



**TECNOLÓGICO NACIONAL DE MÉXICO
CAMPUS SAN MARTÍN TEXMELUCAN
DIVISIÓN DE INGENIERÍA EN SISTEMAS
COMPUTACIONALES**

**SISTEMA WEB DE PARQUÍMETROS PARKING PARA EL MUNICIPIO
DE TLAXCALA DE XICOHTÉNCATL, EN EL ÁREA DE DESARROLLO
DE SOFTWARE DE LA EMPRESA SMARTSOFT AMERICA BUSINESS
APPLICATIONS S.A. DE C.V.**

TESIS



**QUE PARA OBTENER EL GRADO DE:
LICENCIADA EN INGENIERÍA EN SISTEMAS
COMPUTACIONALES**

PRESENTA:

**MITZI SOLLANGE HERNÁNDEZ JUÁREZ
18240013**

ASESOR: M.S.C. IVÁN RAFEL SÁNCHEZ JUÁREZ

SAN MARTÍN TEXMELUCAN, PUE. MAYO 2023

	Nombre del documento: Dictamen para Titulación Integral	Código: ITSSMT-AC-NOR-01-FO-04	
	Referencia del Documento: Lineamiento para la Titulación Integral	Revisión: 1 Página 1 de 1	

San Martín Texmelucan, Pue., a 16 de enero de 2023
ASUNTO: Dictamen para Titulación Integral

C. MITZI SOLLANGE HERNANDEZ JUÁREZ
PASANTE DE LA CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES
P R E S E N T E

En respuesta a su solicitud de titulación integral con el proyecto **“Sistema web de parquímetros Parking para el estado de Tlaxcala, en el área de desarrollo de software de la empresa SmartSoft America Business Applications SA de CV.,** me es grato informarle que fue **aceptado**, mediante la modalidad **TESIS PROFESIONAL**, y se confirma como asesor al **Mtro. Iván Rafael Sánchez Juárez.**

Por lo que le solicito ponerse en contacto con su asesor, en caso de ser necesario.

Además, le informo que deberá pasar al Departamento de Control Escolar, a que le revisen su documentación.



ATENTAMENTE

Excelencia en Educación Tecnológica
“Formación Tecnológica de Vanguardia para el Desarrollo Regional”



MTRA. ALTAGRACIA BERRUECOS XICOHTÉNCATL
JEFA DE LA DIVISIÓN DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

c.c.p. Subdirección Académica. - PSC
 Departamento de Control Escolar. - PSC
 Expediente

	Nombre del documento: Autorización de entrega de informe de Titulación.	Código: ITSSMT-AC-NOR-01-FO-08	
	Referencia del Documento: Lineamiento para la Titulación Integral	Revisión: 2	

San Martín Texmelucan, Puebla a **12 de mayo** de 2023

**MITZI SOLLANGE HERNÁNDEZ JUÁREZ
PASANTE DE LA CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES
P R E S E N T E**

De acuerdo con la normatividad vigente de nuestro instituto y habiendo cumplido con todas las indicaciones que el comité revisor le hizo respecto a su informe de titulación integral titulado: **"Sistema web de parquímetros Parking para el estado de Tlaxcala, en el área de desarrollo de software de la empresa SmartSoft America Business Applications SA de CV."** comunico a Usted que:

HA SIDO LIBERADA Y SE LE AUTORIZA PARA QUE PROCEDA A LA FORMALIZACIÓN DE ESTE

Para lo cual deberá entregar su documento digital a la Subdirección de Estudios Profesionales, en un plazo de diez días hábiles contados a partir de la fecha del presente.



Es importante mencionar que usted deberá elegir la manera en que conservará su trabajo de titulación (libro, disco, etc.).

ATENTAMENTE

*Excelencia en Educación Tecnológica
"Formación Tecnológica de Vanguardia para el Desarrollo Regional"*


**MTRA. ALTAGRACIA BERRUECOS XICHOHTENCATL
JEFA DE LA DIVISIÓN DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

c.c.p. Expediente

	Nombre del documento: Liberación de Proyecto para la Titulación Integral	Código: ITSSMT-AC-NOR-01-FO-03	
	Referencia del Documento: Lineamiento para la Titulación Integral	Revisión: 2	

San Martín Texmelucan, Pue., a **12 de mayo** de 2023
 Asunto: Liberación de proyecto para la titulación integral.

**SUBDIRECCIÓN ACADÉMICA DEL
 INSTITUTO TECNOLÓGICO SUPERIOR DE SAN MARTÍN TEXMELUCAN
 PRESENTE**

Por este medio informo que ha sido liberado el siguiente proyecto para la titulación integral:

Nombre del egresado:	Mitzi Sollange Hernández Juárez
Carrera:	Ingeniería en Sistemas Computacionales
No. de control:	18240013
Nombre del proyecto:	"Sistema web de parquímetros Parking para el estado de Tlaxcala, en el área de desarrollo de software de la empresa SmartSoft America Business Applications SA de CV."
Producto:	Tesis Profesional

Agradezco de antemano su valioso apoyo en esta importante actividad para la formación profesional de nuestros/as egresados/as.

ATENTAMENTE

*Excelencia en Educación Tecnológica
 "Formación Tecnológica de Vanguardia para el Desarrollo Regional"*



**MTRA. ALTAGRACIA BERRUECOS XICHOHTENCATL
 JEFA DE LA DIVISIÓN DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

Nombre y firma del asesor	Nombre y firma de la revisora	Nombre y firma de la revisora
 Mtro. Iván Rafael Sánchez Juárez.	 Mtra. María Petra Paredes Xochihua	 Mtra. Vianney Morales Zamora

c.c.p.- Expediente

Agradecimientos

En primer lugar, a mi padre Fidel Hernández y mis hermanos Alan y Emmanuel, agradezco por todo el apoyo incondicional que me han brindado durante el desarrollo del proyecto y a lo largo de la carrera, por la confianza que han puesto en mí y las palabras de aliento que me han brindado.

Por otro lado, agradezco a mis amigos Mildred Jocelyn Dueñas y Alberto Contreras, que me han ayudado a crecer de manera personal, académica y profesional a lo largo de la carrera, brindándome su apoyo y orientación.

Le agradezco profundamente al Lic. Iván Rafael Sánchez, por la orientación, consejos y apoyo a lo largo del desarrollo del presente proyecto y a lo largo de la carrera, brindándome las bases y conocimientos. También a la profesora María Petra Paredes Xochihua, por los consejos y conocimientos brindados a lo largo de la carrera.

En memoria de mi madre Catalina Juárez, dedico este trabajo, dándome las fuerzas necesarias para culminar mi carrera, siendo mi motivación para seguir adelante y superarme cada día más.

Índice general

Capítulo I Introducción.....	1
1.1. Introducción	2
1.2. Antecedentes.....	2
1.2.1. Parkimovil.....	2
1.2.2. Parquímetro Parkum	3
1.2.3. Mueve Ciudad	4
1.2.4. Blinkay.....	4
1.2.5. iParkMe	5
1.3. Problema de Investigación.....	5
1.4. Justificación	6
1.5. Propósito del Estudio.....	7
1.6. Objetivos del Estudio	7
1.6.1. Objetivo General.....	7
1.6.2. Objetivos Específicos	7
1.7. Preguntas de Investigación.....	8
1.8. Planteamiento de Hipótesis	8
Capítulo II Revisión de la Literatura.....	9
2.1. Conceptos Básicos	10
2.1.1. Multa.....	10
2.1.2. Pago.....	10
2.1.3. Vehículo	10
2.1.4. Parquímetro.....	10
2.1.5. Ticket.....	10
2.1.6. Inmovilizador	11

2.1.7.	Araña Inmovilizadora.....	11
2.1.8.	Conductor.....	11
2.2.	Metodología de Desarrollo.....	11
2.2.1.	Scrum.....	11
2.3.	Base de Datos	13
2.3.1.	MySQL WorkBench.....	13
2.4.	Lenguajes de Programación.....	14
2.4.1.	Java.....	14
2.4.2.	XHTML	15
2.4.3.	CSS.....	16
2.4.4.	AJAX	17
2.4.5.	SQL	17
2.5.	Servidor Web.....	20
2.5.1.	GlassFish	20
2.6.	Herramientas de Desarrollo.....	20
2.6.1.	NetBeans.....	20
2.6.2.	Google Chrome	21
2.7.	Bibliotecas	21
2.7.1.	IText	21
2.7.2.	PrimeFaces	21
2.8.	Herramientas de Diseño.....	22
2.8.1.	Balsamiq Mockups	22
2.9.	Java Persistence API.....	22
2.9.1.	Entity	23
2.9.2.	Persistencia de Campos.....	23

2.9.3. Mapeo de Relaciones.....	24
2.9.4. Entity Manager	24
2.9.5. Unidades de Persistencia.....	24
2.9.6. Arquitectura de JPA.....	24
Capítulo III Metodología	27
3.1. Participantes y/o Representación Estadística.....	28
3.2. Fuentes de Información	29
3.2.1. Tlaxcala Digital	29
3.2.2. Catálogo Estatal de Trámites y Servicios del Gobierno de Tlaxcala	29
3.2.3. Instituto Nacional de Estadística y Geografía (INEGI).....	30
3.3. Técnicas e Instrumentos.....	30
3.3.1. Conductor.....	31
3.3.2. Inmovilizador	32
3.3.3. Revisor	32
3.4. Procedimientos	33
3.5. Aplicación de Metodología Scrum	40
3.6. Planificación de Iteraciones	41
3.7. Ejecución Sprint 1	46
3.8. Resultados Obtenidos del Sprint 1	47
3.8.1. Instalar Navegador	47
3.8.2. Instalar Apache NetBeans 8.2.....	48
3.8.3. Instalar GlassFish 4.1.....	48
3.8.4. Instalar Software Manejador de Base de Datos MySQL WorkBench.....	49
3.8.5. Instalar Balsamiq.....	50
3.8.6. Diseñar Pantalla de Baja Fidelidad Módulo Inicio de Sesión.....	50

3.8.7.	Diseñar Pantalla de Baja Fidelidad Módulo Registro	51
3.8.8.	Diseñar Pantallas de Baja Fidelidad Catálogo Conductor	52
3.8.9.	Diseñar Pantallas de Baja Fidelidad Catálogo Revisor	55
3.8.10.	Diseñar Pantallas de Baja Fidelidad Catálogo Inmovilizador	56
3.8.11.	Diseñar Pantallas de Baja Fidelidad Catálogo Vendedor.....	59
3.8.12.	Diseñar Pantallas de Baja Fidelidad Catálogo Administrador	61
3.9.	Burndown Chart Sprint 1	72
3.10.	Ejecución Sprint 2.....	73
3.11.	Resultados Obtenidos del Sprint 2	75
3.11.1.	Diseñar Pantalla de Alta Fidelidad Módulo Inicio de Sesión	75
3.11.2.	Diseñar Pantalla de Alta Fidelidad Módulo Registro	76
3.11.3.	Diseñar Pantalla de Alta Fidelidad Catálogo de Conductor.....	77
3.11.4.	Diseñar Pantalla de Alta Fidelidad Catálogo de Revisor.....	78
3.11.5.	Diseñar Pantalla de Alta Fidelidad Catálogo de Inmovilizador	79
3.11.6.	Diseñar Pantalla de Alta Fidelidad Catálogo de Vendedor.....	79
3.11.7.	Diseñar Pantalla de Alta Fidelidad Catálogo de Administrador	80
3.11.8.	Creación de View	80
3.11.9.	Creación de Controller	81
3.12.	Burndown Chart Sprint 2	82
3.13.	Ejecución Sprint 3.....	83
3.14.	Resultados Obtenidos del Sprint 3	85
3.14.1.	Creación de Entity	85
3.14.2.	Creación de DAO	87
3.14.3.	Creación de Business	88
3.14.4.	Descargas de Documentos Excel	90

3.14.5.	Descargas de Documentos PDF	92
3.15.	Burndown Chart Sprint 3	93
3.16.	Ejecución Sprint 4.....	94
3.17.	Resultados Obtenidos del Sprint 4	96
3.17.1.	Mensajes.....	96
3.17.2.	Formato de Reportes	98
3.18.	Burndown Chart Sprint 4	98
Capítulo IV Resultados.....		100
4.1.	Resultados de Encuesta de Sistema Implementado	101
4.2.	Comparación de Tiempos Entre Sistemas	104
4.3.	Verificación de Hipótesis	109
4.4.	Respuesta a Preguntas de Investigación	110
Capítulo V Discusión		111
5.1.	Resultados Obtenidos & Esperados	112
5.2.	Conclusiones	113
5.3.	Recomendaciones Y Aportación.....	114
Referencias Bibliográficas		115
GLOSARIO.....		118
ANEXO A.....		126
CONVENIO DE CONFIDENCIALIDAD		126

Índice de tablas

Tabla 1. Muestra para encuesta	28
Tabla 2. Requerimientos funcionales	40
Tabla 3. Requerimientos no funcionales	40
Tabla 4. Product Backlog.....	41
Tabla 5. Sprint 1	46
Tabla 6. Sprint 2	74
Tabla 7. Código de ejemplo de View	80
Tabla 8. Código de ejemplo de Controller	81
Tabla 9. Sprint 3	84
Tabla 10. Código de ejemplo de Entity	85
Tabla 11. Código de ejemplo de DAO	87
Tabla 12. Código de ejemplo de Business	88
Tabla 13. Dependencias para usar DataExporter.....	90
Tabla 14. Código de ejemplo de descargas de documentos Excel con DataExporter ..	91
Tabla 15. Código de ejemplo de descargas de documentos PDF con IText	92
Tabla 16. Sprint 4	94
Tabla 17. Código de ejemplo de mensajes en sección XHTML	96
Tabla 18. Código de ejemplo de mensajes en sección View.....	97
Tabla 19. Tiempos de sistemas para conductor	104
Tabla 20. Tiempo promedio de sistemas para conductor	106
Tabla 21. Tiempos de sistemas para revisor	107
Tabla 22. Tiempo promedio de sistemas para revisor	108
Tabla 23. Tiempos de sistemas para inmovilizador.....	108
Tabla 24. Tiempo promedio de sistemas para inmovilizador.....	109

Índice de figuras

Figura 1. Arquitectura de JPA.....	25
Figura 2. ¿Qué tiempo le toma la compra de tiempo para el parquímetro?	33
Figura 3. ¿Qué tiempo le toma la recarga de tiempo para el parquímetro?	34
Figura 4. En caso de ser acreedor a una multa, ¿Qué tiempo le toma la paga de esta?	34
Figura 5. En caso de tener la araña inmovilizadora en su vehículo, ¿Qué tiempo se toma el personal para retirarlo?.....	34
Figura 6. ¿Cómo calificaría la facilidad de uso del sistema de parquímetro? Donde 1 es la puntuación mínima y 5 la máxima	35
Figura 7. ¿Cree que el proceso de compra de tiempo es intuitivo?.....	35
Figura 8. ¿Ha experimentado algún error o problema técnico al utilizar el sistema de parquímetro?	35
Figura 9. ¿Se siente informado adecuadamente sobre las políticas y reglas del uso del parquímetro?	36
Figura 10. ¿Cuál es el tiempo de respuesta del sistema al ingresar la placa del vehículo para su revisión?	36
Figura 11. ¿Qué tiempo le toma el solicitar la inmovilización de un vehículo?	37
Figura 12. ¿Qué tiempo se toma el inmovilizador en aceptar su solicitud?	37
Figura 13. ¿Cree que es eficiente el sistema de solicitud de inmovilizaciones?.....	37
Figura 14. ¿Ha experimentado algún error o problema técnico al utilizar el sistema de parquímetro?	38
Figura 15. ¿Qué tiempo le toma verificar el pago de una multa?	38
Figura 16. ¿Qué tiempo le toma el retiro de una araña inmovilizadora?	39
Figura 17. ¿Cree que es eficiente el sistema de solicitud de inmovilizaciones?.....	39
Figura 18. ¿Ha experimentado algún error o problema técnico al utilizar el sistema de parquímetro?	39
Figura 19. Instalación de navegador Google Chrome	48
Figura 20. Instalación de Apache NetBeans 8.2.....	48
Figura 21. Instalación de GlassFish 4.1	49
Figura 22. Instalación de MySQL Workbench	49

