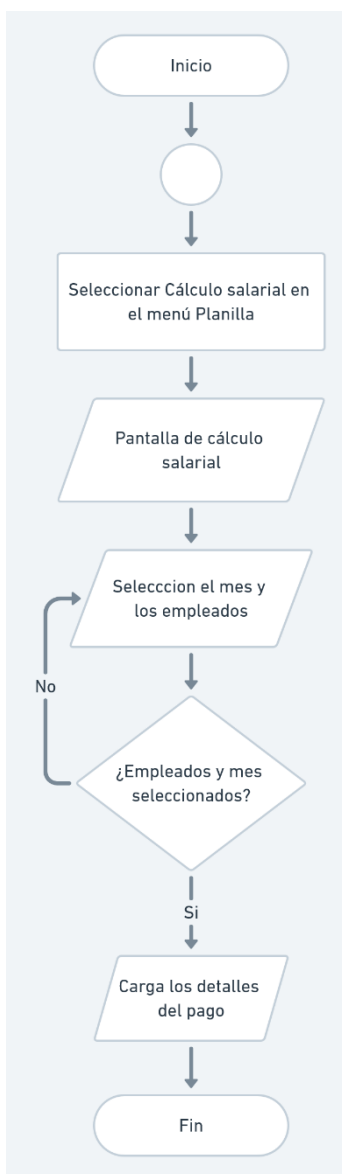
Figura 30. Proceso viáticos



Fuente: Elaboración propia.

El administrador deberá ingresar a la página de viáticos y completar el formulario para poder registrarlos en la base de datos.

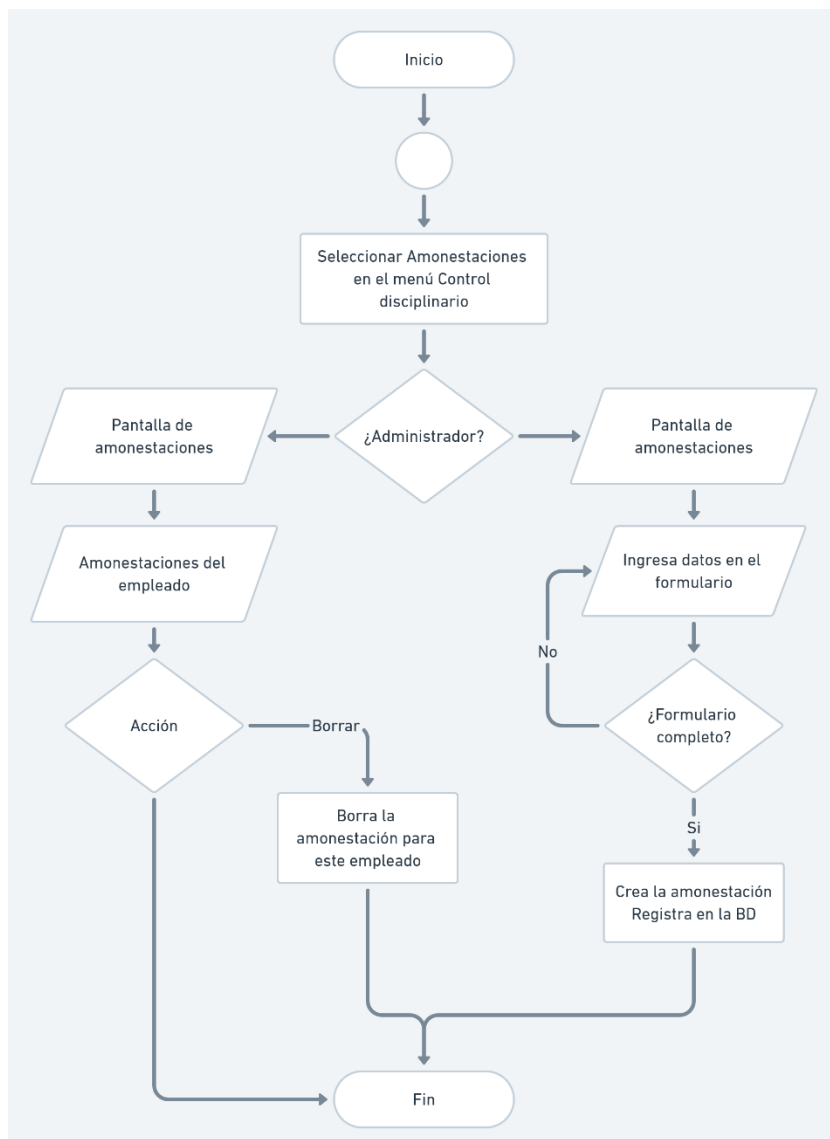
Figura 31. Proceso calculo salarial



Fuente: Elaboración propia.

El administrador deberá ingresar a la página de cálculo salarial, seleccionar un mes, año y los empleados para poder consultar el salario detallado de los empleados seleccionados.

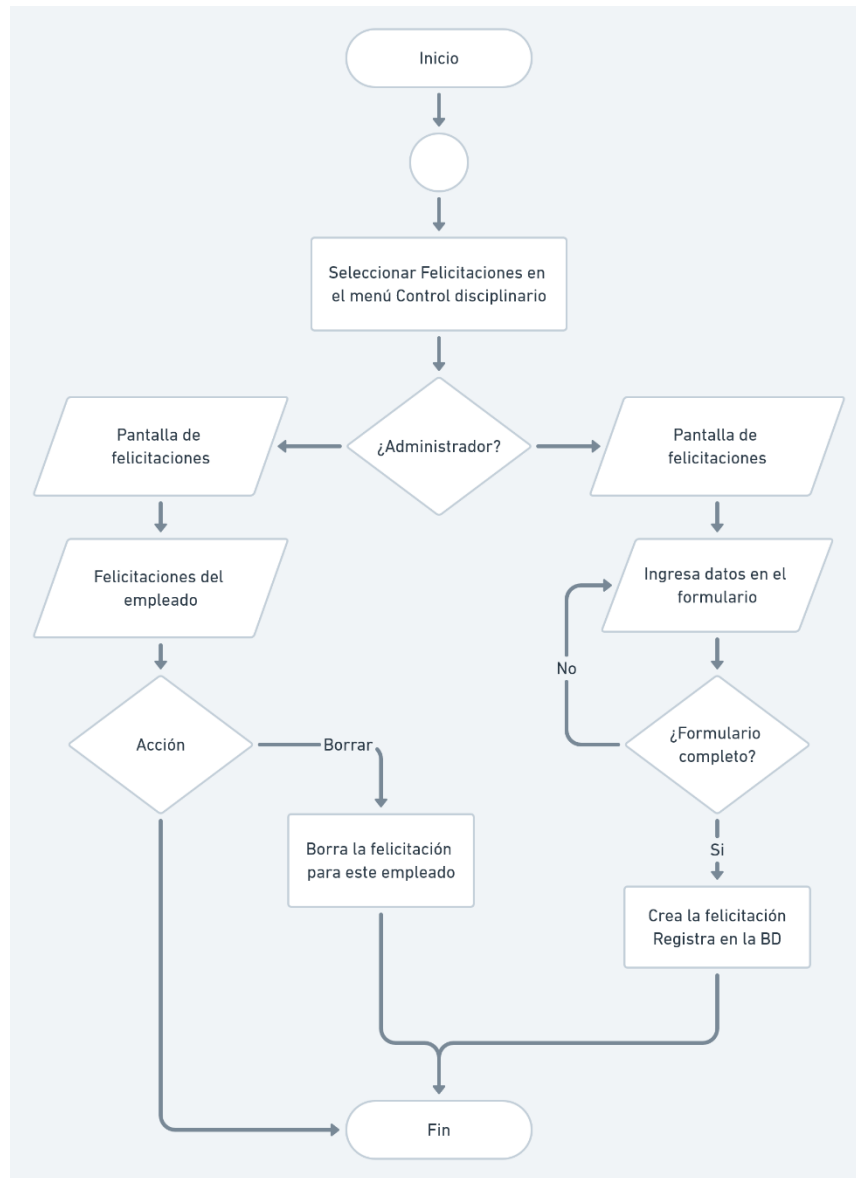
Figura 32. Proceso amonestaciones



Fuente: Elaboración propia.

A los empleados que ingresen a esta página se les mostrará las amonestaciones que le han sido asignadas; mientras tanto, a los administradores se les mostrará un formulario que deben completar en caso de registrar una nueva amonestación.

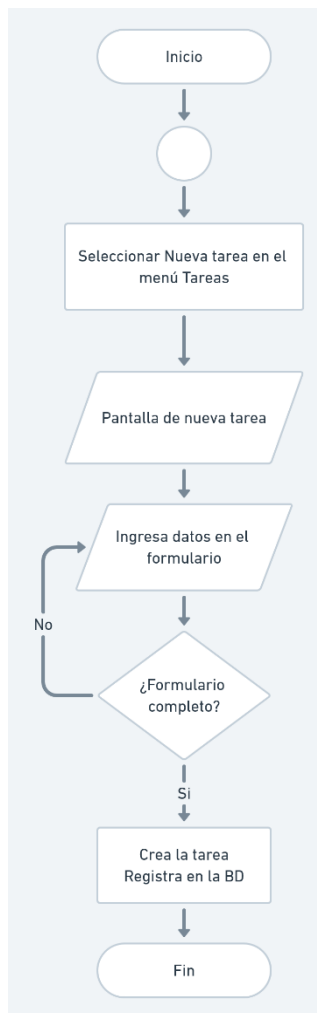
Figura 33. Proceso felicitaciones



Fuente: Elaboración propia.

A los empleados que ingresen a esta página se les mostrará las felicitaciones que le han sido asignadas. Mientras tanto, a los administradores se les mostrará un formulario que deben completar en caso de registrar una nueva felicitación.