Figura 23. Instalación de Balsamiq.....	50
Figura 24. Pantalla de baja fidelidad inicio de sesión	51
Figura 25. Pantalla de baja fidelidad registro de conductor	51
Figura 26. Pantalla de baja fidelidad catálogo conductor	52
Figura 27. Pantalla de baja fidelidad compra de tiempo	53
Figura 28. Pantalla de baja fidelidad para registrar vehículo	53
Figura 29. Pantalla de baja fidelidad para editar información del conductor	54
Figura 30. Pantalla de baja fidelidad historial de saldo.....	55
Figura 31. Pantalla de baja fidelidad catálogo de revisor	55
Figura 32. Pantalla de baja fidelidad búsqueda de vehículo	56
Figura 33. Pantalla de baja fidelidad catálogo inmovilizador	57
Figura 34. Pantalla de baja fidelidad multa pagada.....	57
Figura 35. Pantalla de baja fidelidad histórico de inmovilizaciones	58
Figura 36. Pantalla de baja fidelidad opción	58
Figura 37. Pantalla de baja fidelidad catálogo vendedor	59
Figura 38. Pantalla de baja fidelidad búsqueda de cuenta	60
Figura 39. Pantalla de baja fidelidad venta saldo	60
Figura 40. Pantalla de baja fidelidad pago de vehículo	61
Figura 41. Pantalla de baja fidelidad catálogo administrador sección vendedores.....	62
Figura 42. Pantalla de baja fidelidad editar información de vendedor	62
Figura 43. Pantalla de baja fidelidad nuevo vendedor.....	63
Figura 44. Pantalla de baja fidelidad catálogo administrador sección revisores.....	63
Figura 45. Pantalla de baja fidelidad editar información de revisor	64
Figura 46. Pantalla de baja fidelidad nuevo revisor	64
Figura 47. Pantalla de baja fidelidad catálogo administrador sección inmovilizador	65
Figura 48. Pantalla de baja fidelidad editar información de inmovilizador	65
Figura 49. Pantalla de baja fidelidad nuevo inmovilizador	66
Figura 50. Pantalla de baja fidelidad catálogo administrador sección conductor.....	66
Figura 51. Pantalla de baja fidelidad catálogo administrador sección costos del sistema	67
Figura 52. Pantalla de baja fidelidad nuevo costo del sistema	68

Figura 53. Pantalla de baja fidelidad editar información costos.....	68
Figura 54. Pantalla de baja fidelidad de opción	69
Figura 55. Pantalla de baja fidelidad catálogo administrador sección calles	69
Figura 56. Pantalla de baja fidelidad nueva calle	70
Figura 57. Pantalla de baja fidelidad editar información calle.....	70
Figura 58. Pantalla de baja fidelidad catálogo administrador sección cajones	71
Figura 59. Pantalla de baja fidelidad nuevo cajón	71
Figura 60. Pantalla de baja fidelidad editar información cajón.....	72
Figura 61. Burndown Chart Sprint 1	73
Figura 62. Pantalla de alta fidelidad inicio de sesión	76
Figura 63. Pantalla de alta fidelidad registro de conductor	77
Figura 64. Pantalla de alta fidelidad catálogo conductor	78
Figura 65. Pantalla de alta fidelidad catálogo revisor	78
Figura 66. Pantalla de alta fidelidad catálogo inmovilizador	79
Figura 67. Pantalla de alta fidelidad catálogo vendedor	79
Figura 68. Burndown Chart Sprint 2	83
Figura 69. Burndown Chart Sprint 3	93
Figura 70. Burndown Chart Sprint 4	99
Figura 71. Resultados de encuesta a conductores.....	103
Figura 72. Resultados de encuesta a inmovilizadores	103
Figura 73. Resultados de encuesta a revisores.....	104

CAPÍTULO I

INTRODUCCIÓN

1.1. Introducción

El sistema web de parquímetros parking para el municipio de Tlaxcala de Xicohténcatl, es un sistema que busca la reducción de tiempos para el conductor en los procesos de compra y recarga de tiempo, pago de multas y espera de retiro de la araña inmovilizadora.

Por otro lado, se pretende reducir los tiempos de solicitud de inmovilización por parte del revisor, verificación del pago de multas y retiro de araña por parte de inmovilizador.

Se busca una buena experiencia del usuario dentro del sistema, un sitio intuitivo con el fin de que cualquier usuario pueda navegar dentro de este. Esto se logró por medio de hojas de estilo CSS junto con la librería PrimeFaces.

Con el fin de tener un orden durante este, es necesario el uso de una metodología de desarrollo. Por lo que se usó la metodología ágil Scrum, donde se organizaron los sprints y las tareas a cumplir dentro de estas.

Por último, el proyecto está basado en un modelo de persistencia de datos, este fue creado por la empresa SmartSoft America Business Applications S.A. de C.V., el cual consiste de un total de 5 clases: View, Controller, Business, Entity y DAO.

1.2. Antecedentes

El sistema de parquímetros del municipio de Tlaxcala de Xicohténcatl, cuenta con deficiencias ya que no procesa de forma inmediata los pagos de multa, la solicitud de inmovilización, petición de retiro de araña inmovilizadora, revisión de pago de vehículos y no contempla la compra de tiempo desde su dispositivo y generación de reportes.

1.2.1. Parkimovil

Según la empresa Parkimovil (2020) es una aplicación móvil disponible para dispositivos Android y iOS. Dispone de diferentes soluciones para ciudades inteligentes,

tales como: estacionamiento digital, parquímetros virtuales, control de acceso, valet parking y control de cobro de tianguis.

Funciona sin necesidad de máquinas, todo es por medio del dispositivo móvil, recibe notificaciones cuando el tiempo este por vencer, recarga el tiempo desde este mismo sin necesidad de regresar al sitio de estacionamiento.

Cuenta con un total de 4 formas de pago, la primera es por medio del registro de tarjetas de crédito o débito dentro de la aplicación, compra de saldo en los negocios autorizados, en caso de no contar con un teléfono inteligente se puede enviar un mensaje de texto en el que se ingresa el número de placas del vehículo, por último, se tiene la opción de pagar el tiempo en negocios autorizados.

1.2.2. Parquímetro Parkum

De acuerdo a Parkum (2019) es una aplicación móvil gratuita disponible para dispositivos Android y iOS. Tiene el fin de facilitar y agilizar el pago de parquímetros por medio de un dispositivo móvil, no es necesario dejar el ticket en al vehículo a la hora de la compra y recarga de más tiempo.

La aplicación funciona de la siguiente manera, una vez instalada la aplicación se selecciona la opción de “Iniciar aparcamiento”, seguido de esto se seleccionan las placas del vehículo, por último y antes de su retiro es importante que se seleccione la opción “Terminar aparcamiento”, ya que una de las características de esta aplicación es que solo se cobra el tiempo usado.

Otra de las características de este sistema es que no se maneja efectivo, las recargas y pagos solo se realizan por medio de tarjetas de crédito y débito, una alternativa a esto es el uso de una aplicación intermediaria llamada “ePagum”, donde se realizan el pago de multas y recargas para esta aplicación.

Dentro del funcionamiento de estos parquímetros se encuentran dos encargados, el primero son los sancionadores, estos asignan multas en caso de que no se pague el

tiempo de estancia y colocan la araña inmovilizadora, el segundo es el encargado de retirar el inmovilizador colocado con el fin de optimizar el tiempo al usuario.

1.2.3. Mueve Ciudad

Mueve ciudad (2020) cuenta con una aplicación móvil gratuita que se encuentra en tiendas App Store, Google Play y próximamente en App Galery, además cuenta con un sitio web en el que solo se puede realizar recargas de saldo a tu cuenta.

El uso del parquímetro en esta aplicación funciona de la siguiente forma, se selecciona la placa de un vehículo previamente registrado o se introducen las placas, seguido se selecciona el tiempo a comprar y por último se realiza el pago del tiempo, además de que debes aceptar la comisión que se cobra. Cuando se quiera extender el tiempo contará como una nueva compra y por lo tanto se cobrará nuevamente una comisión.

Cuenta con un total de 4 formas de pago, la primera es por medio del registro de tu tarjeta dentro de la aplicación, otra es por medio de su página web ingresando una tarjeta bancaria, la siguiente es por transferencia de pagos SPEI, por último, se permite el pago en efectivo por medio de los puntos de venta y recarga.

En la parte de los pagos se cobra una comisión a la hora de realizar recargas, además de que el máximo por recarga al día es de \$500, en caso de requerir una recarga más grande tiene que hacerse por medio de transferencia SPEI.

1.2.4. Blinky

La aplicación móvil Blinky (2023) se encuentra de forma gratuita para dispositivos Android y iOS, no cuenta con locales para la compra y recarga de tiempo.

El uso de este parquímetro es totalmente por medio de la aplicación, una vez se haya dado de alta el usuario y el vehículo, se tiene que realizar una recarga, seguido permitirá la compra del tiempo. Cuenta con dos opciones, una para realizar una recarga del tiempo y otra para detenerlo.

Una de las características de este sistema es que solo se paga el tiempo que se usa, en caso de tener tiempo de sobra y parar el tiempo, automáticamente regresará el dinero del tiempo ya comprado. Otra de las características del sistema es que se tiene un tiempo de 15 minutos para que puedas dejar el lugar de estacionamiento.

A la hora de realizar recargas se tiene que realizar por medio de PayPal, tarjeta bancaria y OXXO, no dispone del método de pago en efectivo o de tiendas de recarga de saldo.

1.2.5. iParkMe

Según la aplicación iParkMe (2023) se encuentra disponible en dispositivos iOS, Android y Windows Phone, además de que cuenta con un sitio web para recargas.

El uso de esta aplicación se debe tener un registro previo, ya sea en la misma aplicación o en su sitio web, también se ingresa el método de pago, este puede ser por una tarjeta bancaria, PayPal o recargas por medio de su página.

Una de las características de esta aplicación es que solo se paga por el tiempo usado, una vez se pare el tiempo se bonifica el restante, por otro lado, no hay multa si el tiempo ha vencido y no ha pasado los 15 minutos.

1.3. Problema de Investigación

El estado de Tlaxcala cuenta con un sistema de parquímetros, donde la compra del tiempo debe ser mediante los parquímetros previamente instalados en la calle, cada vez que se requiera más tiempo el conductor se tendrá que dirigir a su parquímetro correspondiente y agregar el ticket a su vehículo.

Por otro lado, en caso de que el conductor sea acreedor a una multa, deberá pagarla en una de las oficinas del gobierno, solicitar el retiro de la araña, esperar que su solicitud sea aceptada por el inmovilizador para finalmente proceder a su retiro, este largo y tardado proceso afecta en la calidad del servicio al conductor.

El revisor verifica manualmente cada ticket de los vehículos con el fin de encontrar aquel vehículo con tiempo vencido, en caso de encontrarlo tendrá que levantar una multa y la colocación de la araña inmovilizadora.

Por último, el inmovilizador procede al retiro de las arañas una vez que se confirme el pago de la multa por parte del conductor.

La falta de pagos y verificación de multas automatizado hace que el conductor tenga una pérdida de tiempo significativa, por otro lado, se puede realizar una mejora en el proceso de pago de multas que involucre al inmovilizador con el fin de reducir el tiempo de espera.

1.4. Justificación

Actualmente el municipio de Tlaxcala de Xicohtécatl cuenta con un sistema de parquímetros, sin embargo, uno de los problemas que se tienen es la compra de tiempo extra, debido a que cada que el tiempo está próximo a terminar el conductor debe acercarse a su parquímetro correspondiente, realizar nuevamente el pago, posteriormente dirigirse a su vehículo y agregarlo.

Por otro lado, el revisor debe dirigirse a cada vehículo, validar el ticket otorgado por el parquímetro, verificar si es acreedor a una multa y posteriormente mandar al inmovilizador para añadir la araña inmovilizadora.

Por último, el inmovilizador tendrá que trasladarse a la localización del vehículo y ponerle la araña, así mismo, tiene que esperar el pago de la multa del conductor para proceder a quitar la araña.

El uso de un sistema de parquímetros web Parking, mejora y facilita los procesos de compra, venta de tiempo, pago de multas, por medio de la interfaz del conductor, esta cuenta con la visualización de sus vehículos registrados, de esta forma podrá comprar tiempo de uno o más vehículos, verificar su estatus y pagar sus multas.

La interfaz del revisor cuenta con un módulo de búsquedas donde ingresará las placas del vehículo, por medio de la búsqueda podrá visualizar la información general, si

su tiempo ha terminado, y la ubicación exacta. De acuerdo a su estatus podrá mandar una solicitud de inmovilización.

El inmovilizador tendrá una interfaz donde visualizará y atenderá las solicitudes realizadas por el revisor, además de las fecha y hora de la solicitud, pago de multas y la colocación de la araña. Para levantar la araña podrá ver el estatus de la multa, si se ha realizado o no el pago, así como el día y la hora en que se realizó.

1.5. Propósito del Estudio

El objetivo de este sistema es desarrollar un sistema web de parquímetros, en el cual se busca reducir el tiempo de compra, recarga de tiempo, retiro de araña inmovilizadora y pago de multas al conductor; mejorar la comunicación entre el inmovilizador y el revisor.

1.6. Objetivos del Estudio

Los objetivos de estudios es una base sobre la cual se realizará una investigación, nos permiten conocer la razón de la solución ante un problema. Por otro lado, se clasifican en objetivo general y objetivos específicos.

1.6.1. Objetivo General

Desarrollar un sistema web de parquímetros Parking en el municipio de Tlaxcala de Xicohtécatl, utilizando Java EE, para la mejora del proceso de compra y venta de tiempo, pago de multas, revisión del tiempo, solicitud e inmovilización del vehículo, así como la generación de reportes.

1.6.2. Objetivos Específicos

- Analizar los requerimientos obtenidos por la empresa SmartSoft America Business Applications S.A. de C.V.
- Diseñar las interfaces de baja fidelidad para el sistema web Parking.

- Desarrollar las interfaces de alta fidelidad de administrador, vendedor, revisor, inmovilizador y conductor del sistema web Parking en el estado de Tlaxcala.
- Implementar la visualización, actualización, eliminación y registro de la información a los respectivos módulos con el gestor de base de datos MySQL.
- Implementar la generación de reportes PDF y Excel, con el uso de las librerías IText y PrimeFaces que permita la visualización de información de los módulos.
- Presentar el sistema web de forma local, sin conexión al servidor del estado de Tlaxcala.

1.7. Preguntas de Investigación

¿Cómo mejorar y optimizar el proceso para la compra y revisión de tiempo en parquímetros e inmovilización de vehículos del municipio de Tlaxcala de Xicohtécatl?

¿Qué tan factible es utilizar una aplicación web para el control de parquímetros en comparación del proceso actual?

1.8. Planteamiento de Hipótesis

Con el uso del sistema web Parking se estima la mejora de tiempo de compra y recarga en un 50% respecto al método que se usa actualmente en el municipio de Tlaxcala de Xicohtécatl, además de la mejora de comunicación y tiempo entre los revisores y los inmovilizadores, haciendo que el tiempo de espera del conductor se reduzca en un 30%.

CAPÍTULO II REVISIÓN DE LA LITERATURA

2.1. Conceptos Básicos

Los conceptos ayudan a entender la terminología y tener el contexto correcto de los términos utilizados durante el desarrollo del proyecto.

2.1.1. Multa

Una multa es una sanción económica que se aplica a un conductor que ha cometido una infracción de tráfico. Esta se aplica cuando el vehículo se haya estacionado en un lugar no permitido, que permanezca un tiempo superior al permitido o no haga el pago realizado en su respectivo parquímetro.

2.1.2. Pago

Pago es toda aquella acción que realizamos para extinguir o cancelar una obligación. Se basa en la entrega de un bien, servicio o activo financiero a cambio de otro bien, servicio o activo financiero (Pedrosa, 2020).

2.1.3. Vehículo

Un vehículo es el medio de transporte que es utilizado por el conductor para desplazarse hacia ciertos puntos de la ciudad, usado para transportar personas o cosas. Se clasifican en dos tipos, vehículos con o sin motor. Por otro lado, se clasifican de acuerdo al tipo de transporte.

2.1.4. Parquímetro

El parquímetro es una máquina que se utiliza para regular y controlar el estacionamiento en la vía pública. Según Hernández (2018), la regulación del estacionamiento mediante parquímetros permite un mejor aprovechamiento del espacio público y una reducción del tráfico generado por la búsqueda de estacionamiento.