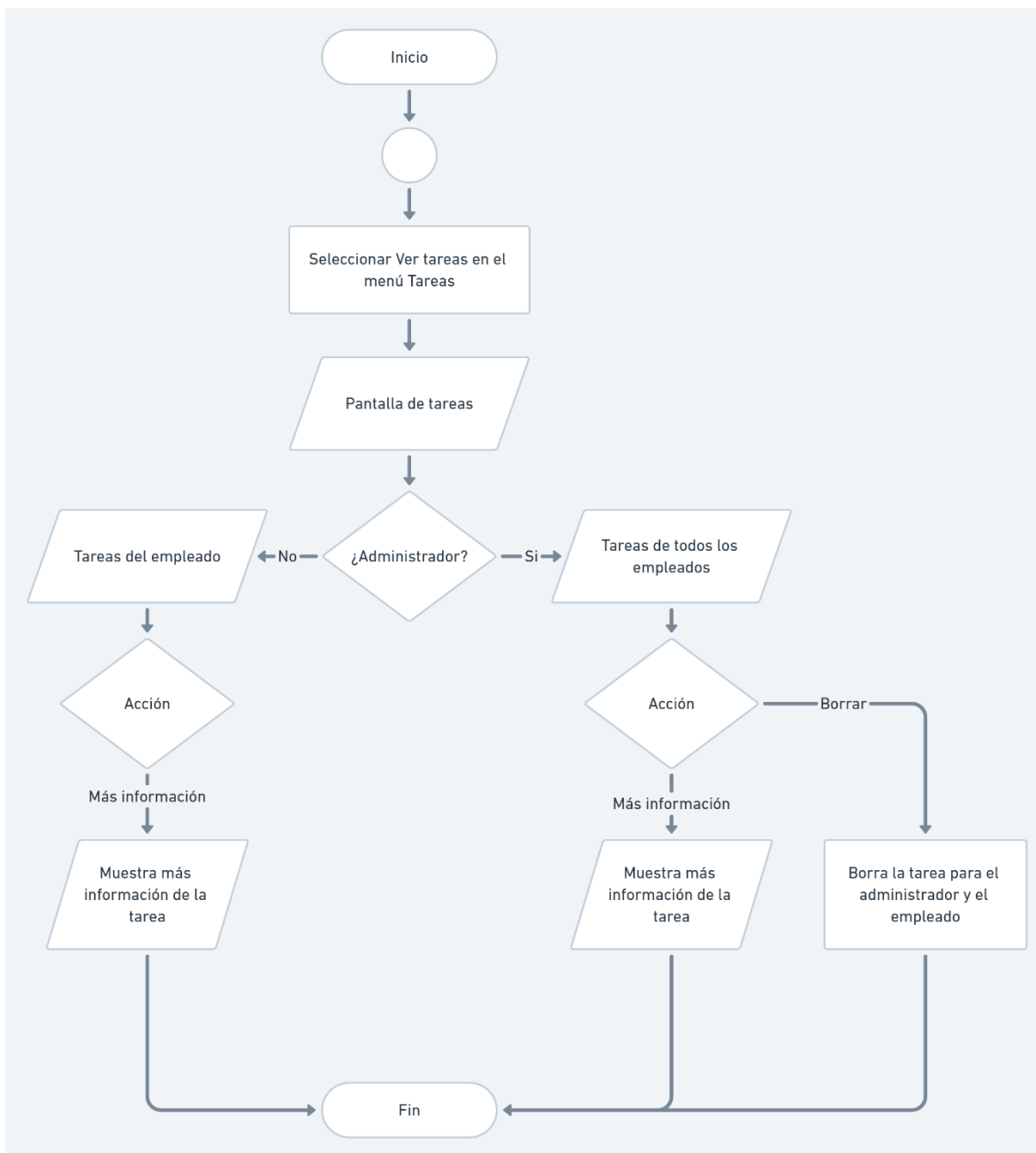
Figura 34. Proceso nueva tarea



Fuente: Elaboración propia.

El administrador deberá ingresar a la página de tareas y completar el formulario para poder registrar una nueva en la base de datos.

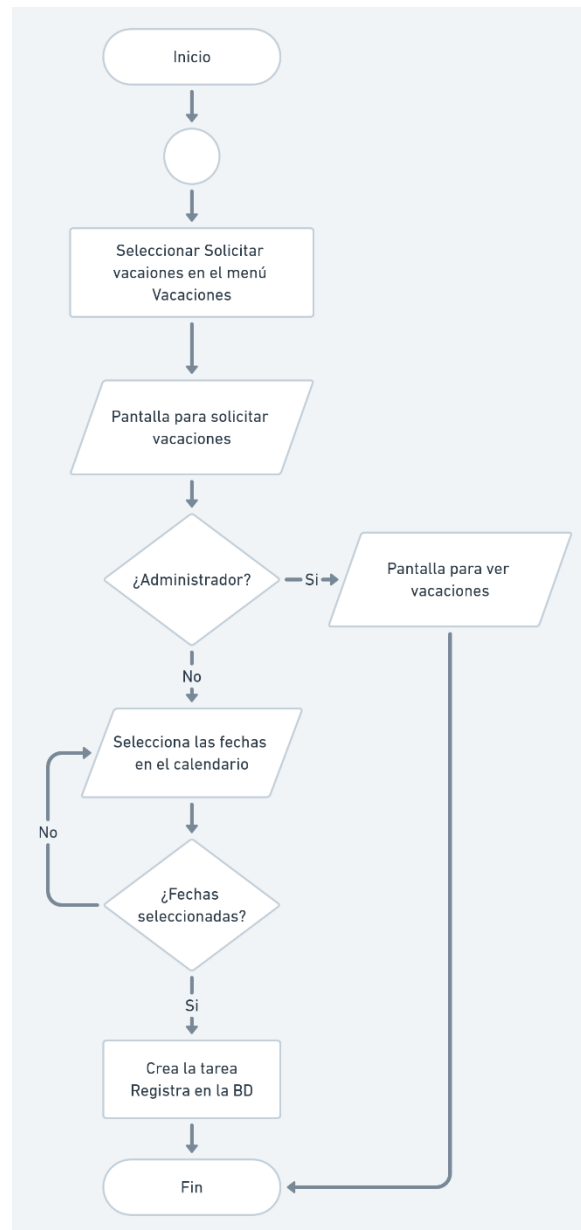
Figura 35. Proceso ver tareas



Fuente: Elaboración propia.

En caso de ser usuario administrador, se cargan todas las tareas de los empleados, si decide borrar esta tarea, se borrará tanto para él como para el empleado. Si el usuario es un empleado se cargarán las tareas que le han sido asignadas y se elige mostrar más información para la tarea se carga una ventana con la información.

Figura 36. Proceso solicitar vacaciones



Fuente: Elaboración propia.

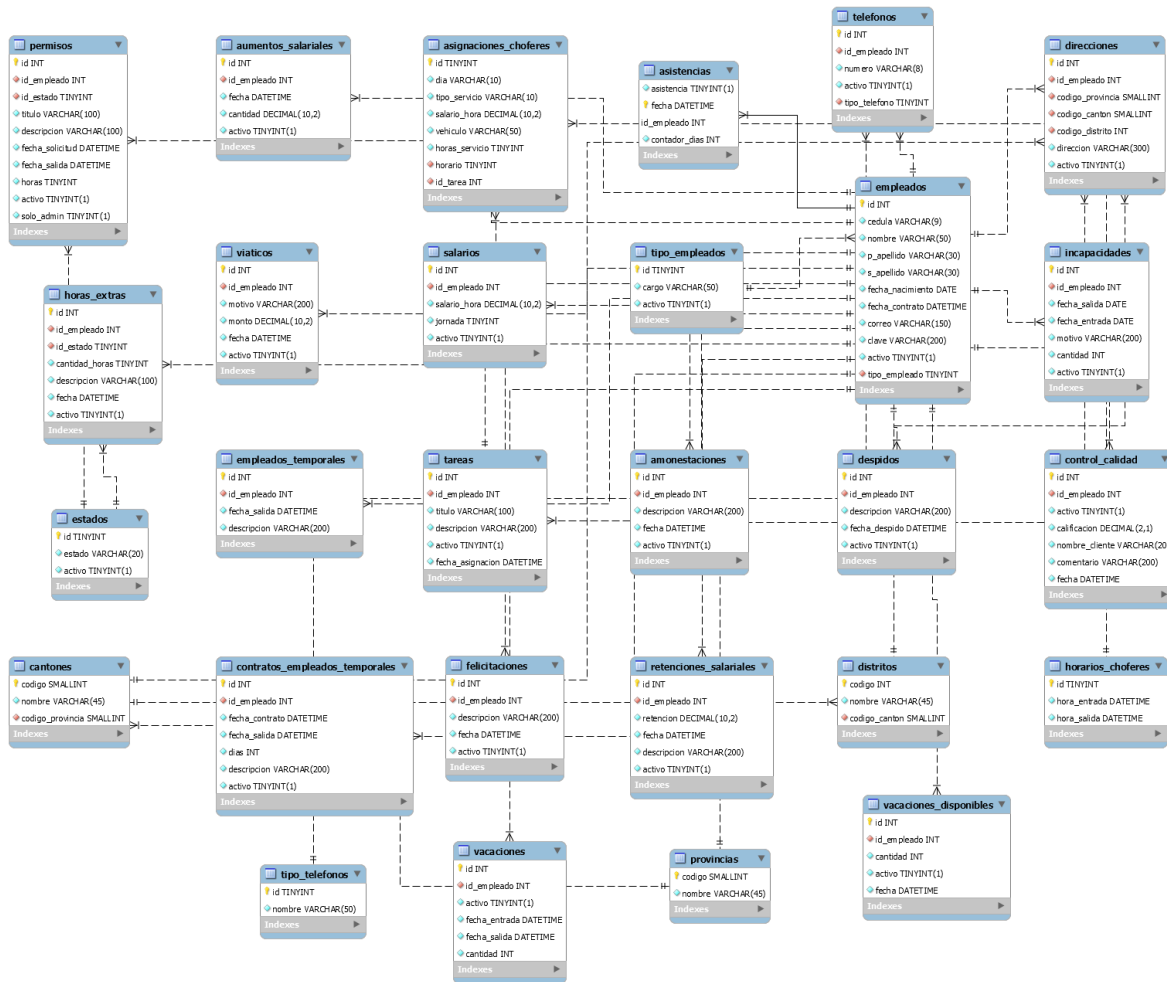
En caso de ser administrador, este será redireccionado a la página para ver las vacaciones solicitadas, en cuanto a los empleados, estos deberán completar el formulario para solicitar vacaciones y ser registradas en la base de dato.



## Diseño de base de datos

El siguiente diseño presenta el modelo de la base de datos utilizado para el desarrollo de este prototipo, este muestra las diferentes tablas y columnas.

Figura 38. Diagrama de base de datos



Fuente: Elaboración propia.

## Diccionario

Relacionado con el diseño de la base de datos, las siguientes tablas muestran el tipo de dato que tiene cada una de las columnas que residen en las tablas.

Tabla 38. Diccionario para asignaciones choferes

Asignaciones de choferes										
Column name	Data Type	P K	N N	U Q	BI N	U N	Z F	A I	Defa ult	Comm ent
id	TINYINT(4)	✓	✓					✓		
dia	VARCHAR(10)		✓							
tipo_servicio	VARCHAR(10)		✓							
salario_hora	DECIMAL(10,2)		✓							
vehiculo	VARCHAR(50)		✓							
horas_servicio	TINYINT(4)		✓							
horario	TINYINT(4)		✓							
id_tarea	INT(11)		✓							

Fuente: Elaboración propia.

Tabla 39. Diccionario para asistencias

asistencias										
Column name	Data Type	P K	N N	U Q	BI N	U N	Z F	A I	Default	Comment
asistencia	TINYINT(1)		✓						'0'	
fecha	DATETIME	✓	✓						CURRENT_TIMESTAMP	
id_empleado	INT(11)	✓	✓							
contador_dias	INT(11)		✓						'0'	

Fuente: Elaboración propia.

Tabla 40. Diccionario para aumentos salariales

Aumentos_Salariales										
Column name	DataType	P K	N N	U Q	BI N	U N	Z F	A I	Default	Comment
<b>id</b>	INT(11)	✓	✓					✓		
<b>id_emp leado</b>	INT(11)		✓							
<b>fecha</b>	DATETIME		✓						CURRENT_TIMESTAMP	
<b>cantida d</b>	DECIMAL(10,2)		✓							
<b>activo</b>	TINYINT(1)		✓						'1'	

Fuente: Elaboración propia.

Tabla 41. Diccionario para cantones

Cantones										
Column name	DataType	P K	N N	U Q	BI N	U N	Z F	A I	Default	Comment
<b>codigo</b>	SMALLINT(5)	✓	✓							
<b>nombre</b>	VARCHAR(45)		✓							
<b>codigo_prov incia</b>	SMALLINT(5)		✓							

Fuente: Elaboración propia.

Tabla 42. Diccionario para contratos empleados temporales

Contratos_Empleados_Temporales										
Column name	DataType	P K	N N	U Q	BI N	U N	Z F	A I	Default	Comment
<b>id</b>	INT(11)	✓	✓					✓		

<b>id_employ</b>	INT(11)		✓							
<b>fecha_contrato</b>	DATETIME		✓							
<b>fecha_salida</b>	DATETIME		✓							
<b>dias</b>	INT(11)		✓						'0'	
<b>descripcion</b>	VARCHAR(300)		✓							
<b>activo</b>	TINYINT(1)		✓						'1'	

Fuente: Elaboración propia.

Tabla 43. Diccionario para control calidad

<b>Control_de Calidad</b>										
Column name	Data Type	P K	N N	U Q	B I N	U N	Z F	A I	Default	Comment
<b>id</b>	INT(11)	✓	✓					✓		
<b>id_employ</b>	INT(11)		✓							
<b>activo</b>	TINYINT(1)		✓						'0'	
<b>calificación</b>	DECIMAL(2,1)		✓							
<b>nombre_cliente</b>	VARCHAR(20)		✓							
<b>comentario</b>	VARCHAR(200)		✓							
<b>fecha</b>	DATETIME		✓						CURRENT_TIMESTAMP	

Fuente: Elaboración propia.

Tabla 44. Diccionario para despidos

<b>Despidos</b>										
Column name	Data Type	P K	N N	U Q	B I N	U N	Z F	A I	Default	Comment

<b>id</b>	INT(11)	✓	✓					✓		
<b>id_empleado</b>	INT(11)		✓							
<b>descripcion</b>	VARCHAR(300)		✓							
<b>fecha_despido</b>	DATE		✓						CURRENT_TIMESTAMP	
<b>activo</b>	TINYINT(1)		✓						'1'	

Fuente: Elaboración propia.

Tabla 45. Diccionario para direcciones

Direcciones										
Column name	DataType	P K	N N	U Q	BI N	U N	Z F	A I	Defa ult	Comm ent
<b>id</b>	INT(11)	✓	✓					✓		
<b>id_empleado</b>	INT(11)		✓							
<b>codigo_provincia</b>	SMALLINT(6)		✓						'1'	
<b>codigo_canton</b>	SMALLINT(6)		✓						'114'	
<b>codigo_distrito</b>	INT(11)		✓						'11401'	
<b>direccion</b>	VARCHAR(300)		✓						''	
<b>activo</b>	TINYINT(1)		✓						'1'	

Fuente: Elaboración propia.

Tabla 46. Diccionario para distritos

Distritos										
Column name	DataType	P K	N N	U Q	BI N	U N	Z F	A I	Defa ult	Comm ent
<b>codigo</b>	INT(10)	✓	✓							

<b>nombre</b>	VARCHAR (45)		✓							
<b>codigo_canton</b>	SMALLINT (5)		✓							

Fuente: Elaboración propia.

Tabla 47. Diccionario para empleados

Empleados										
Column name	Data Type	P K	N N	U Q	B I N	U N	Z F	A I	Default	Comment
<b>id</b>	INT(11)	✓	✓					✓		
<b>cedula</b>	VARCHAR(9)		✓							
<b>nombre</b>	VARCHAR(50)		✓							
<b>p_apellido</b>	VARCHAR(30)		✓							
<b>s_apellido</b>	VARCHAR(30)		✓							
<b>fecha_nacimiento</b>	DATE		✓							
<b>fecha_contrato</b>	DATETIME		✓						CURRENT_TIMESTAMP	
<b>correo</b>	VARCHAR(200)		✓							
<b>clave</b>	VARCHAR(200)		✓							
<b>activo</b>	TINYINT(1)		✓							
<b>tipo_empleado</b>	TINYINT(4)		✓							

Fuente: Elaboración propia.

Tabla 48. Diccionario para empleados temporales

<b>Empleados_Temporales</b>										
Column name	DataType	P K	N N	U Q	BI N	U N	Z F	A I	Defau lt	Comme nt
<b>id</b>	INT(11)	✓	✓					✓		
<b>id_emple ado</b>	INT(11)		✓							
<b>fecha_sal ida</b>	DATETIME		✓							
<b>descripci on</b>	VARCHAR( 300)		✓							

Fuente: Elaboración propia.

Tabla 49. Diccionario para estados

<b>Estados</b>										
Colu mn name	DataType	P K	N N	U Q	BI N	U N	Z F	AI	Defau lt	Comme nt
<b>id</b>	TINYINT(4)	✓	✓					✓		
<b>estad o</b>	VARCHAR( 20)		✓							
<b>activo</b>	TINYINT(1)		✓						'1'	

Fuente: Elaboración propia.

Tabla 50. Diccionario para horarios choferes

<b>Horario de _Choferes</b>										
Column name	DataTyp e	P K	N N	U Q	BI N	U N	Z F	A I	Defau lt	Comme nt
<b>id</b>	TINYINT (4)	✓	✓					✓		
<b>hora_entr ada</b>	DATETI ME		✓							

<b>hora_salida</b>	DATETIME		✓							
--------------------	----------	--	---	--	--	--	--	--	--	--

Fuente: Elaboración propia.

Tabla 51. Diccionario para horas extras

<b>Horas_Extras</b>										
Column name	Data Type	P K	N N	U Q	B I N	U N	Z F	A I	Default	Comment
<b>id</b>	INT(11)	✓	✓					✓		
<b>id_employado</b>	INT(11)		✓							
<b>id_estado</b>	TINYINT(4)		✓						'1'	
<b>cantidad_horas</b>	TINYINT(4)		✓						'0'	
<b>descripcion</b>	VARCHAR(300)		✓							
<b>fecha</b>	DATETIME		✓						CURRENT_TIMESTAMP	
<b>activo</b>	TINYINT(1)		✓						'1'	

Fuente: Elaboración propia.

Tabla 52. Diccionario para incapacidades

<b>Incapacidades</b>										
Column name	Data Type	P K	N N	U Q	B I N	U N	Z F	A I	Default	Comment
<b>id</b>	INT(11)	✓	✓					✓		
<b>id_employado</b>	INT(11)		✓							
<b>fecha_salida</b>	DATE		✓							
<b>fecha_entrada</b>	DATE		✓							

<b>motivo</b>	VARCHAR(200)		✓							
<b>cantidad</b>	INT(11)		✓						'0'	
<b>activo</b>	TINYINT(1)		✓						'1'	

Fuente: Elaboración propia.