2.1.5. Ticket

El ticket es un comprobante de pago que se obtiene tras realizar un pago en las casetas de los parquímetros. Este tiene que ser colocado dentro del vehículo en un lugar

visible para indicar que se ha realizado su respectivo pago de tiempo y no sea acreedor a una multa.

2.1.6. Inmovilizador

El inmovilizador es el encargado de aplicar o retirar la araña inmovilizadora de un vehículo según sea el caso. Este debe verificar los pagos de multas que se hayan generado al conductor para que pueda librar el vehículo o no.

2.1.7. Araña Inmovilizadora

La araña inmovilizadora es un tipo de inmovilizador que se utiliza para inmovilizar el vehículo mediante la colocación de unas pinzas en las ruedas. Según Morales (2017), la aplicación de la araña inmovilizadora es una medida disuasoria efectiva para la prevención de conductas de estacionamiento ilegal.

2.1.8. Conductor

El conductor conduce el vehículo y es el principal responsable de el pago de parquímetros y multas en caso de tenerlas. Este debe cumplir correctamente con las normas de tránsito de la localidad en la que conduce.

2.2. Metodología de Desarrollo

Una metodología de desarrollo son técnicas y métodos, los cuales aplicamos para la creación de soluciones de software. Su objetivo principal es la planeación y organización de equipos de trabajo, con el fin de desarrollar en conjunto y de forma exitosa un proyecto de software.

2.2.1. Scrum

Sean Peek nos menciona que “La metodología Agile Scrum es un sistema de gestión de proyectos basado en sprints con el objetivo de ofrecer el mayor valor a las partes interesadas” (Peek, 2022).

Scrum es una metodología ágil, consta de secciones llamadas sprints, estos son bloques trabajo temporales, cortos y fijos, al final de cada bloque se debe tener un resultado, es decir una parte del proyecto que se está realizando. Una de las características de esta metodología son sus reuniones diarias o también llamadas “daily meeting”, en las cuales se muestra el avance que se ha tenido con el día a día, suelen ser reuniones cortas.

Equipo

Para la aplicación de esta metodología es necesario delegar responsabilidades, los cuales se encargarán de ciertas tareas en específico. Encontramos tres roles: Scrum master, propietario del producto o Product Owner y equipo de desarrollo.

Scrum Master. La persona que lidera al equipo orientándolos a cumplir con las reglas y procesos de la metodología. Scrum master gestiona la reducción de impedimentos del proyecto y trabaja con el Product Owner para maximizar el ROI (Digité, 2022). Este mantiene el Scrum actualizado y brinda apoyo en caso de que lo requieran.

Propietario Del Producto. Representa a las partes interesadas, que suelen ser clientes. Para garantizar que el equipo de Scrum siempre entregue valor a las partes interesadas y al negocio, el propietario del producto determina las expectativas del producto, registra los cambios en el producto y administra un backlog de Scrum, una lista de tareas pendientes detallada y constantemente actualizada para el proyecto de Scrum (Peek, 2022).

Equipo De Desarrollo. El equipo Scrum es un grupo autoorganizado de tres a nueve personas que tienen las habilidades comerciales, de diseño, analíticas y de desarrollo para llevar a cabo el trabajo real, resolver problemas y producir productos entregables (Peek, 2022).

Eventos

Esta metodología cuenta con un total de cinco eventos, estos tienen tiempos límites para realizarlos, su duración es fija, es decir no se pueden alargar o acortar.

Sprint. Son tareas con un tiempo fijo y son consecutivas. No son permitidos cambios durante los sprints, aquí el propietario del producto hace mejoras según se requiera.

Sprint Planning. En esta parte se organiza qué se realizará en el sprint y cómo se desarrollará. Estas planeaciones se realizan al inicio de cada sprint, se plantean preguntas como ¿Por qué es valioso este sprint? ¿Qué se puede hacer en este sprint? ¿Cómo se realizará el trabajo elegido?

Daily Scrum. Por medio de estas reuniones diarias se mide el progreso de avance del sprint, estas se realizan diario mientras dura el sprint y se plantean preguntas como ¿Qué hice ayer? ¿Qué voy a hacer hoy? ¿Qué ayuda necesito?

Sprint Review. En estas reuniones se revisa el sprint terminado, el cual debe tener un avance para poder ser presentado al cliente, se muestran las tareas terminadas y faltantes.

Sprint Retrospective. Se hace una revisión del sprint terminado, analizando las cosas buenas y malas, con el fin de hacer mejoras en procesos y mejorar el plan para el siguiente sprint.

2.3. Base de Datos

Una base de datos es una colección organizada de información estructurada, o datos, normalmente almacenados electrónicamente en un sistema informático. Una base de datos generalmente está controlada por un sistema de administración de bases de datos (DBMS) (ORACLE, 2022).

2.3.1. MySQL WorkBench

MySQL WorkBench es una herramienta visual unificada para arquitectos de bases de datos, desarrolladores y administradores de bases de datos. MySQL WorkBench proporciona modelado de datos, desarrollo de SQL y herramientas de administración integrales para la configuración del servidor, la administración de usuarios, la copia de seguridad (MySQL, 2022).

Funciones de MySQL WorkBench

MySQL WorkBench tiene cuatro funciones principales:

Desarrollo. Crea, ejecuta consultas de SQL y administra conexiones con servidores.

Modelado De Datos. Crea modelos de base de datos de forma gráfica, ejecutar ingeniería inversa y directa, editar la base de datos.

Administración Del Servidor. Administra las instancias del servidor MySQL por medio de la administración de usuarios, supervisión del rendimiento del servidor, ejecución de copias de seguridad y recuperación.

Migración De Datos. Migra desde Microsoft SQL Server, PostgreSQL, Microsoft Access, Sybase ASE, SQLite a MySQL.

2.4. Lenguajes de Programación

Los lenguajes de programación son instrucciones que nos permiten establecer una comunicación, con el fin de desarrollar un programa. Todo esto es gracias a algoritmos los cuales son escritos en base a una sintaxis y semántica, que interpreta la computadora para establecer una comunicación entre el usuario y máquina.

2.4.1. Java

El lenguaje de programación java es un lenguaje de propósito general, concurrente, fuertemente tipado, basado en clases y orientado a objetos (ORACLE, 2020).

Java es un lenguaje de programación, es principalmente usado para aplicaciones web, por otro lado, es un lenguaje multiplataforma y orientado a objetos. Algunos otros usos de java son para el desarrollo de videojuegos, macrodatos, inteligencia artificial, computación en la nube e internet de las cosas.

Ediciones de la Plataforma Java

Java cuenta con cuatro ediciones:

Java Standard Edition (Java SE). Esta es la versión más distribuida de Java, diseñada para crear aplicaciones de escritorio con interfaz gráfica de usuario, concurrencia y comunicación por medio de redes.

Java Enterprise Edition (Java EE). Se enfoca al desarrollo de software que se ejecutará en servidores de aplicaciones. A diferencia de Java SE, en esta edición se agrega la persistencia en base de datos, interfaces de usuario web entre otras cosas.

Java Micro Edition (Java ME). Esta edición se enfoca para sistemas con recursos limitados, como dispositivos móviles, dispositivos de domótica entre otros, actualmente es usualmente usado para dispositivos del internet de las cosas.

Java FX. Esta edición tiene un núcleo ligero que lo hace de fácil distribución, aceleración 3D, producción de gráficos y animaciones, por último, integra un mecanismo simplificado para el diseño de interfaces.

2.4.2. XHTML

Las siglas XHTML significan “Extensible Hypertext Markup Language”, en español significa “Lenguaje extensible de marcación de hipertexto”. Es un lenguaje similar a HTML basado en XML, no obstante, tiene una sintaxis más rigurosa y es compatible con todos los navegadores.

Para XHTML las etiquetas `<!DOCTYPE>`, `<html>`, `<head>`, `<title>` y `<body>` son obligatorias agregarlas al documento, por otro lado, los elementos deben estar correctamente anidados, cerrados, estar en minúsculas, sus atributos también deben estar en minúsculas.

Definición de Tipo de Documento

Otro elemento que se debe agregar al inicio de los archivos, es la Definición de Tipo de Documento o DTD, por medio de este especificamos la versión HTML en la que se está trabajando. Existen tres tipos de DTD:

DTD de Transición. Es compatible con los navegadores más antiguos que no tienen compatibilidad con hojas de estilo en cascada incorporadas (GeeksforGeeks, 2022).

Sintaxis: `<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "DTD/xhtml11-transitional.dtd">`

DTD Estricto. La DTD estricta se usa cuando la página XHTML contiene solo lenguaje de marcado. El DTD estricto se usa junto con las hojas de estilo en cascada, porque este atributo no permite la propiedad CSS en la etiqueta del cuerpo (GeeksforGeeks, 2022).

Sintaxis: `<!DOCTYPEhtml PÚBLICO "-//W3C//DTD XHTML 1.0 Strict//EN" "DTD/xhtml11-strict.dtd">`

Frameset DTD. Es el DTD de conjunto de marcos se utiliza cuando la página XHTML contiene marcos (GeeksforGeeks, 2022).

Sintaxis: `<!DOCTYPEhtml PÚBLICO "-//W3C//DTD XHTML 1.0 Strict//EN" "DTD/xhtml11-strict.dtd">`

2.4.3. CSS

Hojas de Estilo en Cascada (del inglés Cascading Style Sheets) o CSS es el lenguaje de estilos utilizado para describir la presentación de documentos HTML o XML (en-US) (incluyendo varios lenguajes basados en XML como SVG, MathML o XHTML) (MDN web docs, 2022).

CSS permite aplicar diseños o estilo a los diferentes elementos de HTML, haciendo nuestra página web más atractiva para el usuario, donde se puede modificar los colores, tamaños y tipografía.

Anatomía de una Regla CSS

Para establecer estilos se tienen que seguir reglas, estas con llamadas “Reglas predeterminadas”, la estructura se compone de 4 elementos:

Selector. Estos son los elementos de HTML, como: p, button, input, entre otros. Puede haber uno o más elementos, para separarlos se tiene que ser por medio de una coma (,).

Declaración. En esta se tiene la propiedad y el valor de la propiedad, pueden ser escritas más de una y deben ser separados por medio del signo punto y coma (;).

Propiedades. Las propiedades ayudarán a cambiar el formato del elemento antes escrito.

Valor de la Propiedad. Este va escrito después de la propiedad, seguido de dos puntos (:), aquí se escribe el valor a cambiar.

2.4.4. AJAX

AJAX significa JavaScript asíncrono y XML (Asynchronous JavaScript and XML). Es un conjunto de técnicas de desarrollo web que permiten que las aplicaciones web funcionen de forma asíncrona, procesando cualquier solicitud al servidor en segundo plano (Gustavo, 2022).

2.4.5. SQL

El lenguaje de consulta estructurado (SQL) es un lenguaje de programación estandarizado que se utiliza para administrar bases de datos relacionales y realizar varias operaciones en los datos que contienen (Loshin, 2022).

SQL nos ayuda a reformar tablas de una base de datos, por otro lado, sirve para agregar, eliminar, actualizar y consultar datos de tablas.

Los comandos SQL se dividen en diferentes tipos:

- Los comandos del lenguaje de definición de datos (DDL)
- Los comandos del lenguaje de manipulación de datos (DML)
- El lenguaje de consulta de datos (DQL)
- Los comandos del lenguaje de control de datos (DCL)
- Los comandos del lenguaje de control de transacciones (TCL)

DDL

Un lenguaje de definición de datos (DDL) es un lenguaje informático que se utiliza para crear y modificar la estructura de los objetos de base de datos en una base de datos. Estos objetos de base de datos incluyen vistas, esquemas, tablas, índices, etc. (Techopedia, 2022).

Los DDL de uso más frecuente son:

Create. Es usado para la creación de nuevas tablas. Usa la siguiente sintaxis:

```
CREATE TABLE [nombre de la tabla] ([definición de columnas]) [parámetros de tabla];
```

Alter. Nos ayuda a realizar modificaciones a tablas ya existentes, puede agregar o eliminar columnas y cambiar tipo de datos. Usa la siguiente sintaxis:

```
ALTER TABLE [nombre de la tabla] [ADD | DROP COLUMN | ALTER COLUMN]]  
[nombre columna tipo columna | nombre columna | nombre columna tipo columna];
```

Drop. Es usado para eliminar tablas de una base de datos, índice o vista. Usa la siguiente sintaxis:

DROP [TABLE | INDEX | VIEW] [nombre de tabla | nombre de índice | nombre de vista];

Truncate. Es usado para eliminar todos los registros de una tabla. Usa la siguiente sintaxis:

```
TRUNCATE TABLE [nombre de tabla];
```

DML

Los comandos SQL que se ocupan de la manipulación de datos presentes en la base de datos pertenecen a DML o Lenguaje de Manipulación de Datos y esto incluye la mayoría de las declaraciones SQL. (GeeksForGeeks, 2022).

Mejora la interacción entre el usuario y el sistema, esto por medio de los comandos:

Select. Con este comando se realizan consultas, recuperar información de una o más tablas. Usa la siguiente sintaxis:

```
SELECT [nombre/s de la columna] FROM [nombre de la tabla] WHERE [condiciones].
```

Update. Con este comando se modifican los datos de uno o más registros. Usa la siguiente sintaxis:

```
UPDATE [nombre de la tabla] SET [nombre de la columna = valor] WHERE [condición].
```

Insert. Con este comando se agregan uno o más registros a una tabla. Usa la siguiente sintaxis:

```
INSERT INTO [nombre de la tabla] [columna(s)] VALUES [valor(es)].
```

Delete. Con este comando se elimina uno o más registros de una tabla. Usa la siguiente sintaxis:

DELETE FROM [*nombre de la tabla*] WHERE [*condiciones*].

2.5. Servidor Web

Los servidores web son un componente de los servidores que tienen como principal función almacenar, en web hosting, todos los archivos propios de una página web y transmitirlos a los usuarios a través de los navegadores mediante el protocolo HTTP (Souza, 2019).

2.5.1. GlassFish

GlassFish es un servidor completo de código abierto para la plataforma Java EE, desarrollado por Sun Microsystems, ahora adquirido por Oracle Corporation. Posibilita ejecutar aplicaciones que siguen esta especificación. Además, de ser un servidor de aplicaciones, también funciona como web server (Genos, 2022).

2.6. Herramientas de Desarrollo

Las herramientas de desarrollador (o "herramientas de desarrollo" o "DevTools" abreviado) son programas que permiten a un desarrollador crear, probar y depurar software (MDN web docs, 2022).

2.6.1. NetBeans

NetBeans IDE es un entorno de desarrollo integrado de código abierto y gratuito para el desarrollo de aplicaciones en los sistemas operativos Windows, Mac, Linux y Solaris. El IDE simplifica el desarrollo de aplicaciones web, empresariales, de escritorio y móviles que utilizan las plataformas Java y HTML5. Además, ofrece soporte para el desarrollo de aplicaciones PHP y C/C++ (ORACLE, 2022).

2.6.2. Google Chrome

Google Chrome es un navegador web gratuito desarrollado por Google. Se basa en el proyecto de código abierto Chromium. Algunas diferencias clave se describen en BrowserStack. Chrome admite su propio diseño llamado Blink (MDN web docs, 2022).

2.7. Bibliotecas

Una librería es un conjunto de archivos que se utiliza para desarrollar software. Suele estar compuesta de código y datos, y su fin es ser utilizada por otros programas de forma totalmente autónoma (devCamp, 2020).

2.7.1. IText

IText es una librería de código abierto, esta permite crear y revisar archivos de tipo PDF, HTML y RFT en Java.

Por medio de esta librería es posible añadir al documento características como: márgenes, tablas, cabeceras, pies de páginas, numeración, diferentes tipografías, imágenes, marcas de agua y salto de página.

A diferencia con otras librerías, IText no hace la transformación XSLT ya que esta clase ya está ejecutada, por ende, el sistema consume menos recursos y su generación es más rápida.

2.7.2. PrimeFaces

PrimeFaces es una librería ligera que no depende de otras librerías de terceros y no necesita configuraciones, se distribuye en un único JAR y nos proporciona una serie de utilidades y componentes para usar con JavaServer Faces (JSF) (Yáñez, 2022).

PrimeFaces es una librería de interfaz de usuario para aplicaciones Java web de aplicaciones empresariales Java EE. Algunas de las características son que es ligera, esto se debe a que se encuentra encapsulada en solo un JAR y no se requiere de otra

instalación, por otro lado, sus componentes son desarrollados con el propósito de esconder la complejidad manteniendo su flexibilidad.