Tabla 53. Diccionario para permisos

Permisos										
Column name	Data Type	P K	N N	U Q	BI N	U N	Z F	A I	Default	Comment
<b>id</b>	INT(11)	✓	✓					✓		
<b>id_empleado</b>	INT(11)		✓							
<b>id_estado</b>	TINYINT(4)		✓						'1'	
<b>titulo</b>	VARCHAR(100)		✓							
<b>descripcion</b>	VARCHAR(100)		✓							
<b>fecha_solicitud</b>	DATETIME		✓						CURRENT_TIMESTAMP	
<b>fecha_salida</b>	DATETIME		✓							
<b>horas</b>	TINYINT(4)		✓							
<b>activo</b>	TINYINT(1)		✓						'1'	
<b>solo_admin</b>	TINYINT(1)		✓						'0'	

Fuente: Elaboración propia.

Tabla 54. Diccionario para provincias

Provincias										
Column name	Data Type	P K	N N	U Q	BI N	U N	Z F	AI	Default	Comment

<b>codigo</b>	SMALLINT(5)	✓	✓					✓		
<b>nombre</b>	VARCHAR(45)		✓							

Fuente: Elaboración propia.

Tabla 55. Diccionario para retenciones salariales

<b>Retención_Salarial</b>										
Column name	DataType	P K	N N	U Q	BI N	U N	Z F	A I	Default	Comment
<b>id</b>	INT(11)	✓	✓					✓		
<b>id_employado</b>	INT(11)		✓							
<b>retencion</b>	DECIMAL(10,2)		✓						'0.00'	
<b>fecha</b>	DATETIME		✓						CURRENT_TIMESTAMP	
<b>descripcion</b>	VARCHAR(300)		✓							
<b>activo</b>	TINYINT(1)		✓						'1'	

Fuente: Elaboración propia.

Tabla 56. Diccionario para salarios

<b>Salarios</b>										
Column name	DataType	P K	N N	U Q	BI N	U N	Z F	A I	Default	Comment
<b>id</b>	INT(11)	✓	✓					✓		
<b>id_employado</b>	INT(11)		✓							
<b>salario_hora</b>	DECIMAL(10,2)		✓							
<b>jornada</b>	TINYINT(4)		✓							
<b>activo</b>	TINYINT(1)		✓						'1'	

Fuente: Elaboración propia.

Tabla 57. Diccionario para tareas

Tareas										
Column name	Data Type	P K	N N	U Q	B I N	U N	Z F	A I	Default	Comment
<b>id</b>	INT(11)	✓	✓					✓		
<b>id_employado</b>	INT(11)		✓							
<b>titulo</b>	VARCHAR(100)		✓							
<b>descripcion</b>	VARCHAR(300)		✓							
<b>activo</b>	TINYINT(1)		✓						'1'	
<b>fecha_asignacion</b>	DATE TIME		✓						CURRENT_TIMESTAMP	

Fuente: Elaboración propia.

Tabla 58. Diccionario para teléfonos

Teléfonos										
Column name	Data Type	P K	N N	U Q	B I N	U N	Z F	A I	Default	Comment
<b>id</b>	INT(11)	✓	✓					✓		
<b>id_employado</b>	INT(11)		✓							
<b>numero</b>	VARCHAR(8)		✓							
<b>activo</b>	TINYINT(1)		✓						'0'	
<b>tipo_telefono</b>	TINYINT(4)		✓						'1'	

Fuente: Elaboración propia.

Tabla 59. Diccionario para tipo empleados

Tipo de_Empleados										
Column name	Data Type	P K	N N	U Q	BI N	U N	Z F	AI	Default	Comment
<b>id</b>	TINYINT(4)	✓	✓					✓		
<b>cargo</b>	VARCHAR(50)		✓							
<b>activo</b>	TINYINT(1)		✓						'1'	

Fuente: Elaboración propia.

Tabla 60. Diccionario para tipo teléfonos

Tipo de_Teléfono										
Column name	Data Type	P K	N N	U Q	BI N	U N	Z F	AI	Default	Comment
<b>id</b>	TINYINT(4)	✓	✓					✓		
<b>nombre</b>	VARCHAR(50)		✓							

Fuente: Elaboración propia.

Tabla 61. Diccionario para vacaciones

Vacaciones										
Column name	Data Type	P K	N N	U Q	BI N	U N	Z F	AI	Default	Comment
<b>id</b>	INT(11)	✓	✓					✓		
<b>id_empleado</b>	INT(11)		✓							
<b>activo</b>	TINYINT(1)		✓						'1'	
<b>fecha_entrada</b>	DATETIME		✓							
<b>fecha_salida</b>	DATETIME		✓							
<b>cantidad</b>	INT(11)		✓							

Fuente: Elaboración propia.

Tabla 62. Diccionario para vacaciones disponibles

<b>Vacaciones_Disponibles</b>										
<b>Column name</b>	<b>Data Type</b>	<b>P K</b>	<b>N N</b>	<b>U Q</b>	<b>BI N</b>	<b>U N</b>	<b>Z F</b>	<b>A I</b>	<b>Default</b>	<b>Comment</b>
<b>id</b>	INT(11)	✓	✓					✓		
<b>id_empleado</b>	INT(11)		✓							
<b>cantidad</b>	INT(11)		✓						'0'	
<b>activo</b>	TINYINT(1)		✓						'1'	
<b>fecha</b>	DATETIME		✓						CURRENT_TIMESTAMP	

Fuente: Elaboración propia.

## Programación

En esta sección se muestran fragmentos del código fuente del sistema en relación a las diferentes entradas, salidas, procesos, validaciones y módulos del mismo. Cada uno de estos módulos fue desarrollado en base a los diseños presentados anteriormente.

### Entradas y salidas

Figura 39. Entrada vacaciones

```

<ValidationObserver ref="observer" v-slot="{ invalid, reset }" tag="div" class="columns is-multiline">
  <div class="column is-half">
    <div class="columns is-multiline">
      <div class="column is-full">
        <nav class="level">
          <div class="level-item has-text-centered">
            <div>
              <p class="heading">Días disponibles</p>
              <p class="title">{{ availableDays.cantidad || 0 }}</p>
            </div>
          </div>
        </nav>
      </div>

      <div class="column is-full">
        <nav class="level">
          <div class="level-item has-text-centered">
            <div>
              <p class="heading">Días solicitados</p>
              <p class="title">{{ requestedDays }}</p>
            </div>
          </div>
        </nav>
      </div>

      <div class="column is-full">
        <nav class="level">
          <div class="level-item has-text-centered">
            <div>
              <p class="heading">Días disponibles después de la solicitud</p>
              <p class="title">{{ newAvailableDays }}</p>
            </div>
          </div>
        </nav>
      </div>
    </div>
  </div>

  <ValidationProvider rules="required" v-slot="{ errors }" tag="div" class="column is-half">
    <b-field :message="errors" :type="{ 'is-danger': errors[0] }">
      <b-datepicker v-model="dates" :day-names="dayNames" :month-names="monthNames" range inline />
    </b-field>
  </ValidationProvider>

  <div class="column is-full">
    <b-button type="is-primary" @click="solicitar(); reset();" :disabled="invalid" expanded>Solicitar vacaciones</b-button>
  </div>
</ValidationObserver>

```

Fuente: Elaboración propia.

En esta figura se muestra el código preciso para el formulario que recibe los datos necesarios para registrar una nueva solicitud de vacaciones.

Figura 40. Entrada permiso

```

<ValidationObserver ref="observer" v-slot="{ invalid, reset }" tag="div" class="columns is-multiline">
  <ValidationProvider :rules="{ required: true, alpha_spaces: /^[a-zA-Z\s]*$/}" v-slot="{ errors, valid }" tag="div" class="column is-full">
    <b-field label="Título" :message="errors" :type="{ 'is-danger': errors[0], 'is-success': valid }" expanded>
      <b-input v-model="title"></b-input>
    </b-field>
  </ValidationProvider>

  <ValidationProvider :rules="{ required: true, alpha_spaces: /^[a-zA-Z\s]*$/, max: 100 }" v-slot="{ errors, valid }" tag="div"
  class="column is-full">
    <b-field label="Descripción" :message="errors" :type="{ 'is-danger': errors[0], 'is-success': valid }" expanded>
      <b-input v-model="description"></b-input>
    </b-field>
  </ValidationProvider>

  <ValidationProvider rules="required" v-slot="{ errors, valid }" tag="div" class="column is-half">
    <b-field label="Para el" :message="errors" :type="{ 'is-danger': errors[0], 'is-success': valid }">
      <b-datetimepicker v-model="outDate" placeholder="Fecha y hora" icon="calendar-today" :horizontal-time-picker="true"
      :datepicker="{
        'day-names': dayNames,
        'month-names': monthNames,
        'first-day-of-week': 1,
      }"
      >
    </b-datetimepicker>
  </b-field>
</ValidationProvider>

  <ValidationProvider rules="required|numeric|max_value:5"
  v-slot="{ errors, valid }" tag="div" class="column is-half">
    <b-field label="Duración" :message="errors" :type="{ 'is-danger': errors[0], 'is-success': valid }" expanded>
      <b-input v-model="hours"></b-input>
    </b-field>
  </ValidationProvider>

  <div class="column is-full">
    <button class="button is-primary is-pulled-right" :disabled="invalid" @click=" requestPermit(); reset();">
      Solicitar
    </button>
  </div>
</ValidationObserver>

```

Fuente: Elaboración propia.

Este es el código necesario del formulario que se encarga de recibir los datos para registrar un nuevo permiso.