2.8. Herramientas de Diseño

Las herramientas de diseño son programas que nos permite crear diseños por medio de un ordenador.

2.8.1. Balsamiq Mockups

Balsamiq Mockups es un programa que facilita la creación de prototipos o bocetos de manera digital, con el propósito de tener una referencia de la interfaz que se desarrollará. En esta se plasman las ideas y elementos que contendrá dicha ventana, permitiendo los cambios y mejoras de forma inmediata, a este proceso se le llama wireframing.

Por medio de esta herramienta se pueden realizar cambios fácilmente, ya que la creación de las interfaces tan solo es necesario arrastrar o eliminar el elemento que se requiera. También se pueden duplicar interfaces ya creadas, esto en caso de que requiera la misma ventana y solo se requieran hacer ligeros cambios en la interfaz.

Esta herramienta nos proporciona diversos objetos prediseñados para agregar los elementos según se requieran, tanto para dispositivos móviles como diseño web. Por otro lado, se permite compartir el trabajo y trabajar en línea, a fin de recibir retroalimentaciones o revisiones, ver los trabajos en tiempo real, trabajar desde diferentes dispositivos e incluso permite cambiar el modo de vista para visualizar el trabajo en modo de boceto o una apariencia final.

2.9. Java Persistence API

La persistencia de datos es un medio mediante el cual una aplicación puede recuperar información desde un sistema de almacenamiento no volátil y hacer que esta persista. La persistencia de datos es vital en las aplicaciones empresariales debido al acceso necesario a las bases de datos relacionales. Las aplicaciones desarrolladas para este entorno deben gestionar por su cuenta la persistencia o

utilizar soluciones de terceros para manejar las actualizaciones y recuperaciones de las bases de datos con persistencia (IBM, 2022).

JPA es el encargado de hacer una conversión entre los objetos de Java e instrucciones para el manejador de base de datos (MDB). JPA está diseñado para funcionar dentro y fuera de Java Enterprise Edition (Java EE).

La persistencia de datos de Java abarca:

API EntityManager. Se encarga de eliminar, actualizar, recuperar y aplicar la persistencia de datos en una base de datos.

API EntityManager Y Metadatos Objeto/Relacional. Se encarga de manejar las operaciones de la base de datos sin escribir el código SQL con el fin de mantener la persistencia de datos.

Lenguaje De Consulta O Java Persistence Query Lenguaje (JPQL). Se encarga de recuperar objetos sin grabar las consultas realizadas en la base de datos en la que se está trabajando.

2.9.1. Entity

Una clase que debe persistir en una base de datos debe anotarse con `javax.persistence.Entity`. Tal clase se llama Entity. JPA utiliza una tabla de base de datos para cada entidad. Las instancias persistentes de la clase se representarán como una fila en la tabla (Vogel, 2021).

2.9.2. Persistencia de Campos

Los campos de la Entidad serán guardados en la base de datos. JPA puede usar sus variables de instancia (campos) o los getters y setters correspondientes para acceder a los campos (Vogel, 2021).

2.9.3. Mapeo de Relaciones

JPA permite definir relaciones entre clases, por ejemplo, se puede definir que una clase es parte de otra clase (contención). Las clases pueden tener relaciones de uno a uno, de uno a muchos, de muchos a uno y de muchos a muchos con otras clases (Vogel, 2021).

Anotaciones de relación:

- @OneToOne
- @OneToMany
- @ManyToOne
- @ManyToMany

2.9.4. Entity Manager

El administrador de entidades `javax.persistence.EntityManager` proporciona las operaciones desde y hacia la base de datos, por ejemplo, buscar objetos, conservarlos, eliminar objetos de la base de datos, etc. (Vogel, 2021).

2.9.5. Unidades de Persistencia

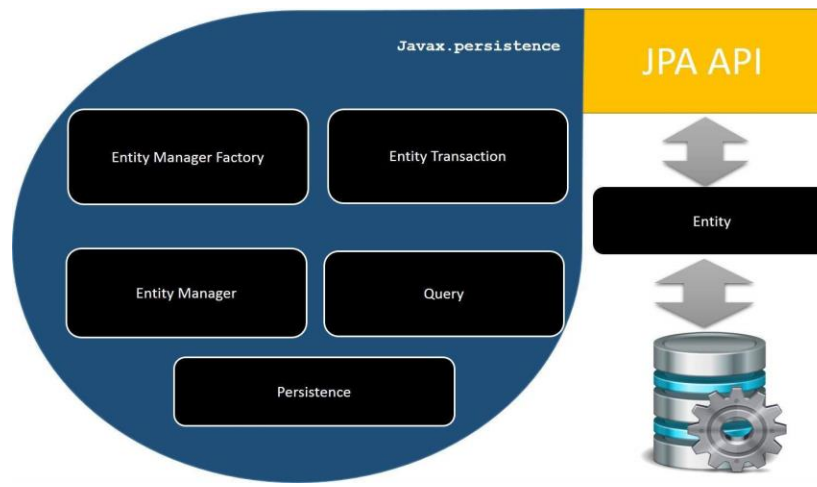
El `EntityManager` es creado por el `EntityManagerFactory` que está configurado por la unidad de persistencia. La unidad de persistencia se describe a través del `persistence.xml` archivo en el META-INF directorio de la carpeta de origen (Vogel, 2021).

2.9.6. Arquitectura de JPA

La arquitectura de JPA está representada en la figura 8, en la cual se observa como interactúa el Entity Manager Factory, Entity Transaction, Entity Manager, Query, Persistence y Entity, los cuales se explicará la función que desarrolla cada uno de estos elementos.

Figura 1

Arquitectura de JPA



Nota. Reproducido de “JPA - Arquitectura”, por Tutorialspoint, 2022, (https://www.tutorialspoint.com/es/jpa/jpa_architecture.htm). Todos los derechos reservados 2022 por Licenciatarario. Reproducido con permiso del autor.

EntityManagerFactory

Esta es una clase de fábrica de EntityManager. Crea y gestiona múltiples instancias EntityManager (Tutorialspoint, 2022).

EntityManager

Es una interfaz, que gestiona la persistencia de objetos. Funciona como instancia de consulta (Tutorialspoint, 2022).

Entidad

Las entidades son los objetos de persistencia, tiendan como registros en la base de datos (Tutorialspoint, 2022).

EntityTransaction

Tiene una relación de uno a uno con EntityManager. Para cada método EntityManager, se mantienen las operaciones de EntityTransaction clase (Tutorialspoint, 2022).

Persistencia

Esta clase contiene métodos estáticos para obtener EntityManagerFactory (Tutorialspoint, 2022).

Consulta

Esta interfaz es implementada por cada proveedor JPA relacional para obtener objetos que cumplan los criterios (Tutorialspoint, 2022).

CAPÍTULO III METODOLOGÍA

3.1. Participantes y/o Representación Estadística

De acuerdo al INEGI (2021) en su última encuesta realizada en el año 2021, se encontró que en el estado de Tlaxcala se han registrado 544,563 vehículos de motor y dentro del municipio de Tlaxcala de Xicohténcatl se han registrado 121,678 vehículos de motor.

De acuerdo al INEGI (2020) en el año 2020, se registró que en el estado de Tlaxcala se tiene una población total de 1,342,977 y en el municipio de Tlaxcala de Xicohténcatl se cuenta con un total de 99,896 habitantes.

En base a la anterior información, se podría decir que de cada 1.218 ciudadanos del municipio de Tlaxcala hay 1 vehículo registrado con motor.

En esta investigación, se enfocará en los conductores, revisores e inmovilizadores del municipio de Tlaxcala de Xicohténcatl, por consiguiente, se tendrá que seleccionar un grupo de muestra para la proporción de información respecto al uso del sistema.

En la tabla 1, se muestran los grupos y la cantidad de personas a realizar encuestas. Dentro de la selección se conductores se tomaron en cuenta usuarios de las diferentes calles del municipio de Tlaxcala de Xicohténcatl, esto con el propósito de tener diferentes respuestas respecto a la funcionalidad y experiencia de estos. Por otro lado, los inmovilizadores y revisores son una parte de las personas trabajando actualmente en los parquímetros del municipio.

Tabla 1

Muestra para encuesta

Usuario	Cantidad
Conductores	50
Inmovilizadores	10
Revisores	10
Total	70

3.2. Fuentes de Información

Una fuente de información nos ayuda a satisfacer una necesidad informativa, además de facilitar la identificación y ubicación de información.

3.2.1. Tlaxcala Digital

Tlaxcala digital es un periódico digital del estado de Tlaxcala, el cual abarca noticias tanto nacionales como municipales. Está conformado por analistas, profesionales de la comunicación destacados del estado.

De acuerdo a Montiel (2020), en el año 2020 los parquímetros del estado de Tlaxcala habían suspendido sus servicios en forma de apoyo a la economía de los ciudadanos por la pandemia del Covid-19, sin embargo, a partir del 14 de julio de 2020 volvieron a funcionar con normalidad.

De esta forma se reactivó la movilidad vehicular, de igual forma se aplicaron nuevamente el reglamento de Parquímetros del municipio de Tlaxcala en las calles principales de este.

3.2.2. Catálogo Estatal de Trámites y Servicios del Gobierno de Tlaxcala

Por medio de este catálogo se pueden realizar y consultar los diferentes trámites y servicios del estado, son divididos en diferentes categorías para su fácil navegación, tal como educación, deporte, financieros, medio ambiente, transporte entre otros.

En la sección de transporte se pueden encontrar los trámites y servicios que respectan a este, así mismo se encuentra su dependencia a cargo, horarios de atención y en caso de que el trámite se pueda realizar en línea se mostrará.

Entre los trámites de transporte se encuentra el de expedición de certificado de no infracción de placa o tarjeta de circulación, el cual ayudará a comprobar a los ciudadanos si tienen algún tipo de adeudo de infracción.

3.2.3. Instituto Nacional de Estadística y Geografía (INEGI)

El INEGI se encarga de realizar censos en todos los estados, con el fin de recabar y difundir información de México respecto a sus recursos, economía, población y territorio.

Dentro de estas encuestas se puede consultar el total de personas residentes en un estado y en un municipio en específico, por otro lado, se pueden encontrar resultados de vehículos registrados dentro de todo el estado y por municipios.

3.3. Técnicas e Instrumentos

Según Rus (2020), hay un total de 6 tipos de investigaciones y a su vez dentro de estas se subdividen. Los tipos de investigaciones son por el objetivo perseguido, nivel de profundización, tipo de inferencia, por la forma de manipular las variables, tipo de dato y por el periodo de tiempo.

En este caso se centrará en una investigación por tipo de dato, esta se subclasifica en: cualitativa, cuantitativa y mixta. La principal característica de la primera es que sus datos no son cuantificables, tienen en parte una subjetividad, por otro lado, se recopila datos de grabaciones, bibliografías y documentos. Por otro lado, la cuantitativa sus datos son medibles, se logran hacer inferencias y análisis en base a la información obtenida.

Una investigación cuantitativa tiene un enfoque matemático y estadístico, por medio de estas formas se realiza un análisis de la información obtenida. Los datos son obtenidos por medio de cuestionarios o encuestas y esta a su vez es presentada en tablas o gráficas en forma estadística, con el fin de encontrar algún comportamiento o relación.

Esta al ser una investigación de tipo cuantitativa, se dispondrá de una encuesta para recabar la información del sistema de parquímetros actual. En esta encuesta se analizarán los tiempos tomados para realizar cada tarea y funciones, así mismo como la experiencia del usuario dentro del sistema.

La encuesta será de tipo opción múltiple con diferentes opciones de tiempos según la pregunta, esto con el fin de medir la funcionalidad. Con respecto a la experiencia del

usuario, se utilizará el método Likert, el cual consiste en cinco respuestas en forma de escala, que van desde “Totalmente de acuerdo” a “Totalmente en desacuerdo”.

Los tipos de usuarios a realizar la encuesta son: conductor, revisor e inmovilizador, ya que en base a sus respuestas se podrá ejecutar un análisis y a su vez realizar una comparación, con el fin de verificar el cumplimiento de la hipótesis planeada.

3.3.1. Conductor

Para conocer los tiempos de funcionalidad del sistema respecto al conductor se realizará una encuesta en la cual se evalué el tiempo de compra, recarga de tiempo, multas, retiro de arañas.

- ¿Qué tiempo le toma la compra de tiempo para el parquímetro?
- ¿Qué tiempo le toma la recarga de tiempo para el parquímetro?
- En caso de ser acreedor a una multa, ¿Qué tiempo le toma la paga de esta?
- En caso de tener la araña inmovilizadora en su vehículo, ¿Qué tiempo se toma el personal para retirarlo?

Respecto a la experiencia del conductor, en las siguientes preguntas se empleará el método Likert, donde se tiene una escala que va desde “Totalmente de acuerdo” a “Totalmente en desacuerdo”.

- ¿Cómo calificaría la facilidad de uso del sistema de parquímetro?
- ¿Cree que el proceso de compra de tiempo es intuitivo?
- ¿Se siente informado adecuadamente sobre las políticas y reglas del uso del parquímetro?
- ¿Ha experimentado algún error o problema técnico al utilizar el sistema de parquímetro?

3.3.2. Inmovilizador

En los tiempos del inmovilizador es imprescindible conocer la verificación de multas, ya que en base a eso se realizará el retiro o no de la araña, que a su vez puede ser una buena o mala experiencia para el conductor dependiendo el tiempo de esta tarea.

- ¿Qué tiempo le toma verificar el pago de una multa?
- ¿Qué tiempo le toma el retiro de una araña inmovilizadora?

Respecto a la experiencia del usuario inmovilizador, se utilizará el método de Likert, donde se evalúa su perspectiva del sistema.

- ¿Cree que es eficiente el sistema de solicitud de inmovilizaciones?
- ¿Ha experimentado algún error o problema técnico al utilizar el sistema de parquímetro?

3.3.3. Revisor

El revisor tiene que tener buen tiempo de respuesta a la hora de realizar sus consultas, a su vez el tiempo de la solicitud y aceptación de la inmovilización ya que se forma una cadena hasta llegar al conductor, este termina siendo afectado cuando estos procesos llevan mucho tiempo.

- ¿Cuál es el tiempo de respuesta al ingresar la placa del vehículo para su revisión?
- ¿Qué tiempo le toma el solicitar la inmovilización de un vehículo?
- ¿Qué tiempo se toma el inmovilizador en aceptar su solicitud?

Finalmente, se medirá la experiencia del revisor a la hora de usar el sistema por medio del método Likert.

- ¿Cree que es eficiente el sistema de solicitud de inmovilizaciones?

- ¿Ha experimentado algún error o problema técnico al utilizar el sistema de parquímetro?

3.4. Procedimientos

A continuación, se plasmarán los resultados de las encuestas antes planteadas. Como primer usuario tenemos al conductor, se le realizaron un total de 6 preguntas en las cuales se busca analizar los tiempos que ocupa para cada función y su experiencia realizándolas.

De la figura 2 a la 5 se presentan los resultados respecto a los tiempos de las funcionalidades del conductor y de la 6 a la 9 se muestran de acuerdo a su experiencia. Gracias a estos resultados se podrán realizar comparaciones de tiempo y observar si la hipótesis planteada se cumple o no.

Figura 2

¿Qué tiempo le toma la compra de tiempo para el parquímetro?



Figura 3

¿Qué tiempo le toma la recarga de tiempo para el parquímetro?



Figura 4

En caso de ser acreedor a una multa, ¿Qué tiempo le toma la paga de esta?



Figura 5

En caso de tener la araña inmovilizadora en su vehículo, ¿Qué tiempo se toma el personal para retirarlo?



Figura 6

¿Cómo calificaría la facilidad de uso del sistema de parquímetro? Donde 1 es la puntuación mínima y 5 la máxima.

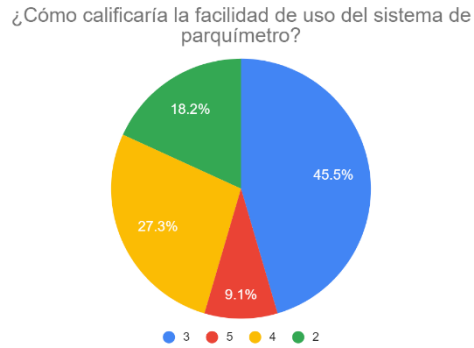


Figura 7

¿Cree que el proceso de compra de tiempo es intuitivo?



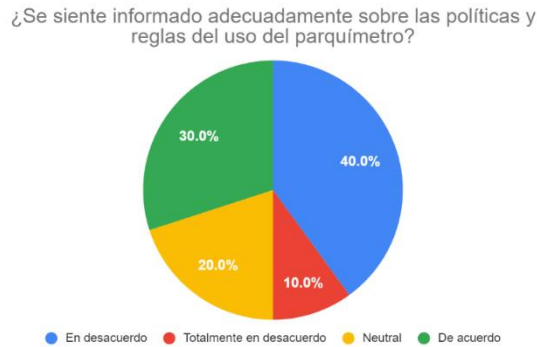
Figura 8

¿Ha experimentado algún error o problema técnico al utilizar el sistema de parquímetro?



Figura 9

¿Se siente informado adecuadamente sobre las políticas y reglas del uso del parquímetro?



Por otro lado, se presentarán los resultados de las encuestas realizadas a los revisores. En esta encuesta se realizaron un total de 5 preguntas, en las cuales se contemplan los tiempos de tareas y la experiencia del usuario dentro del sistema.

De la figura 10 a la 12 se presentan las respuestas que nos ayudarán a analizar los tiempos del sistema, por otro lado, la figura 13 y 14 se contempla la experiencia del usuario dentro del sistema.

Figura 10

¿Cuál es el tiempo de respuesta del sistema al ingresar la placa del vehículo para su revisión?



Figura 11

¿Qué tiempo le toma el solicitar la inmovilización de un vehículo?

¿Qué tiempo le toma el solicitar la inmovilización de un vehículo?

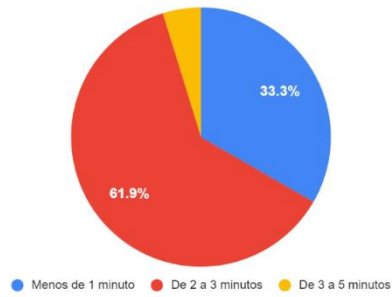


Figura 12

¿Qué tiempo se toma el inmovilizador en aceptar su solicitud?

¿Qué tiempo se toma el inmovilizador en aceptar su solicitud?

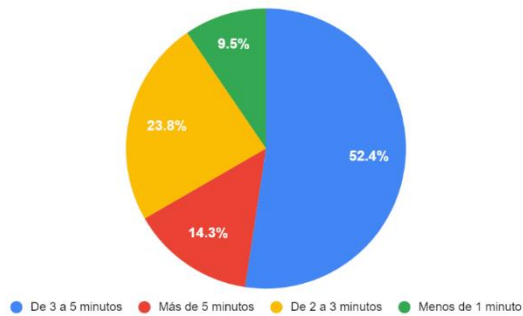


Figura 13

¿Cree que es eficiente el sistema de solicitud de inmovilizaciones?

¿Cree que es eficiente el sistema de solicitud de inmovilizaciones?

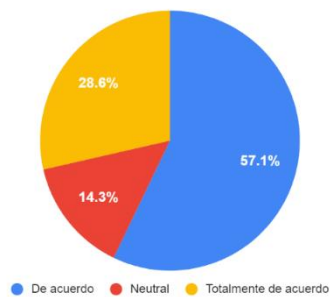


Figura 14

¿Ha experimentado algún error o problema técnico al utilizar el sistema de parquímetro?



Finalmente, se mostrarán los resultados de la encuesta realizada a los inmovilizadores, este cuenta con un total de 4 preguntas, dentro de sus funcionalidades es la verificación de pago de multas y el retiro de la araña inmovilizadora.

Las figuras 15 y 16 se recaban los tiempos para cada tarea realizada por el inmovilizador, por otro lado, las figuras 17 y 18 presentan la experiencia del usuario respecto a este usuario.

Figura 15

¿Qué tiempo le toma verificar el pago de una multa?



Figura 16

¿Qué tiempo le toma el retiro de una araña inmovilizadora?

¿Qué tiempo le toma el retiro de una araña inmovilizadora?

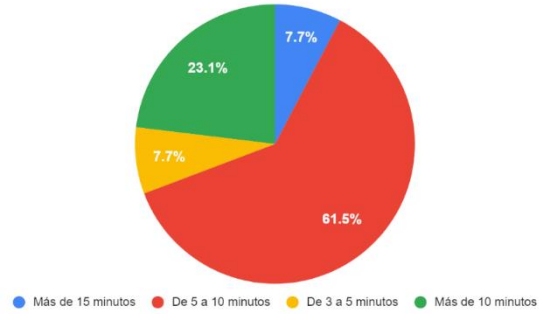


Figura 17

¿Cree que es eficiente el sistema de solicitud de inmovilizaciones?

¿Cree que es eficiente el sistema de solicitud de inmovilizaciones?

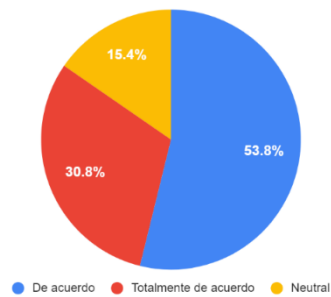
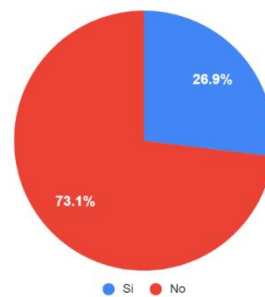


Figura 18

¿Ha experimentado algún error o problema técnico al utilizar el sistema de parquímetro?

¿Ha experimentado algún error o problema técnico al utilizar el sistema de parquímetro?



3.5. Aplicación de Metodología Scrum

El primer paso en la metodología Scrum que debe realizarse es la selección de requerimientos conforme a un listado. Dividiéndose en requerimientos funcionales en la tabla 2 y no funcionales en la tabla 3.

Tabla 2

Requerimientos funcionales

Requerimientos funcionales			
No.	Clave	Nombre	Fecha
1	RF001	Inicio de sesiones	18/07/2022
2	RF002	Registro de usuarios	18/07/2022
3	RF003	Registro de vehículos	18/07/2022
4	RF004	Venta de saldo	18/07/2022
5	RF005	Visualización de la ubicación del vehículo	18/07/2022
6	RF006	Generación de reportes	18/07/2022
7	RF007	Búsqueda de vehículos	19/07/2022
8	RF008	Registro de calles y cajones	19/07/2022
9	RF009	Registro para costos del sistema	19/07/2022
10	RF010	Pago de multas	19/07/2022
11	RF011	Pago de tiempo	19/07/2022

Tabla 3

Requerimientos no funcionales

Requerimientos no funcionales			
No.	Clave	Nombre	Fecha
1	RNF001	Información de encabezado	19/07/2022
2	RNF002	Logotipos encabezados	19/07/2022

3	RNF003	Paleta de colores	19/07/2022
4	RNF004	Diseño de pantallas de baja fidelidad	19/07/2022
5	RNF005	Diseño de pantallas de alta fidelidad	19/07/2022
6	RNF006	Formato de descargas Excel y PDF	19/07/2022

3.6. Planificación de Iteraciones

En la primera etapa se realiza una lista de tareas (Sprint Backlog) para el desarrollo del sistema, que se presentan en la tabla 4.

Tabla 4

Product Backlog

Sprint Backlog					
Actividades	Descripción del requerimiento	Tareas	Responsable	Fecha	Sprint
1	Instalación de herramientas de diseño y desarrollo	Instalar navegador Google Chrome	Mitzi Sollange Hernández Juárez	18/07/2022	1
		Instalar Apache NetBeans 8.2	Mitzi Sollange Hernández Juárez	18/07/2022	1
		Instalar GlassFish 4.1	Mitzi Sollange Hernández Juárez	18/07/2022	1
		Instalar software Manejador de Base de datos SQL WorkBench	Mitzi Sollange Hernández Juárez	19/07/2022	1
		Instalar Balsamiq	Mitzi Sollange Hernández Juárez	19/07/2022	1
2	Diseño de prototipos de baja fidelidad	Diseñar pantalla de baja fidelidad módulo Inicio de sesión	Mitzi Sollange Hernández Juárez	20/07/2022	1
		Diseñar pantalla de baja fidelidad módulo registro	Mitzi Sollange Hernández Juárez	21/07/2022	1

Sprint Backlog					
Actividades	Descripción del requerimiento	Tareas	Responsable	Fecha	Sprint
		Diseñar pantalla de baja fidelidad catálogo conductor	Mitzi Sollange Hernández Juárez	22/07/2022	1
		Diseñar pantalla de baja fidelidad catálogo revisor	Mitzi Sollange Hernández Juárez	25/07/2022	1
		Diseñar pantalla de baja fidelidad catálogo inmovilizador	Mitzi Sollange Hernández Juárez	26/07/2022	1
		Diseñar pantalla de baja fidelidad catálogo vendedor	Mitzi Sollange Hernández Juárez	27/07/2022	1
		Diseñar pantalla de baja fidelidad catálogo administrador	Mitzi Sollange Hernández Juárez	28/07/2022	1
3	Establecer estructura de pantallas de alta fidelidad con XHTML, CSS Y Librería PrimeFaces	Diseñar pantalla de alta fidelidad módulo Inicio de sesión	Mitzi Sollange Hernández Juárez	01/08/2022	2
		Diseñar pantalla de alta fidelidad módulo registro	Mitzi Sollange Hernández Juárez	03/08/2022	2
		Diseñar pantalla de alta fidelidad catálogo de conductor	Mitzi Sollange Hernández Juárez	05/08/2022	2
		Diseñar pantalla de alta fidelidad catálogo de revisor	Mitzi Sollange Hernández Juárez	09/08/2022	2
		Diseñar pantalla de alta fidelidad catálogo de inmovilizador	Mitzi Sollange Hernández Juárez	11/08/2022	2
		Diseñar pantalla de alta fidelidad catálogo de vendedor	Mitzi Sollange Hernández Juárez	15/08/2022	2
		Diseñar pantalla de alta fidelidad catálogo de administrador	Mitzi Sollange Hernández Juárez	17/08/2022	2

Sprint Backlog					
Actividades	Descripción del requerimiento	Tareas	Responsable	Fecha	Sprint
4	Implementar funcionalidad a pantallas utilizando Java	Crear View y Controller para Catálogo de Cajones	Mitzi Sollange Hernández Juárez	22/08/2022	2
		Crear View y Controller para Catálogo de calles	Mitzi Sollange Hernández Juárez	24/08/2022	2
		Crear View y Controller para Catálogo de usuarios	Mitzi Sollange Hernández Juárez	26/08/2022	2
		Crear View y Controller para Catálogo de costo tiempos	Mitzi Sollange Hernández Juárez	30/08/2022	2
		Crear View y Controller para Catálogo de inmobilizaciones	Mitzi Sollange Hernández Juárez	01/09/2022	2
		Crear View y Controller para Catálogo de revisiones	Mitzi Sollange Hernández Juárez	05/09/2022	2
		Crear View y Controller para Catálogo de sesión	Mitzi Sollange Hernández Juárez	07/09/2022	2
		Crear View y Controller para Catálogo de venta de saldo	Mitzi Sollange Hernández Juárez	09/09/2022	2
5	Desarrollo Back - End con Java	Crear Script SQL para la base de datos	Mitzi Sollange Hernández Juárez	14/09/2022	3
		Crear Script de conexión Back - End con Front - End	Mitzi Sollange Hernández Juárez	16/09/2022	3
		Crear Entity, Dao y Business para operación CRU en Catálogo de Cajones	Mitzi Sollange Hernández Juárez	20/09/2022	3
		Crear Entity, Dao y Business para operación CRU en Catálogo de calles	Mitzi Sollange Hernández Juárez	22/09/2022	3

Sprint Backlog					
Actividades	Descripción del requerimiento	Tareas	Responsable	Fecha	Sprint
		Crear Entity, Dao y Business para operación CRUD en Catálogo de usuarios	Mitzi Sollange Hernández Juárez	26/09/2022	3
		Crear Entity, Dao y Business para operación CRUD en Catálogo de costos tiempo	Mitzi Sollange Hernández Juárez	28/09/2022	3
		Crear Entity, Dao y Business para operación CR en Catálogo de venta saldo	Mitzi Sollange Hernández Juárez	30/09/2022	3
		Crear Entity, Dao y Business para operación CR en Catálogo de revisiones	Mitzi Sollange Hernández Juárez	04/10/2022	3
		Crear Entity, Dao y Business para Catálogo de sesión	Mitzi Sollange Hernández Juárez	06/10/2022	3
		Crear Entity, Dao y Business para operación RU en parámetros de configuración	Mitzi Sollange Hernández Juárez	11/10/2022	3
		Crear Entity, Dao y Business para operación R en catálogo de saldo utilizado	Mitzi Sollange Hernández Juárez	13/10/2022	3
		Crear Entity, Dao y Business para operación CRUD en Catálogo de vehículos	Mitzi Sollange Hernández Juárez	18/10/2022	3
		Implementación de generación de reportes en Excel y PDF	Mitzi Sollange Hernández Juárez	18/10/2022	3
6	Pruebas funcionales del sistema web de prueba	Prueba de Módulo inicio de sesión	Mitzi Sollange Hernández Juárez	20/10/2022	4
		Prueba permisos para tipos de usuarios	Mitzi Sollange Hernández Juárez	25/10/2022	4
		Operación CRU en catálogo de cajones	Mitzi Sollange Hernández Juárez	27/10/2022	4

Sprint Backlog					
Actividades	Descripción del requerimiento	Tareas	Responsable	Fecha	Sprint
		Operación CRU en catálogo de catálogo de calles	Mitzi Sollange Hernández Juárez	27/10/2022	4
		Operación CRUD en catálogo de usuarios	Mitzi Sollange Hernández Juárez	28/10/2022	4
		Operación CRUD en catálogo de costos tiempo	Mitzi Sollange Hernández Juárez	28/10/2022	4
		Operación CR en catálogo de venta saldo	Mitzi Sollange Hernández Juárez	31/10/2022	4
		Operación CR en catálogo de revisiones	Mitzi Sollange Hernández Juárez	31/10/2022	4
		Operación RU en catálogo de parámetros de configuración	Mitzi Sollange Hernández Juárez	01/11/2022	4
		Operación R en catálogo de catálogo de saldo utilizado	Mitzi Sollange Hernández Juárez	01/11/2022	4
		Operación CRUD en catálogo de catálogo de vehículos	Mitzi Sollange Hernández Juárez	02/11/2022	4
		Prueba módulo de generación de reportes	Mitzi Sollange Hernández Juárez	02/11/2022	4
		Prueba descargas Excel y PDF	Mitzi Sollange Hernández Juárez	03/11/2022	4
7	Mejoras al sistema web de prueba	Mensajes de advertencia y confirmación	Mitzi Sollange Hernández Juárez	04/11/2022	4
		Formato de reportes institucionales	Mitzi Sollange Hernández Juárez	08/11/2022	4

Sprint Backlog					
Actividades	Descripción del requerimiento	Tareas	Responsable	Fecha	Sprint
		Formato de descargas Excel y PDF	Mitzi Sollange Hernández Juárez	10/11/2022	4

3.7. Ejecución Sprint 1

En la primera iteración se realizaron las primeras 12 actividades, como inicio para el desarrollo del sistema. Si no se cumplen en su totalidad las actividades en el primer sprint planeado se considerarán en el siguiente sprint.

En la tabla 5 se presentan las tareas del sprint 1 junto con la fecha programada, estado del desarrollo y el tiempo que se requiere.