Figura 41. Salida salario

```

<div class="column is-full">
  <div class="box">
    <h4 class="title is-4">Detalles</h4>
    <b-message type="is-info">
      <p>El porcentaje de cargas sociales corresponde a un 10.5%.</p>
      <p>Expande la fila para ver mas informacion.</p>
    </b-message>
    <b-table :data="wages" :paginated="true" :per-page="10" :mobile-cards="true" :hoverable="true" :scrollable="true" :bordered="true"
      :loading="isLoading" detailed
    >
      <template slot-scope="props">
        <b-table-column label="Empleado" width="300">{{ props.row.empleado }}</b-table-column>
        <b-table-column label="Salario por hora" width="150" centered>{{ formatSalary(props.row.salario_hora) }}</b-table-column>
        <b-table-column label="Jornada diaria" width="150" centered>{{ props.row.jornada }}</b-table-column>
        <b-table-column label="Dias trabajados" width="150" centered>{{ props.row.dias_trabajados }}</b-table-column>
        <b-table-column label="Horas extras" width="150" centered>{{ props.row.horas_extra }}</b-table-column>
        <b-table-column label="Salario bruto" centered>{{ formatSalary(props.row.salario_bruto) }}</b-table-column>
        <b-table-column label="Salario neto" centered>{{ formatSalary(props.row.salario_net) }}</b-table-column>
      </template>

      <template slot="detail" slot-scope="props">
        <b-field grouped group-multiline>
          <div class="control">
            <b-taglist attached>
              <b-tag type="is-dark">Retenciones</b-tag>
              <b-tag type="is-info">{{ props.row.retenciones }}</b-tag>
            </b-taglist>
          </div>
          <div class="control">
            <b-taglist attached>
              <b-tag type="is-dark">Horas descontadas por permisos</b-tag>
              <b-tag type="is-info">{{ props.row.permisos }}</b-tag>
            </b-taglist>
          </div>
          <div class="control">
            <b-taglist attached>
              <b-tag type="is-dark">Impuesto de renta</b-tag>
              <b-tag type="is-info">{{ formatSalary(props.row.impuesto_renta * -1) }}</b-tag>
            </b-taglist>
          </div>
          <div class="control">
            <b-taglist attached>
              <b-tag type="is-dark">C.C.S.S</b-tag>
              <b-tag type="is-info">{{ formatSalary(props.row.total_deducion * -1) }}</b-tag>
            </b-taglist>
          </div>
          <div class="control">
            <b-taglist attached>
              <b-tag type="is-dark">Aguinaldo</b-tag>
              <b-tag type="is-info">{{ formatSalary(props.row.aguinaldo) }}</b-tag>
            </b-taglist>
          </div>
        </b-field>
      </template>

      <template slot="empty">
        <section class="section">
          <div class="content has-text-grey has-text-centered">
            <p><b-icon icon="emoticon-sad" size="is-large"></b-icon></p>
            <p>No se ha calculado ningun salario.</p>
          </div>
        </section>
      </template>
    </b-table>
  </div>
</div>

```

Fuente: Elaboración propia.

En este esquema de código, el componente de la tabla recibe una lista con todos los salarios calculados, el componente se encarga internamente de recorrer la lista y construir la tabla que muestra toda la información correspondiente, únicamente se debe indicar el orden de las columnas.

Figura 42. Salida tarea

```

<template>
  <div class="card">
    <div class="card-content">
      <h5 class="is-size-5 has-text-weight-semibold mb-2">{{ task.titulo }}</h5>
      <h6 class="is-size-6 has-text-weight-semibold">Descripción</h6>
      <p class="mb-4">{{ task.descripcion }}</p>
      <b-taglist class="mb-0 important" attached>
        <b-tag type="is-dark">Asignada a</b-tag>
        <b-tag type="is-info">{{ task.asignada_a }}</b-tag>
      </b-taglist>
      <p class="has-text-grey leading-none">Hace {{ createdAgo(task.fecha_asignacion) }}</p>
    </div>
    <footer class="card-footer">
      <a class="card-footer-item" @click="openTaskModal(task.id)">Mas información</a>
      <a class="card-footer-item" v-if="userType === 1" @click="removeTask(task.id)">Eliminar tarea</a>
    </footer>
  </div>
</template>

```

Fuente: Elaboración propia.

Este código es el necesario para mostrar las tareas de los empleados, al ser un componente, puede ser reutilizado para ser ejecutado gran cantidad de veces. Cuando se consulta las tareas de los empleados se recibe una lista la cual es recorrida por un bucle, cada tarea es pasada a este componente el cual se encarga de mostrar la información correspondiente.

## Procesos

Figura 43. Proceso de configuración del servidor

```
//Archivo app.module.ts
class AppModule {
  app: Express;

  constructor() {
    this.app = express();
    this.commonMiddleware();
    this.modules();
  }

  private commonMiddleware() {
    this.app.use(morgan('dev'));
    this.app.use(express.json());
    this.app.use(express.urlencoded({ extended: true }));
    this.app.use(helmet());
    this.app.use(compression());
    this.app.use(cors());
    this.app.use(cookieParser());
  }

  private modules() {
    this.app.use('/v1/login', new LoginModule().router);
    this.app.use('/v1/users', Auth.user, new UsersModule().router);
    this.app.use('/v1/permissions', Auth.user, new PermissionsModule().router);
    this.app.use('/v1/overtime', Auth.user, new OvertimeModule().router);
    this.app.use('/v1/vacations', Auth.user, new VacationsModule().router);
    this.app.use('/v1/quality', new QualityModule().router);
    this.app.use('/v1/profile', Auth.user, new ProfilesModule().router);
    this.app.use('/v1/reports', Auth.user, new ReportsModule().router);
    this.app.use('/v1/tasks', Auth.user, new TasksModule().router);
    this.app.use('/v1/salary', Auth.user, new WageModule().router);
  }
}

export default new AppModule();

//Archivo server.ts
import http from 'http';
import AppModule from './app.module';

const port = process.env.PORT || 3000;

const server = http.createServer(AppModule.app);
server.listen(port, () => {
  console.log(`Server is running on port: ${port}`);
});
```

Fuente: Elaboración propia.

El código mostrado en la imagen muestra cómo configurar un servidor con Node.js utilizando la biblioteca Express y el lenguaje de programación TypeScript.

## Validaciones

Figura 44. Validación de la sesión

```
public user = (req: Request, res: Response, next: NextFunction) => {
  const bearer = req.headers['authorization'];

  if (!bearer)
    return res.status(401).send({
      status: 401,
      type: 'UnauthorizedException',
      message: 'Tienes que haber iniciado sesión para realizar esta acción',
    });

  const token = bearer.split(' ')[1];

  jwt.verify(token, process.env.JWT_KEY || 'secret', (error, decoded) => {
    if (error)
      return res.status(401).send({
        status: 401,
        type: 'UnauthorizedException',
        message: 'Tienes que haber iniciado sesión para realizar esta acción',
      });
    res.locals.authenticated = decoded;
    next();
  });
};
```

Fuente: Elaboración propia.

El fragmento de código mostrado en la imagen, valida si el usuario cuenta con una sesión activa en el sistema, en caso de que no, el sistema retorna un código de error y mensaje indicando que debe iniciar sesión para hacer uso del sistema. Además, se valida si la sesión activa ya expiró, si es el caso, se retorna un código de error y mensaje como en la validación anterior. Estas validaciones se enfocan en validar los tokens encontrados en las cabeceras de cada petición HTTP; además, de verificar la autenticidad de los mismos.

## Módulos

Figura 45. Módulo tareas

```

public createTaskDriver = async (req: Request, res: Response) => {
  const { userId } = req.params;
  const { titulo, descripcion, asignacion_chofer } = req.body;

  if (!asignacion_chofer) {
    return res.status(400).send({ message: 'Faltan datos para asignar esta tarea' });
  }

  try {
    const userInfo = await this.userBusiness.getEmployee(parseInt(userId));
    if (userInfo.tipo_empleado !== Rol.Chofer) {
      return res.status(400).send({
        message: 'Este tipo de empleado no puede recibir este tipo de tarea',
      });
    }

    let driverTask: DriverTask;

    if (asignacion_chofer.tipo_servicio === 'Especial') {
      driverTask = {
        vehiculo: asignacion_chofer.vehiculo,
        horario: 4,
        tipo_servicio: asignacion_chofer.tipo_servicio,
        horas_servicio: asignacion_chofer.horas_servicio,
        salario_hora: asignacion_chofer.salario_hora,
      };
    } else {
      driverTask = {
        vehiculo: asignacion_chofer.vehiculo,
        horario: asignacion_chofer.horario,
        tipo_servicio: asignacion_chofer.tipo_servicio,
        horas_servicio: userInfo.jornada,
        salario_hora: userInfo.salario_hora,
      };
    }

    await this.taskBusiness.insertTaskDriver(
      { id_empleado: parseInt(userId), titulo, descripcion },
      driverTask,
      asignacion_chofer.dias
    );

    return res.status(200).send({ message: 'Tarea asignada' });
  } catch (error) {
    return res.status(400).send(error);
  }
};

```

Fuente: Elaboración propia.

La figura anterior muestra un esquema de código del módulo de tareas en el cual se confirma que existan los datos necesarios; además, de corroborar que al empleado al que se le esta asignando dicha tarea no posee un rol de administrador. En específico fragmento agrega una tarea de servicio de transporte para un empleado con rol de chofer.

## Pruebas

Una vez finalizado el proceso en el cual se desarrolla el prototipo funcional, se procede con la etapa final en cual consiste en la realización de pruebas cuyo principal objetivo es la disminución de aquellos posibles errores que pueden afectar el correcto funcionamiento del prototipo. A continuación, se muestran en detalle todas las pruebas realizadas al sistema; además, de los resultados que se esperaban y los resultados obtenidos.

Tabla 63. Caso de prueba 01

<b>Caso de prueba</b>	Inicio de sesión
<b>Código del caso de uso</b>	CP - 01
<b>Objetivo</b>	Comprobar el funcionamiento del inicio de sesión.
<b>Resultado esperado</b>	Observar un mensaje de error en caso de que las credenciales ingresadas por el usuario no estén registradas.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar un mensaje indicando que las credenciales son incorrectas.

Fuente: Elaboración propia.

Tabla 64. Caso de prueba 02

<b>Caso de prueba</b>	Ver empleados
<b>Código del caso de uso</b>	CP - 02
<b>Objetivo</b>	Ver todos los empleados que se encuentran laborando en la empresa.
<b>Resultado esperado</b>	Una lista de todos los empleados que se encuentran laborando.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar el listado de todos los empleados.

Fuente: Elaboración propia.

Tabla 65. Caso de prueba 03

<b>Caso de prueba</b>	Ver información de un empleado
<b>Código del caso de uso</b>	CP - 03
<b>Objetivo</b>	Ver toda la información relacionada al empleado.
<b>Resultado esperado</b>	Toda la información personal, de contacto y laboral relacionada al empleado.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar toda la información relacionada al empleado.

Fuente: Elaboración propia.

Tabla 66. Caso de prueba 04

<b>Caso de prueba</b>	Actualizar empleado, ver formulario
<b>Código del caso de uso</b>	CP - 04
<b>Objetivo</b>	Ver una ventana con un formulario y los campos necesarios para actualizar la información del empleado.
<b>Resultado esperado</b>	Ventana con el formulario y los campos requeridos para la actualización.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar la ventana con el formulario.

Fuente: Elaboración propia.

Tabla 67. Caso de prueba 05

<b>Caso de prueba</b>	Actualizar empleado
<b>Código del caso de uso</b>	CP - 05
<b>Objetivo</b>	Comprobar el funcionamiento del formulario para actualizar un empleado.
<b>Resultado esperado</b>	Observar un mensaje de error en caso de que los campos requeridos no sean válidos.

<b>Resultado obtenido</b>	Resultado exitoso al mostrar un mensaje indicando que los campos completados son inválidos.
---------------------------	---

Fuente: Elaboración propia.

Tabla 68. Caso de prueba 06

<b>Caso de prueba</b>	Despedir empleado, ver formulario
<b>Código del caso de uso</b>	CP - 06
<b>Objetivo</b>	Ver una ventana con el campo necesario para despedir al empleado.
<b>Resultado esperado</b>	Ventana con el campo requerido para el despido.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar la ventana con el formulario.

Fuente: Elaboración propia.

Tabla 69. Caso de prueba 07

<b>Caso de prueba</b>	Actualizar empleado
<b>Código del caso de uso</b>	CP - 07
<b>Objetivo</b>	Comprobar el funcionamiento del formulario para despedir a un empleado.
<b>Resultado esperado</b>	Observar un mensaje de error en caso de que el campo requerido no sea válido.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar un mensaje indicando que el campo completados es inválido.

Fuente: Elaboración propia.

Tabla 70. Caso de prueba 08

<b>Caso de prueba</b>	Registrar empleado
<b>Código del caso de uso</b>	CP - 08

<b>Objetivo</b>	Comprobar el funcionamiento del formulario para registrar a un nuevo empleado.
<b>Resultado esperado</b>	Observar una notificación de error en caso de que alguno de los campos requeridos no sea válido.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar una notificación indicando que hay campos requeridos que son inválidos.

Fuente: Elaboración propia.

Tabla 71. Caso de prueba 09

<b>Caso de prueba</b>	Solicitar nuevo permiso
<b>Código del caso de uso</b>	CP - 09
<b>Objetivo</b>	Comprobar el funcionamiento del formulario para que los empleados soliciten un permiso.
<b>Resultado esperado</b>	Observar una notificación de error en caso de que alguno de los campos requeridos no sea válido.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar una notificación indicando que hay campos requeridos que son inválidos.

Fuente: Elaboración propia.

Tabla 72. Caso de prueba 10

<b>Caso de prueba</b>	Ver permisos administradores
<b>Código del caso de uso</b>	CP - 10
<b>Objetivo</b>	Ver todos los permisos solicitados por los empleados.

<b>Resultado esperado</b>	Cargar todos los permisos que los empleados han solicitado.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar todos los permisos solicitados y categorizándolos en pendiente, aceptados y rechazados.

Fuente: Elaboración propia.

Tabla 73. Caso de prueba 11

<b>Caso de prueba</b>	Ver permisos empleados
<b>Código del caso de uso</b>	CP - 11
<b>Objetivo</b>	Ver todos los permisos que ha solicitado cierto empleado.
<b>Resultado esperado</b>	Cargar todos los permisos que el empleado ha solicitado.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar todos los permisos solicitados y categorizándolos en pendiente, aceptados y rechazados.

Fuente: Elaboración propia.

Tabla 74. Caso de prueba 12

<b>Caso de prueba</b>	Aceptar permiso
<b>Código del caso de uso</b>	CP - 12
<b>Objetivo</b>	Aceptar un permiso con estado pendiente.
<b>Resultado esperado</b>	Mostrar una notificación indicando que el permiso ha sido aceptado.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar la notificación de que el permiso ha sido aceptado.

Fuente: Elaboración propia.

Tabla 75. Caso de prueba 13

<b>Caso de prueba</b>	Rechazar permiso
-----------------------	------------------

<b>Código del caso de uso</b>	CP - 13
<b>Objetivo</b>	Rechazar un permiso con estado pendiente o aceptado.
<b>Resultado esperado</b>	Mostrar una notificación indicando que el permiso ha sido rechazado.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar la notificación de que el permiso ha sido rechazado.

Fuente: Elaboración propia.

Tabla 76. Caso de prueba 14

<b>Caso de prueba</b>	Borrar permiso administrador
<b>Código del caso de uso</b>	CP - 14
<b>Objetivo</b>	Borrar un permiso con cualquier tipo de estado.
<b>Resultado esperado</b>	Mostrar una notificación indicando que el permiso ha sido borrado y que también fue borrado para el empleado.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar la notificación de que el permiso ha sido borrado, además de no mostrarse ante el empleado.

Fuente: Elaboración propia.

Tabla 77. Caso de prueba 15

<b>Caso de prueba</b>	Editar permiso empleado
<b>Código del caso de uso</b>	CP - 15
<b>Objetivo</b>	Comprobar el funcionamiento del formulario para editar el permiso.
<b>Resultado esperado</b>	Observar una notificación mostrando un mensaje de error en caso de que algún campo del formulario sea invalido.

<b>Resultado obtenido</b>	Resultado exitoso al mostrar una notificación indicando que hay campos requeridos que son inválidos.
---------------------------	--

Fuente: Elaboración propia.

Tabla 78. Caso de prueba 16

<b>Caso de prueba</b>	Borrar permiso empleado
<b>Código del caso de uso</b>	CP - 16
<b>Objetivo</b>	Borrar un permiso con cualquier tipo estado siendo un empleado.
<b>Resultado esperado</b>	Mostrar una notificación indicando que el permiso ha sido borrado para sí mismo, pero no para el administrador.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar la notificación de que el permiso ha sido borrado, además de mostrarse ante el administrador.

Fuente: Elaboración propia.

Tabla 79. Caso de prueba 17

<b>Caso de prueba</b>	Solicitar nuevas horas extras
<b>Código del caso de uso</b>	CP - 17
<b>Objetivo</b>	Comprobar el funcionamiento del formulario para que los empleados soliciten horas extras.
<b>Resultado esperado</b>	Observar una notificación de error en caso de que alguno de los campos requeridos no sea válido.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar una notificación indicando que hay campos requeridos que son inválidos.

Fuente: Elaboración propia.

Tabla 80. Caso de prueba 18

<b>Caso de prueba</b>	Ver horas extras administradores
<b>Código del caso de uso</b>	CP - 18
<b>Objetivo</b>	Ver todas las solicitudes de horas extras realizadas por los empleados.
<b>Resultado esperado</b>	Cargar todas las horas extras que los empleados han solicitado.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar todas las horas extras solicitados y categorizándolas en pendientes, aceptadas y rechazadas.

Fuente: Elaboración propia.

Tabla 81. Caso de prueba 19

<b>Caso de prueba</b>	Ver horas extras empleados
<b>Código del caso de uso</b>	CP - 19
<b>Objetivo</b>	Ver todas las horas extras que ha solicitado cierto empleado.
<b>Resultado esperado</b>	Cargar todas las horas extras que el empleado ha solicitado.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar todas las horas extras solicitadas y categorizándolas en pendientes, aceptadas y rechazadas.

Fuente: Elaboración propia.

Tabla 82. Caso de prueba 20

<b>Caso de prueba</b>	Aceptar horas extra
<b>Código del caso de uso</b>	CP - 20
<b>Objetivo</b>	Aceptar una solicitud de horas extra con estado pendiente.
<b>Resultado esperado</b>	Mostrar una notificación indicando que la solicitud de horas extra ha sido aceptada.

<b>Resultado obtenido</b>	Resultado exitoso al mostrar la notificación de que la solicitud de horas extra ha sido aceptada.
---------------------------	---

Fuente: Elaboración propia.

Tabla 83. Caso de prueba 21

<b>Caso de prueba</b>	Rechazar horas extra
<b>Código del caso de uso</b>	CP - 21
<b>Objetivo</b>	Rechazar una solicitud de horas extra con estado pendiente o aceptado.
<b>Resultado esperado</b>	Mostrar una notificación indicando que la solicitud de horas extra ha sido rechazada.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar la notificación de que la solicitud de horas extra ha sido rechazada.

Fuente: Elaboración propia.