Tabla 5

Sprint 1

Sprint 1						
No.	Tarea	Responsable	Fecha	Sprint	Estado	Horas
1	Instalar navegador Google Chrome	Mitzi Sollange Hernández Juárez	18/07/2022	1	Completado	1
2	Instalar Apache NetBeans 8.2	Mitzi Sollange Hernández Juárez	18/07/2022	1	Completado	1
3	Instalar GlassFish 4.1	Mitzi Sollange Hernández Juárez	18/07/2022	1	Completado	1
4	Instalar software Manejador de Base de datos SQL WorkBench	Mitzi Sollange Hernández Juárez	19/07/2022	1	Completado	1
5	Instalar Balsamiq	Mitzi Sollange Hernández Juárez	19/07/2022	1	Completado	1

Sprint 1						
No.	Tarea	Responsable	Fecha	Sprint	Estado	Horas
6	Diseñar pantalla de baja fidelidad módulo Inicio de sesión	Mitzi Sollange Hernández Juárez	20/07/2022	1	Completado	1
7	Diseñar pantalla de baja fidelidad módulo registro	Mitzi Sollange Hernández Juárez	21/07/2022	1	Completado	1
8	Diseñar pantalla de baja fidelidad catálogo conductor	Mitzi Sollange Hernández Juárez	22/07/2022	1	Completado	1
9	Diseñar pantalla de baja fidelidad catálogo revisor	Mitzi Sollange Hernández Juárez	25/07/2022	1	Completado	1
10	Diseñar pantalla de baja fidelidad catálogo inmovilizador	Mitzi Sollange Hernández Juárez	26/07/2022	1	Completado	1
11	Diseñar pantalla de baja fidelidad catálogo vendedor	Mitzi Sollange Hernández Juárez	27/07/2022	1	Completado	1
12	Diseñar pantalla de baja fidelidad catálogo administrador	Mitzi Sollange Hernández Juárez	28/07/2022	1	Completado	1

3.8. Resultados Obtenidos del Sprint 1

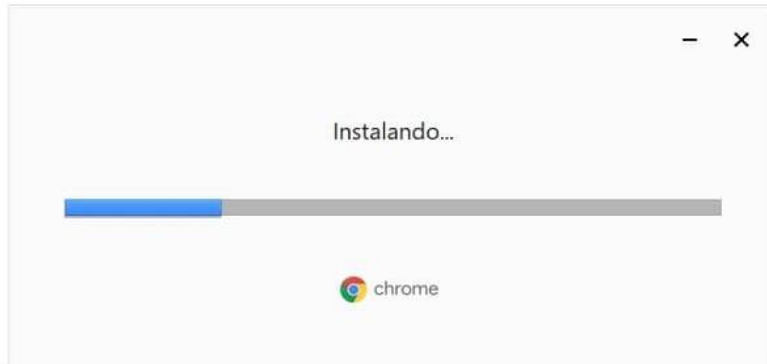
A continuación, se muestran los resultados obtenidos en el sprint 1, los cuales se describieron anteriormente en la tabla 5.

3.8.1. Instalar Navegador

Se descargó e instaló el navegador Google Chrome, dónde se visualizará el funcionamiento del sistema web.

Figura 19

Instalación de navegador Google Chrome

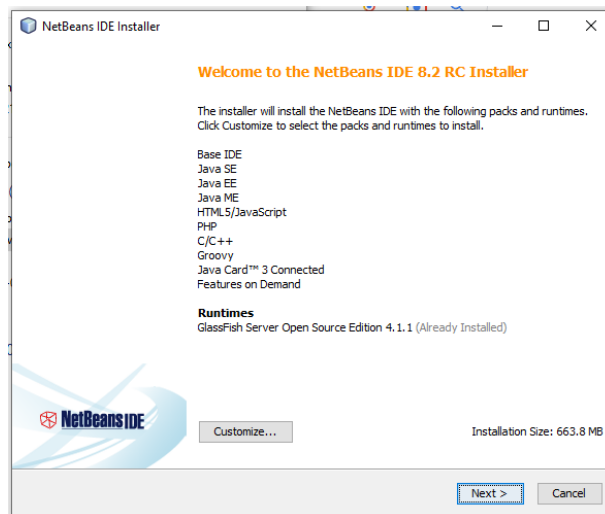


3.8.2. Instalar Apache NetBeans 8.2

Esta herramienta se instaló con el fin de editar los ficheros del proyecto. El programa fue descargado desde su página oficial de internet.

Figura 20

Instalación de Apache NetBeans 8.2

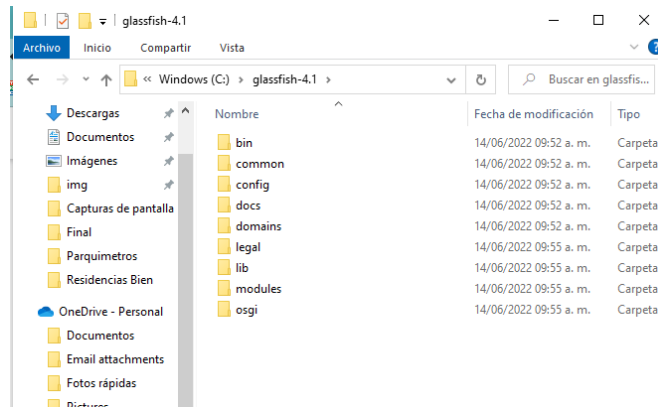


3.8.3. Instalar GlassFish 4.1

Es un servidor de aplicaciones de código abierto, el cual puede ser utilizado como servidor web, además de que puede gestionar aplicaciones Java EE.

Figura 21

Instalación de GlassFish 4.1

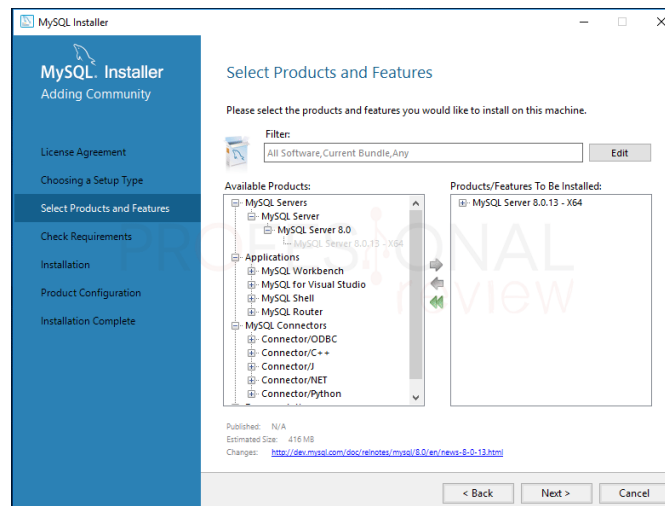


3.8.4. Instalar Software Manejador de Base de Datos MySQL WorkBench

Para su instalación y configuración de toda la paquetería de MySQL, fue necesario descargarlo de su sitio oficial de internet.

Figura 22

Instalación de MySQL WorkBench

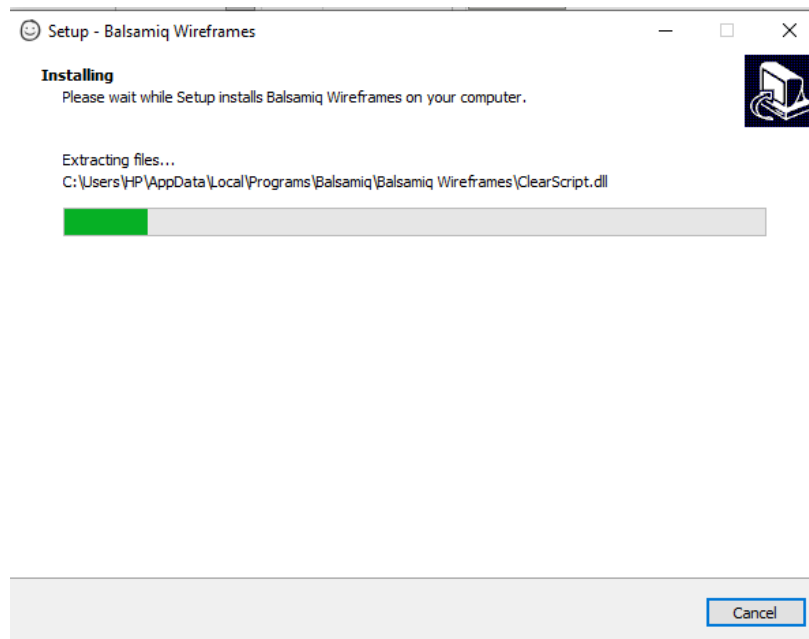


3.8.5. Instalar Balsamiq

Esta herramienta servirá para desarrollar las diferentes pantallas de baja fidelidad del presente proyecto, siendo una herramienta fácil para el boceto de la interfaz del sistema.

Figura 23

Instalación de Balsamiq



3.8.6. Diseñar Pantalla de Baja Fidelidad Módulo Inicio de Sesión

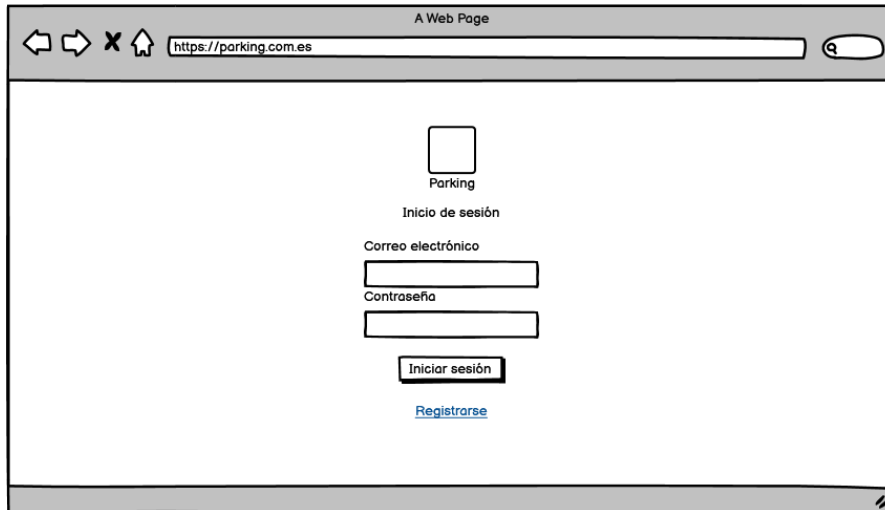
El sistema web contará con un apartado de autenticación, donde el administrador, consultor, revisor, inmovilizador y vendedor podrán acceder a su respectivo módulo.

Para poder autenticarse harán uso de la información que ingresaron previamente en su registro, siendo el correo electrónico y contraseña respectiva.

En caso de no contar con una cuenta, en la parte inferior se encuentra un enlace que redireccionará a un formulario que permitirá su registro.

Figura 24

Pantalla de baja fidelidad inicio de sesión

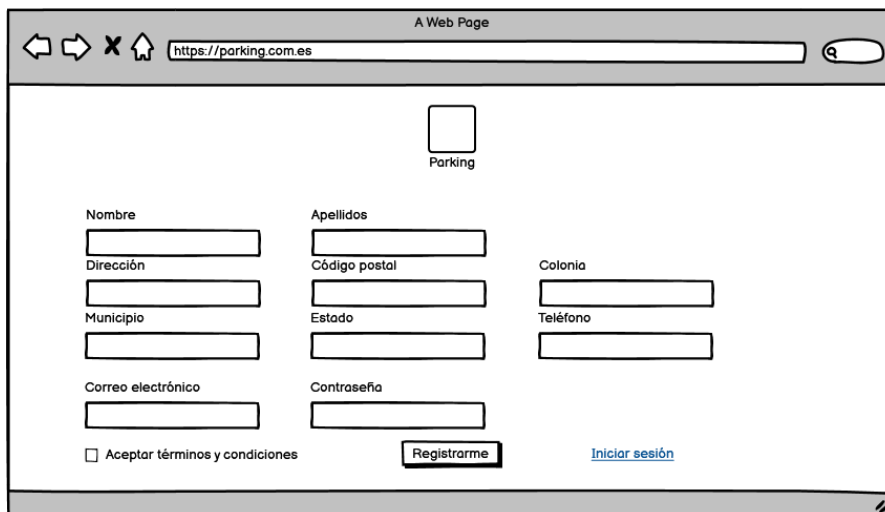


3.8.7. Diseñar Pantalla de Baja Fidelidad Módulo Registro

En la sección de registro se pedirán los datos: nombre, apellidos, dirección, código postal, colonia, municipio, estado, teléfono, correo y contraseña, al final del formulario el usuario deberá aceptar los términos y condiciones para permitir finalizar el registro.

Figura 25

Pantalla de baja fidelidad registro de conductor



3.8.8. Diseñar Pantallas de Baja Fidelidad Catálogo Conductor

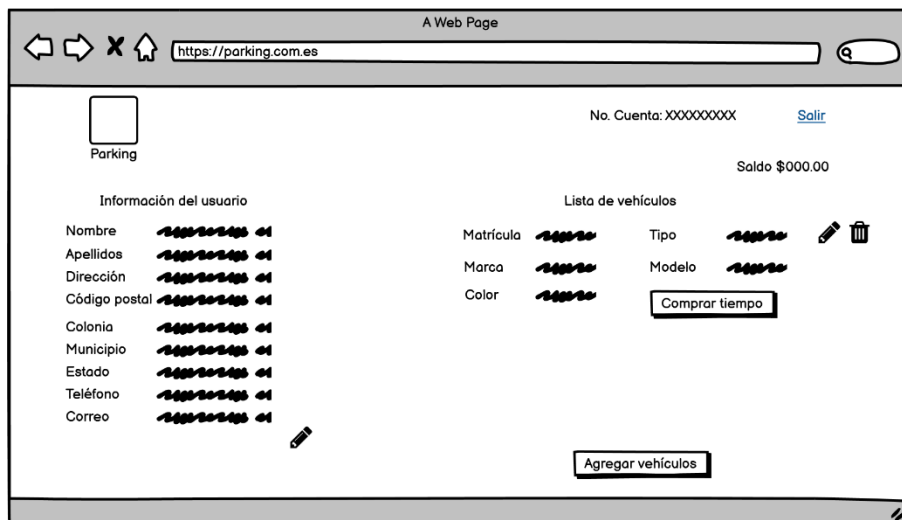
Dentro del módulo del conductor se dividirá en dos secciones la ventana, del lado izquierdo se encontrará la información del usuario, como su nombre completo, dirección, teléfono y correo. Del lado contrario se visualizarán todos los vehículos registrados por el conductor, además se mostrará la información de dicho vehículo.

En la parte superior se encuentra su saldo disponible, el cual servirá para mostrar un historial de compra, por último, en la parte inferior se encuentra un botón para agregar vehículos.

En ambos lados se encuentran iconos para la edición y eliminación de la información presentada.

Figura 26

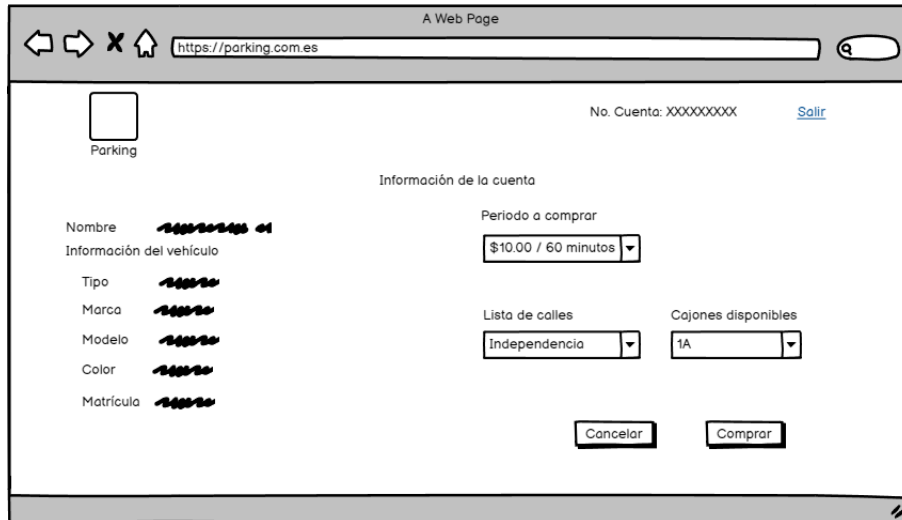
Pantalla de baja fidelidad catálogo conductor



Al seleccionar la opción “Comprar tiempo” mostrará el nombre del conductor y la información del vehículo, por otro lado, podrá seleccionar el tiempo a comprar, la calle y el cajón. También se cuenta con un botón de cancelar la compra en caso de que se haya seleccionado el vehículo incorrecto.

Figura 27

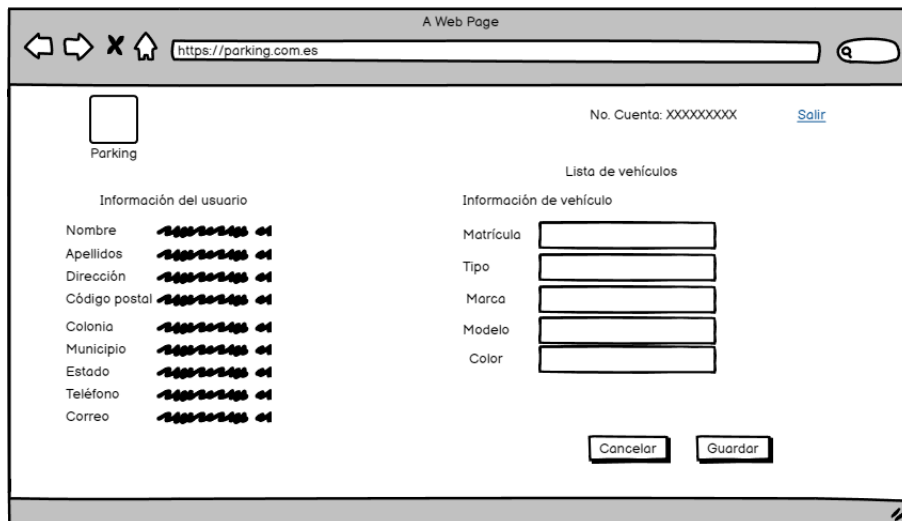
Pantalla de baja fidelidad compra de tiempo



Con la opción “Agregar vehículo” cambiará la sección donde se muestra toda la información de los vehículos por un formulario, en este se ingresará la matrícula, tipo, marca, modelo y color. Debajo contará con dos botones, uno para guardar la información registrada y otro para cancelar el registro.

Figura 28

Pantalla de baja fidelidad para registrar vehículo

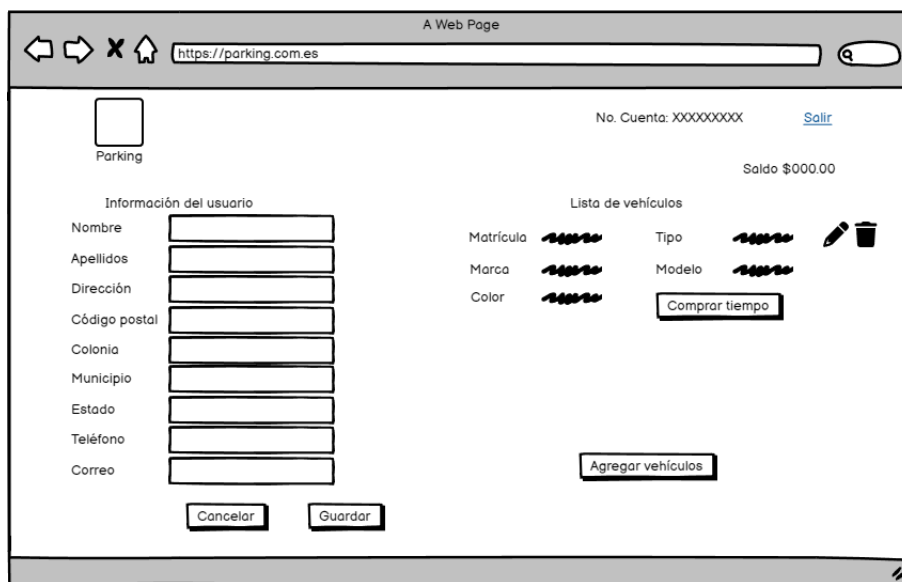


Con la opción de editar información, el conductor podrá actualizar su información personal en caso de requerirlo. Al seleccionar el icono abrirá un formulario con toda su información.

En la parte inferior se contará con dos botones, uno para guardar la información actualizada y otro para cancelar.

Figura 29

Pantalla de baja fidelidad para editar información del conductor



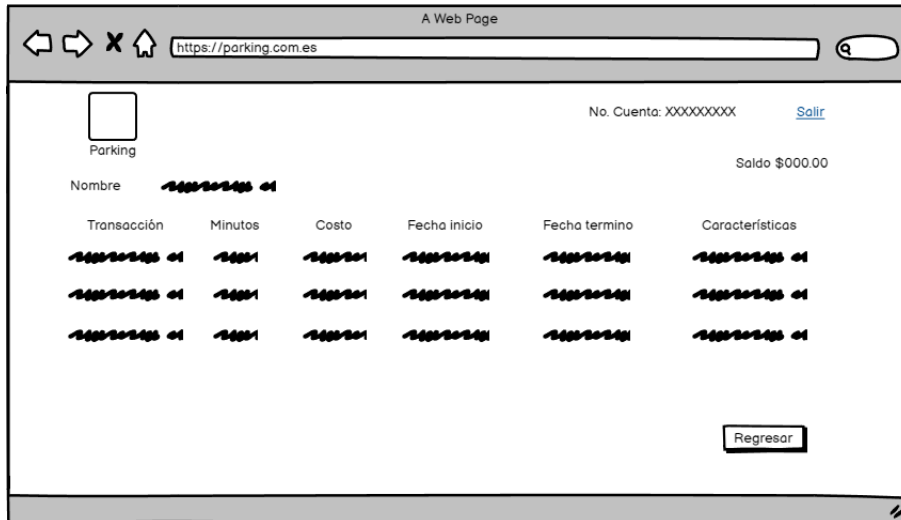
Por último, al seleccionar el saldo disponible mostrará un historial de este. Se indicará las compras de tiempo y pago de multas que han realizado respecto a todos los vehículos, en la cual se presenta información sobre la transacción, costo, fechas y características.

En la parte superior se mostrará el saldo disponible del usuario y en la parte inferior un botón para regresar a su página principal.

El historial del saldo será mostrado desde los más actuales a los últimos, es decir, de forma descendente respecto a la fecha se compra.

Figura 30

Pantalla de baja fidelidad historial de saldo

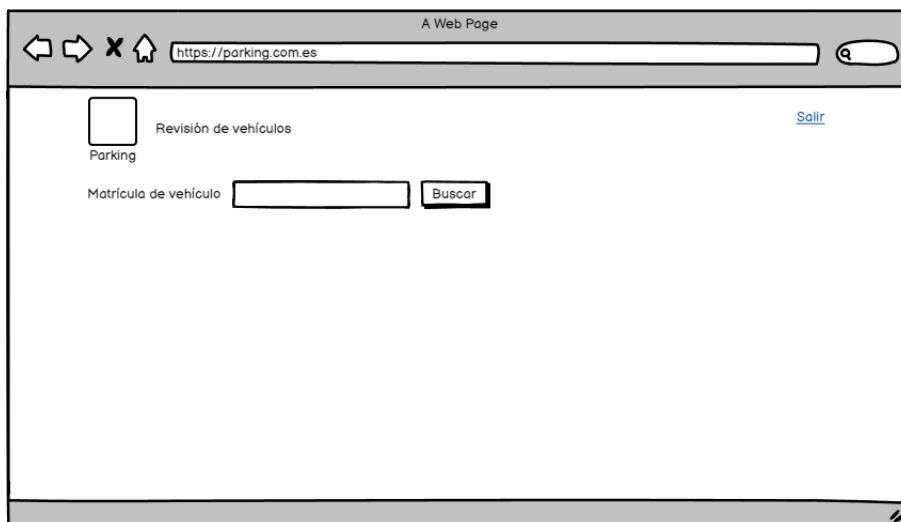


3.8.9. Diseñar Pantallas de Baja Fidelidad Catálogo Revisor

Dentro del catálogo de revisor contará con una sección de búsqueda, donde se ingresarán las placas del vehículo.

Figura 31

Pantalla de baja fidelidad catálogo de revisor



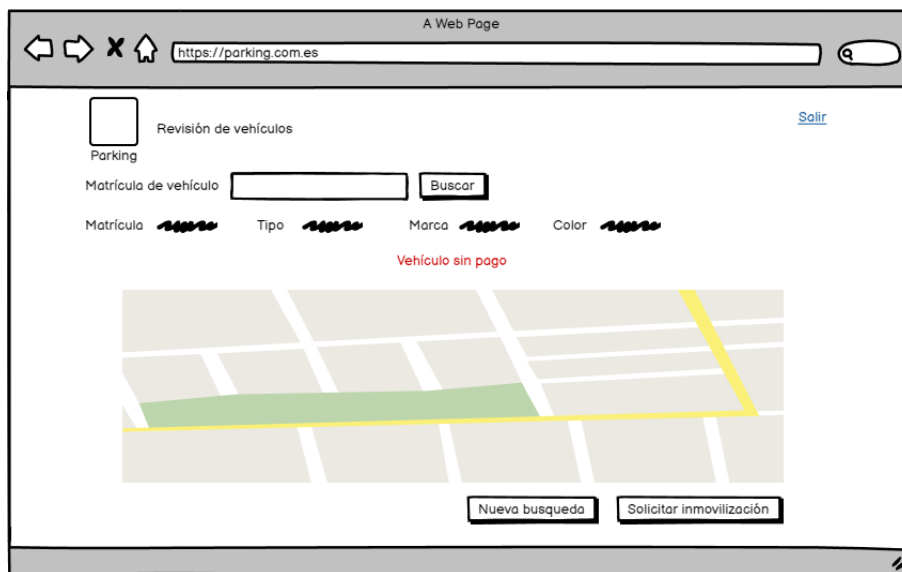
Una vez que se ingrese la placa, en la parte inferior tendrá toda la información del vehículo, como su matrícula, tipo, marca y color, al final se indicará si su tiempo se ha expirado o no y la ubicación del vehículo.

En la parte inferior se contará con dos botones, el primero para realizar una nueva búsqueda y el segundo para solicitar la inmovilización en caso de requerirse.

Una vez que el tiempo se ha terminado, se tienen minutos de tolerancia para permitir al conductor comprar más tiempo, en este lapso de tiempo el revisor no podrá solicitar la inmovilización del vehículo.

Figura 32

Pantalla de baja fidelidad búsqueda de vehículo

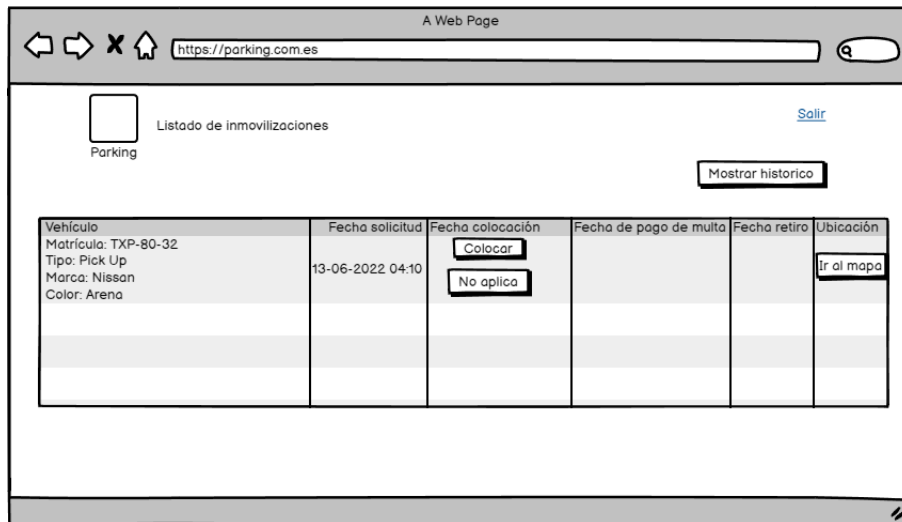


3.8.10. Diseñar Pantallas de Baja Fidelidad Catálogo Inmovilizador

Dentro del catálogo de inmovilizador se contará con una tabla en la que se visualizarán todas las solicitudes realizadas por el revisor que aún no han sido evaluadas por el inmovilizador. Se visualizará la información del vehículo, fecha de solicitud y la ubicación de este.

Figura 33

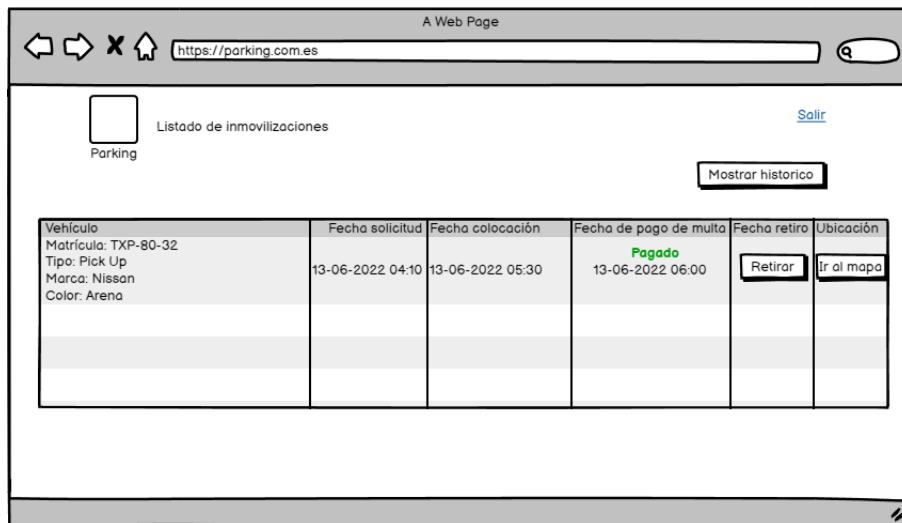
Pantalla de baja fidelidad catálogo inmovilizador



Una vez que el conductor haya pagado la multa se agregará la fecha de pago de multa y tendrá la opción de retirar la araña inmovilizadora, seguido guardará la fecha y hora en tiempo real.

Figura 34

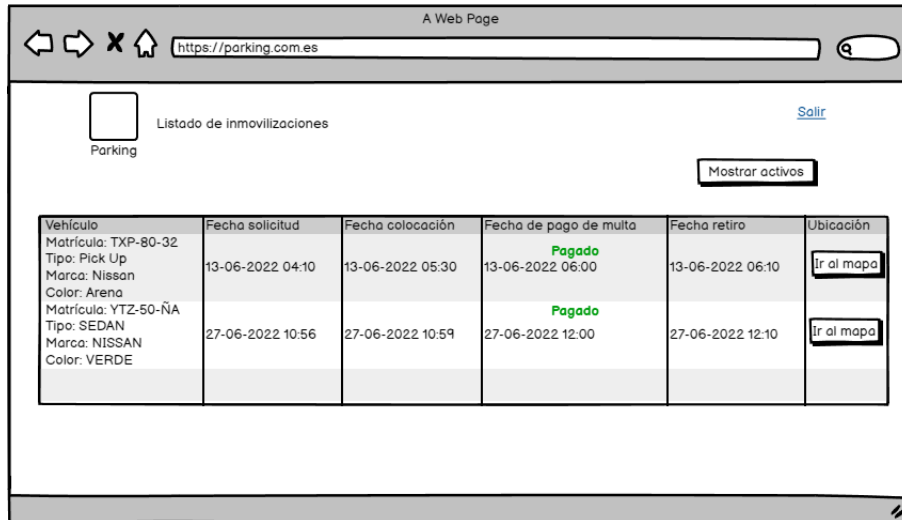
Pantalla de baja fidelidad multa pagada



Con la opción de “Mostrar histórico” consultará todos los movimientos realizados.

Figura 35

Pantalla de baja fidelidad histórico de inmobilizaciones



Con el botón "Ir al mapa", se visualizará la ubicación donde se estacionó el vehículo, junto con su matrícula, tipo, marca y color.

Figura 36

Pantalla de baja fidelidad opción "Ir al mapa"

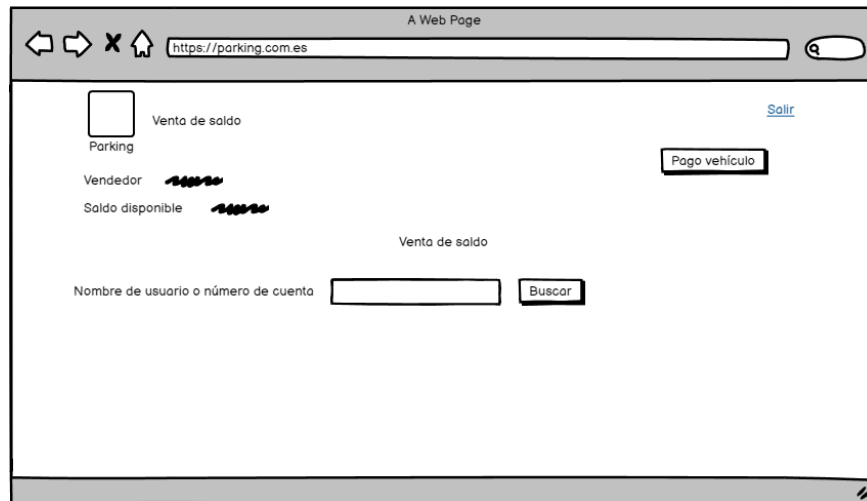


3.8.11. Diseñar Pantallas de Baja Fidelidad Catálogo Vendedor

En el catálogo de vendedor contará con una sección de búsqueda la cual se podrá realizar por número de cuenta o nombre, de esta forma el vendedor seleccionará la cantidad de saldo a vender.