Tabla 84. Caso de prueba 22

<b>Caso de prueba</b>	Borrar horas extra administrador
<b>Código del caso de uso</b>	CP - 22
<b>Objetivo</b>	Borrar una solicitud de horas extra con cualquier tipo de estado.
<b>Resultado esperado</b>	Mostrar una notificación indicando que la solicitud ha sido borrada y que también fue borrada para el empleado.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar la notificación de que la solicitud ha sido borrada, además de no mostrarse ante el empleado.

Fuente: Elaboración propia.

Tabla 85. Caso de prueba 23

<b>Caso de prueba</b>	Editar horas extras
<b>Código del caso de uso</b>	CP - 23
<b>Objetivo</b>	Comprobar el funcionamiento del formulario para editar la solicitud de horas extra.
<b>Resultado esperado</b>	Observar una notificación mostrando un mensaje de error en caso de que algún campo del formulario sea inválido.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar una notificación indicando que hay campos requeridos que son inválidos.

Fuente: Elaboración propia.

Tabla 86. Caso de prueba 24

<b>Caso de prueba</b>	Borrar horas extra empleado
<b>Código del caso de uso</b>	CP - 24
<b>Objetivo</b>	Borrar una solicitud de horas extra con cualquier tipo estado siendo un empleado.
<b>Resultado esperado</b>	Mostrar una notificación indicando que la solicitud ha sido borrada para sí mismo, pero no para el administrador.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar la notificación de que la solicitud ha sido borrada, además de mostrarse ante el administrador.

Fuente: Elaboración propia.

Tabla 87. Caso de prueba 25

<b>Caso de prueba</b>	Cálculo salarial
<b>Código del caso de uso</b>	CP - 25
<b>Objetivo</b>	Cargar los detalles del cálculo al seleccionar uno o más empleados y un mes.

<b>Resultado esperado</b>	Carga los detalles del cálculo para cada empleado seleccionado.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar el detalle del cálculo para los empleados seleccionados.

Fuente: Elaboración propia.

Tabla 88. Caso de prueba 26

<b>Caso de prueba</b>	Nuevo aumento salarial
<b>Código del caso de uso</b>	CP - 26
<b>Objetivo</b>	Comprobar el funcionamiento del formulario para registrar un aumento salarial.
<b>Resultado esperado</b>	Observar una notificación de error en caso de que alguno de los campos del formulario sea inválido.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar una notificación indicando que campos completados son inválidos.

Fuente: Elaboración propia.

Tabla 89. Caso de prueba 27

<b>Caso de prueba</b>	Ver aumentos salariales
<b>Código del caso de uso</b>	CP - 27
<b>Objetivo</b>	Revisar los aumentos salariales realizados.
<b>Resultado esperado</b>	Cargar todos los aumentos salariales realizados.
<b>Resultado obtenido</b>	Resultado satisfactorio al mostrar los aumentos realizados.

Fuente: Elaboración propia.

Tabla 90. Caso de prueba 28

<b>Caso de prueba</b>	Nueva retención salarial
<b>Código del caso de uso</b>	CP - 28
<b>Objetivo</b>	Comprobar el funcionamiento del formulario para registrar una retención salarial.
<b>Resultado esperado</b>	Observar una notificación de error en caso de que alguno de los campos del formulario sea inválido.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar una notificación indicando que campos completados son inválidos.

Fuente: Elaboración propia.

Tabla 91. Caso de prueba 29

<b>Caso de prueba</b>	Ver retenciones salariales
<b>Código del caso de uso</b>	CP - 29
<b>Objetivo</b>	Revisar las retenciones salariales realizadas.
<b>Resultado esperado</b>	Carga todas las retenciones salariales realizadas.
<b>Resultado obtenido</b>	Resultado satisfactorio al mostrar las retenciones realizados.

Fuente: Elaboración propia.

Tabla 92. Caso de prueba 30

<b>Caso de prueba</b>	Borrar retención salarial
<b>Código del caso de uso</b>	CP - 30
<b>Objetivo</b>	Comprobar que la retención ha sido borrada satisfactoriamente.
<b>Resultado esperado</b>	Borrar la retención salarial siendo un administrador.

<b>Resultado obtenido</b>	Resultado satisfactorio al borrar la retención salarial.
---------------------------	--

Fuente: Elaboración propia.

Tabla 93. Caso de prueba 31

<b>Caso de prueba</b>	Nueva incapacidad
<b>Código del caso de uso</b>	CP - 31
<b>Objetivo</b>	Comprobar el funcionamiento del formulario para registrar una nueva incapacidad.
<b>Resultado esperado</b>	Observar una notificación de error en caso de que alguno de los campos del formulario sea inválido.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar una notificación indicando que campos completados son inválidos.

Fuente: Elaboración propia.

Tabla 94. Caso de prueba 32

<b>Caso de prueba</b>	Nuevo viático
<b>Código del caso de uso</b>	CP - 32
<b>Objetivo</b>	Comprobar el funcionamiento del formulario para registrar nuevos viáticos.
<b>Resultado esperado</b>	Observar una notificación de error en caso de que alguno de los campos del formulario sea inválido.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar una notificación indicando que campos completados son inválidos.

Fuente: Elaboración propia.

Tabla 95. Caso de prueba 33

<b>Caso de prueba</b>	Ver viáticos, administrador
<b>Código del caso de uso</b>	CP - 33
<b>Objetivo</b>	Revisar los viáticos registrados siendo administrador.
<b>Resultado esperado</b>	Carga todos los viáticos registrados.
<b>Resultado obtenido</b>	Resultado satisfactorio al mostrar los viáticos registrados.

Fuente: Elaboración propia.

Tabla 96. Caso de prueba 34

<b>Caso de prueba</b>	Ver viáticos, empleado
<b>Código del caso de uso</b>	CP - 34
<b>Objetivo</b>	Revisar los viáticos otorgados siendo un empleado.
<b>Resultado esperado</b>	Carga todos los viáticos otorgados.
<b>Resultado obtenido</b>	Resultado satisfactorio al mostrar los viáticos registrados.

Fuente: Elaboración propia.

Tabla 97. Caso de prueba 35

<b>Caso de prueba</b>	Borrar viáticos
<b>Código del caso de uso</b>	CP - 35
<b>Objetivo</b>	Comprobar que los viáticos han sido borrados satisfactoriamente.
<b>Resultado esperado</b>	Borrar los viáticos siendo un administrador.

<b>Resultado obtenido</b>	Resultado satisfactorio al mostrar una notificación indicando que los viáticos han sido borrados.
---------------------------	---

Fuente: Elaboración propia.

Tabla 98. Caso de prueba 36

<b>Caso de prueba</b>	Nueva amonestación
<b>Código del caso de uso</b>	CP - 36
<b>Objetivo</b>	Comprobar el correcto funcionamiento del formulario para registrar una nueva amonestación.
<b>Resultado esperado</b>	Observar una notificación de error en caso de que alguno de los campos del formulario sea inválido.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar una notificación indicando que campos completados son inválidos.

Fuente: Elaboración propia.

Tabla 99. Caso de prueba 37

<b>Caso de prueba</b>	Ver amonestaciones, administrador
<b>Código del caso de uso</b>	CP - 37
<b>Objetivo</b>	Revisar los viáticos registrados siendo administrador.
<b>Resultado esperado</b>	Carga todos los viáticos registrados.
<b>Resultado obtenido</b>	Resultado satisfactorio al mostrar los viáticos registrados.

Fuente: Elaboración propia.

Tabla 100. Caso de prueba 38

<b>Caso de prueba</b>	Ver amonestaciones, empleado
-----------------------	------------------------------

<b>Código del caso de uso</b>	CP - 38
<b>Objetivo</b>	Revisar los viáticos registrados siendo un empleado.
<b>Resultado esperado</b>	Carga todos los viáticos registrados.
<b>Resultado obtenido</b>	Resultado satisfactorio al mostrar los viáticos registrados.

Fuente: Elaboración propia.

Tabla 101. Caso de prueba 39

<b>Caso de prueba</b>	Nueva felicitación
<b>Código del caso de uso</b>	CP - 39
<b>Objetivo</b>	Comprobar el correcto funcionamiento del formulario para registrar una nueva felicitación.
<b>Resultado esperado</b>	Observar una notificación de error en caso de que alguno de los campos del formulario sea inválido.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar una notificación indicando que campos completados son inválidos.

Fuente: Elaboración propia.

Tabla 102. Caso de prueba 40

<b>Caso de prueba</b>	Ver felicitaciones, administrador
<b>Código del caso de uso</b>	CP - 40
<b>Objetivo</b>	Revisar las felicitaciones registradas siendo administrador.
<b>Resultado esperado</b>	Carga todas las felicitaciones registradas.
<b>Resultado obtenido</b>	Resultado satisfactorio al mostrar las felicitaciones registradas.

Fuente: Elaboración propia.

Tabla 103. Caso de prueba 41

<b>Caso de prueba</b>	Ranking de choferes
<b>Código del caso de uso</b>	CP - 41
<b>Objetivo</b>	Revisar el ranking de los choferes según las calificaciones que recibieron por parte de los clientes.
<b>Resultado esperado</b>	Cargar el ranking de los choferes.
<b>Resultado obtenido</b>	Resultado satisfactorio al mostrar el ranking de los choferes.

Fuente: Elaboración propia.

Tabla 104. Caso de prueba 42

<b>Caso de prueba</b>	Detalles de las calificaciones
<b>Código del caso de uso</b>	CP – 42
<b>Objetivo</b>	Revisar los detalles de las calificaciones que realizaron los clientes para determinado empleado.
<b>Resultado esperado</b>	Cargar los detalles de las calificaciones para el empleado seleccionado.
<b>Resultado obtenido</b>	Resultado satisfactorio al mostrar todos los detalles de las calificaciones.

Fuente: Elaboración propia.

Tabla 105. Caso de prueba 43

<b>Caso de prueba</b>	Nueva tarea
<b>Código del caso de uso</b>	CP - 43
<b>Objetivo</b>	Comprobar el correcto funcionamiento del formulario para registrar una nueva tarea.

<b>Resultado esperado</b>	Observar una notificación de error en caso de que alguno de los campos del formulario sea inválido.
<b>Resultado obtenido</b>	Resultado exitoso al mostrar una notificación indicando que campos completados son inválidos.

Fuente: Elaboración propia.

Tabla 106. Caso de prueba 44

<b>Caso de prueba</b>	Ver tareas, administrador
<b>Código del caso de uso</b>	CP - 44
<b>Objetivo</b>	Revisar las tareas registradas siendo administrador.
<b>Resultado esperado</b>	Cargar todas las tareas registradas y asignadas a empleados.
<b>Resultado obtenido</b>	Resultado satisfactorio al mostrar todas las tareas registradas.

Fuente: Elaboración propia.

Tabla 107. Caso de prueba 45

<b>Caso de prueba</b>	Ver tareas, empleado
<b>Código del caso de uso</b>	CP - 45
<b>Objetivo</b>	Revisar las tareas registradas siendo un empleado.
<b>Resultado esperado</b>	Cargar todas las tareas registradas y le han sido asignadas.
<b>Resultado obtenido</b>	Resultado satisfactorio al mostrar todas las tareas registradas y asignadas.

Fuente: Elaboración propia.

Tabla 108. Caso de prueba 46

<b>Caso de prueba</b>	Ver más información de la tarea
<b>Código del caso de uso</b>	CP - 46
<b>Objetivo</b>	Revisar información más detallada de la tarea seleccionada.
<b>Resultado esperado</b>	Cargar una ventana con la información detallada de la tarea.
<b>Resultado obtenido</b>	Resultado satisfactorio al mostrar la ventana con la información detallada.

Fuente: Elaboración propia.

Tabla 109. Caso de prueba 47

<b>Caso de prueba</b>	Borrar tarea
<b>Código del caso de uso</b>	CP - 47
<b>Objetivo</b>	Comprobar que la tarea ha sido borrada satisfactoriamente.
<b>Resultado esperado</b>	Borrar la tarea siendo un administrador.
<b>Resultado obtenido</b>	Resultado satisfactorio al mostrar una notificación indicando que la tarea ha sido borrada.

Fuente: Elaboración propia.

Tabla 110. Caso de prueba 48

<b>Caso de prueba</b>	Nueva solicitud de vacaciones
<b>Código del caso de uso</b>	CP - 48
<b>Objetivo</b>	Comprobar el correcto funcionamiento del formulario para registrar una nueva solicitud de vacaciones.
<b>Resultado esperado</b>	Observar una notificación de error en caso de que alguno de los campos del formulario sea inválido.

<b>Resultado obtenido</b>	Resultado exitoso al mostrar una notificación indicando que campos completados son inválidos.
---------------------------	---

Fuente: Elaboración propia.

## ANEXOS

### Anexo #1. Entrevista realizada



1. ¿Han adquirido un sistema de recursos humanos anteriormente?
2. ¿Qué procesos de recursos humanos realizan y que tan complejos son?
3. ¿Que desean realizar con el módulo de control de calidad de los choferes?
4. ¿Qué funcionalidad esperan para el cálculo de la planilla?
5. Relacionada a la anterior pregunta, ¿qué rubros utilizan para este cálculo?
6. ¿Qué requerimientos tienen para este tipo de sistema de información?

## REFERENCIAS

- Alegsa, L. (29 de Junio de 2016). *Definición de Requerimientos (desarrollo de software)*. Obtenido de alegsa.com.ar: <http://www.alegsa.com.ar/Dic/requerimientos.php>
- Alegsa, L. (31 de Julio de 2018). *Definición de aplicación web*. Obtenido de alegsa.com.ar: [http://www.alegsa.com.ar/Dic/aplicacion\\_web.php](http://www.alegsa.com.ar/Dic/aplicacion_web.php)
- Alegsa, L. (16 de Mayo de 2018). *Definición de Sistema informático*. Obtenido de Alegsa.com.ar: [http://www.alegsa.com.ar/Dic/sistema\\_informatico.php](http://www.alegsa.com.ar/Dic/sistema_informatico.php)
- Alvarez-Gayou Jurgenson, J. L. (2003). *Cómo hacer investigación cualitativa*. México, D. F.: Paidós.
- Apiumhub. (15 de Agosto de 2017). *Arquitectura de software y sus beneficios*. Obtenido de apiumhub.com: <https://apiumhub.com/es/tech-blog-barcelona/arquitectura-de-software/>
- Arcia, I. (7 de Enero de 2010). *La Investigación Científica: Las Variables*. Obtenido de investigadorcientifico.blogspot.com: <https://investigadorcientifico.blogspot.com/2010/01/las-variables.html>
- Asamblea Legislativa de Costa Rica. (7 de Julio de 2011). *Protección de la Persona frente al tratamiento de sus datos personales*. Obtenido de [http://www.pgrweb.go.cr/scij/Busqueda/Normativa/Normas/nrm\\_texto\\_completo.aspx?param1=NRTC&nValor1=1&nValor2=70975&nValor3=85989&param2=1&strTipM=TC&lResultado=3&strSim=simp](http://www.pgrweb.go.cr/scij/Busqueda/Normativa/Normas/nrm_texto_completo.aspx?param1=NRTC&nValor1=1&nValor2=70975&nValor3=85989&param2=1&strTipM=TC&lResultado=3&strSim=simp)
- Ayala, I. (27 de Marzo de 2017). *Lenguajes de programación*. Obtenido de ccm.net: <https://es.ccm.net/contents/304-lenguajes-de-programacion>
- Cabana, J. (1 de Agosto de 2017). *¿Qué es Node.js y para qué sirve?* Obtenido de drauta.com: <https://www.drauta.com/que-es-nodejs-y-para-que-sirve>
- Charte Ojeda, F. (2015). *SQL (Manual imprescindible)*. Madrid: Anaya Multimedia.
- García, A. (23 de Marzo de 2011). *Metodología de la Investigación*. Obtenido de <https://sites.google.com/site/metodologiadelainvestigacionb7/capitulo-5-sampieri>

- García, I. (11 de Diciembre de 2017). *Definición de Recursos Humanos*. Obtenido de economiasimple.net: <https://www.economiasimple.net/glosario/recursos-humanos>
- González, C. (13 de Abril de 2015). *Conceptos: Universo, Población y Muestra*. Obtenido de cgonzalez.cl: <https://www.cgonzalez.cl/conceptos-universo-poblacion-y-muestra/>
- Hernandez, U. (3 de Junio de 2018). *Qué es TypeScript*. Obtenido de codigofacilito.com: <https://www.codigofacilito.com/articulos/typescript>
- León, J. (15 de Marzo de 2017). *Investigación cuantitativa y cualitativa en comunicación*. Obtenido de tecdeinvestigacionycomunicacion.blogspot.com: <https://tecdeinvestigacionycomunicacion.blogspot.com/2017/03/cuantitativo-y-cualitativo-la.html>
- Maranto, M., & González, M. (Febrero de 2015). *Fuentes de Información*. Obtenido de <https://repository.uaeh.edu.mx/bitstream/bitstream/handle/123456789/16700/LECT132.pdf>
- Mdpstranieri. (15 de Junio de 2015). *Técnicas de recolección de datos*. Obtenido de proyectosinstitucionalesunitec.wordpress.com: <https://proyectosinstitucionalesunitec.wordpress.com/2015/06/15/capitulo-iii-tecnicas-de-recoleccion-de-datos/>
- Ministerio de Trabajo y Seguridad Social . (2020). *mtss.go.cr*. Obtenido de [http://www.mtss.go.cr/elministerio/marco-legal/documentos/Codigo\\_Trabajo\\_RPL.pdf](http://www.mtss.go.cr/elministerio/marco-legal/documentos/Codigo_Trabajo_RPL.pdf)
- Pacha, B. (10 de Noviembre de 2015). *Análisis Y Diseño: MODELO EN CASCADA*. Obtenido de ingenexescom.blogspot.com: <https://damian11eter.blogspot.com/2015/11/modelo-en-cascada.html>
- Pelhon, L. (27 de Mayo de 2019). *Qué es JavaScript: primeros pasos*. Obtenido de hostgator.mx: <https://www.hostgator.mx/blog/que-es-javascript/>
- Riquelme, M. (20 de Noviembre de 2017). *Nómina de la empresa (definición y características)*. Obtenido de webyempresas.com: <https://www.webyempresas.com/nomina-de-la-empresa/>

Torres, A. (27 de Mayo de 2018). *¿Qué es una API?* Obtenido de josefacchin.com: <https://josefacchin.com/que-es-rest-api-wordpress/#rest-api-%c2%bfque-es,-cuales-son-sus-ventajas-y-como-se-utiliza-en-wordpress?>

Universia. (04 de Septiembre de 2017). *Tipos de investigación: Descriptiva, Exploratoria y Explicativa.* Obtenido de noticias.universia.cr: <https://noticias.universia.cr/educacion/noticia/2017/09/04/1155475/tipos-investigacion-descriptiva-exploratoria-explicativa.html>

Vásquez, E. (23 de Marzo de 2018). *¿Qué es una base de datos, para qué sirve y que tipos hay?* Obtenido de mentecuriosa.net: <https://mentecuriosa.net/una-base-datos-sirve-tipos/>

You, E. (2017). *¿Qué es Vue.js?* Obtenido de es.vuejs.org: <https://es.vuejs.org/v2/guide/>