Figura 37

Pantalla de baja fidelidad catálogo vendedor



Cuando se ingrese el nombre o la cuenta del conductor aparecerá una tabla con los resultados de la búsqueda en caso de tener más de un resultado en el sistema, en la parte inferior se mostrará el número total de resultados obtenidos.

A continuación, se visualizará en la tabla el número de cuenta, nombre completo y correo electrónico de los resultados de la búsqueda.

Además de la información presentada se tendrá un campo de opciones que tendrá un icono, al seleccionar dicho icono, se visualizará una sección en la cual se ingresará el monto a vender a la cuenta seleccionada.

En la parte inferior se cuenta con dos botones, uno para realizar la venta de saldo y otro para cancelar la venta.

Figura 38

Pantalla de baja fidelidad búsqueda de cuenta

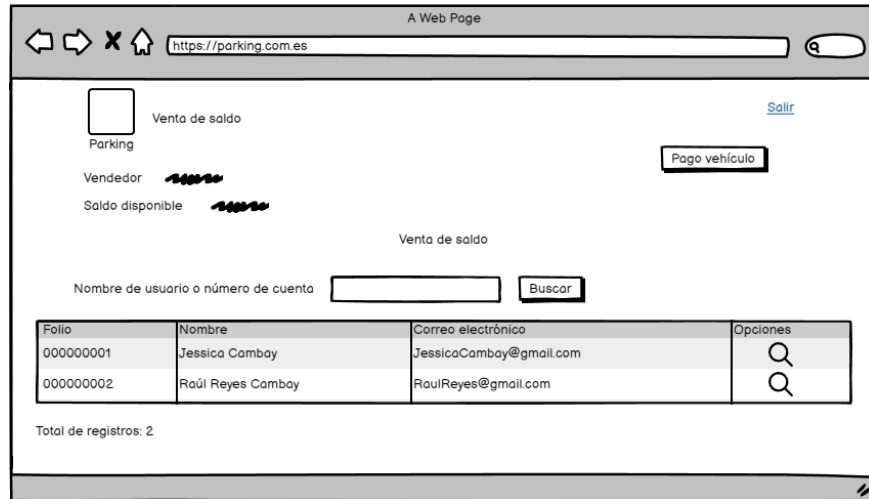
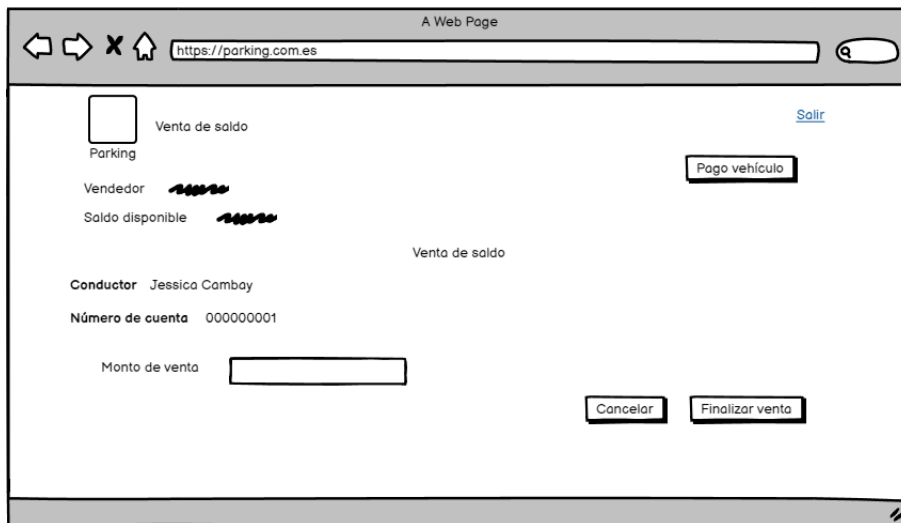


Figura 39

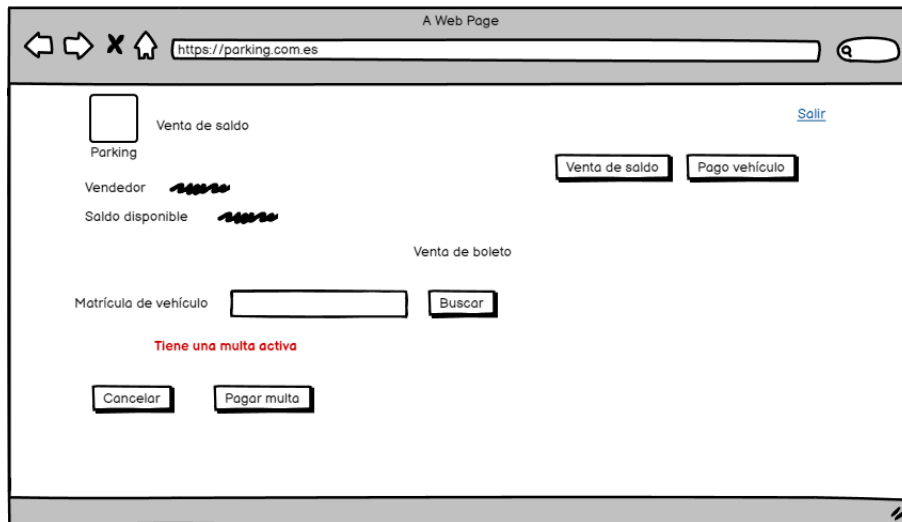
Pantalla de baja fidelidad venta saldo



En el caso de “Pago vehículo” se ingresará la placa del vehículo, en caso de que el conductor tenga una multa se mostrará en la pantalla junto con la opción de pagar o cancelar. En caso que el conductor no deba multas, se le permitirá comprar tiempo para el vehículo ingresado, solo se ingresará el tiempo a comprar.

Figura 40

Pantalla de baja fidelidad pago de vehículo



3.8.12. Diseñar Pantallas de Baja Fidelidad Catálogo Administrador

El catálogo de administrador contará con un menú con 6 ventanas disponibles, las cuales son: vendedores, revisores, inmovilizadores, conductores, costos del sistema, calles y cajones.

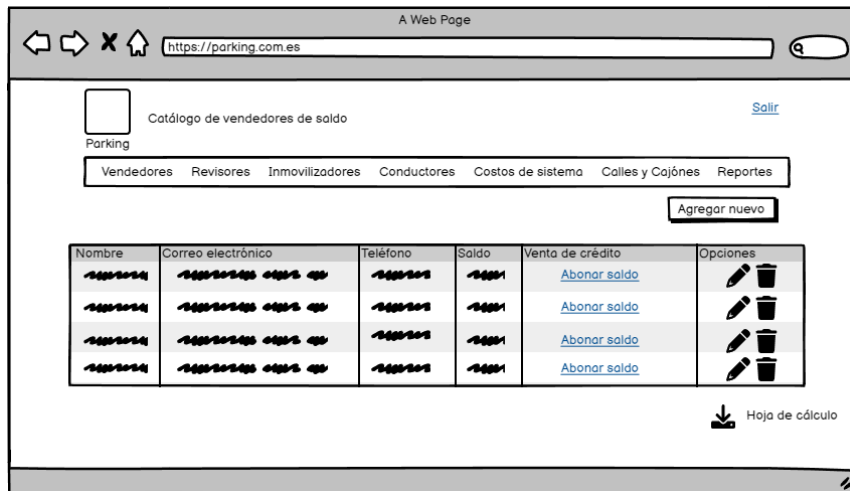
El administrador tendrá permitido agregar, eliminar, editar y visualizar la información del sistema. En cada módulo el administrador tendrá diferentes permisos, así mismo, podrá descargar reportes con información de cada módulo.

En la ventana de vendedores se visualizarán todos los vendedores registrados en el sistema. El administrador tendrá los permisos de agregar, visualizar, eliminar, editar la información del usuario y abonarle saldo.

En la parte inferior de la tabla, se contará con un apartado dónde permitirá descargar un reporte con esta información en formato de hoja de cálculo y PDF, del otro lado el número total de registros.

Figura 41

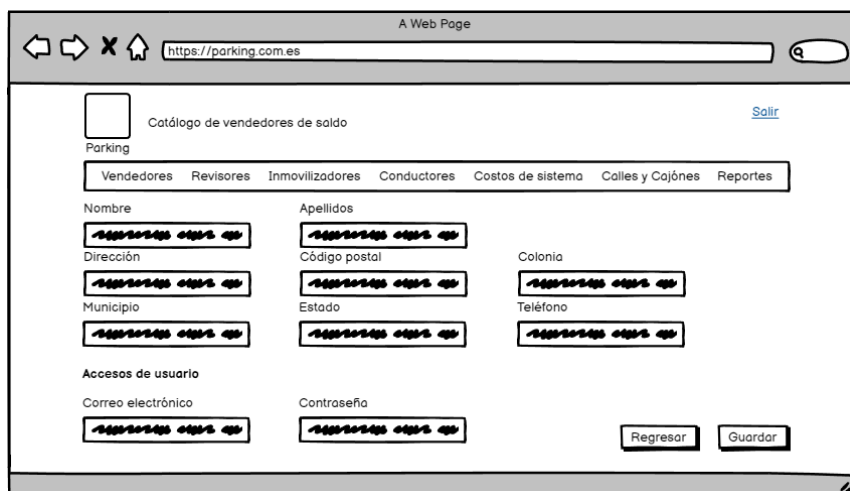
Pantalla de baja fidelidad catálogo administrador sección vendedores



Cuando se seleccione la opción de editar se visualizará un formulario con toda la información del vendedor seleccionado. Debajo contará con dos botones, uno para guardar la información y otro para regresar a la ventana anterior.

Figura 42

Pantalla de baja fidelidad editar información de vendedor



En el caso de seleccionar la opción “Agregar nuevo”, se mostrará la estructura del formulario, tal como la figura anterior. Se ingresará el nombre completo, dirección y datos

de acceso. Al final se cuenta con dos botones, uno para guardar la información y otro para regresar a la ventana anterior.

Figura 43

Pantalla de baja fidelidad nuevo vendedor

En la ventana de revisores se visualizará información general de estos, como el nombre, correo y teléfono, el administrador contará con permisos para agregar, editar y eliminar. Al final de la tabla se encontrará una opción para descargar la información.

Figura 44

Pantalla de baja fidelidad catálogo administrador sección revisores

Cuando se seleccione la opción de editar se visualizará un formulario con toda la información del revisor seleccionado.

Figura 45

Pantalla de baja fidelidad editar información de revisor

A Web Page
https://parking.com.es

Catálogo de revisores de vehículos [Salir](#)

Parking

Vendedores Revisores Inmovilizadores Conductores Costos de sistema Calles y Cajónes Reportes

Nombre [input] Apellidos [input]
Dirección [input] Código postal [input] Colonia [input]
Municipio [input] Estado [input] Teléfono [input]

Accesos de usuario
Correo electrónico [input] Contraseña [input]

Regresar Guardar

En el caso de seleccionar la opción “Agregar nuevo”, se mostrará la estructura del formulario, tal como la figura 45 y contará con botones para guardar o cancelar.

Figura 46

Pantalla de baja fidelidad nuevo revisor

A Web Page
https://parking.com.es

Catálogo de revisores de vehículos [Salir](#)

Parking

Vendedores Revisores Inmovilizadores Conductores Costos de sistema Calles y Cajónes Reportes

Nombre [input] Apellidos [input]
Dirección [input] Código postal [input] Colonia [input]
Municipio [input] Estado [input] Teléfono [input]

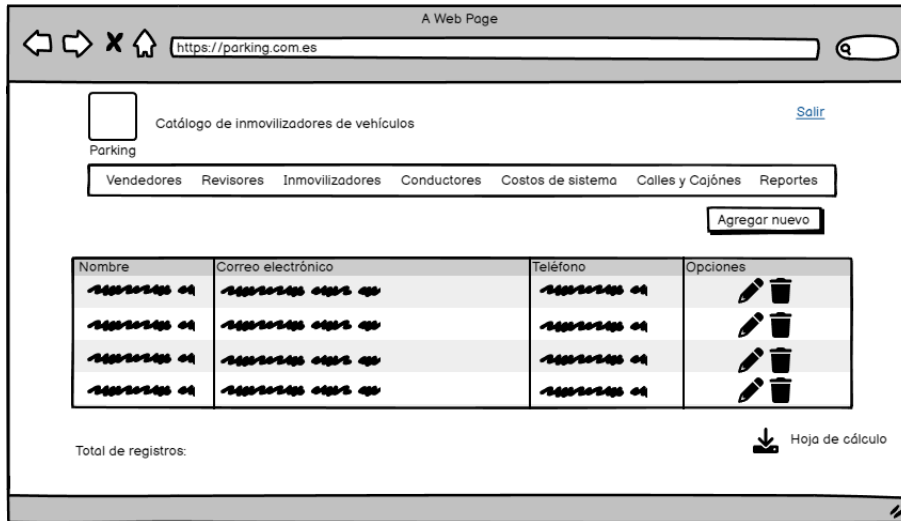
Accesos de usuario
Correo electrónico [input] Contraseña [input]

Regresar Guardar

En la ventana de inmovilizadores se visualizará información general de estos, el administrador contará con permisos para agregar, editar y eliminar. Al final de la tabla se encuentra una opción para descargar la información.

Figura 47

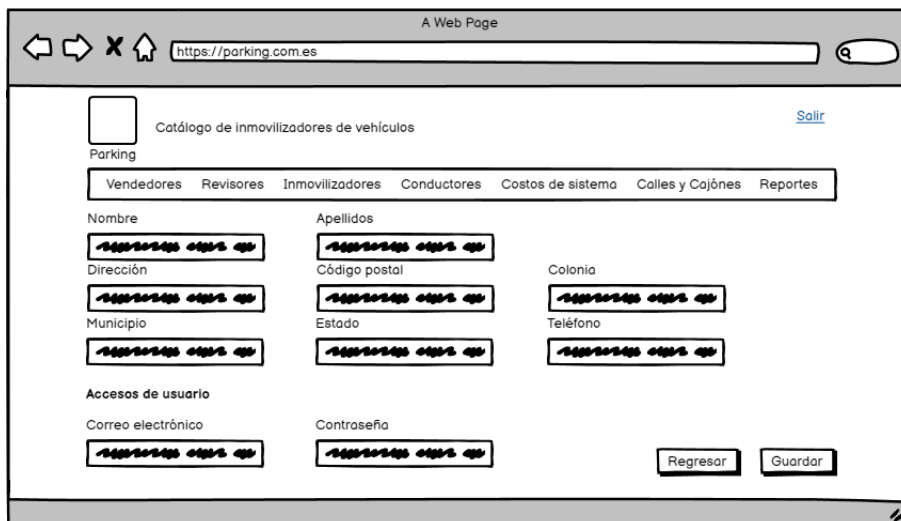
Pantalla de baja fidelidad catálogo administrador sección inmovilizador



Con la opción de editar, tendrá el formulario para actualizar la información.

Figura 48

Pantalla de baja fidelidad editar información de inmovilizador



En el caso de seleccionar la opción “Agregar nuevo”, se mostrará la estructura del formulario, tal como la figura anterior.

Figura 49

Pantalla de baja fidelidad nuevo inmovilizador

Browser: A Web Page
Address: https://parking.com.es

Page Title: Catálogo de inmovilizadores de vehículos
Link: Salir

Navigation: Vendedores Revisores Inmovilizadores Conductores Costos de sistema Calles y Cajones Reportes

Form Fields:
Nombre: [] Apellidos: []
Dirección: [] Código postal: [] Colonia: []
Municipio: [] Estado: [] Teléfono: []
Accesos de usuario:
Correo electrónico: [] Contraseña: []

Buttons: Regresar Guardar

En la ventana de conductores se visualizará la información general de estos. Al final de la tabla se encuentra una opción para descargar la información.

Figura 50

Pantalla de baja fidelidad catálogo administrador sección conductor

Browser: A Web Page
Address: https://parking.com.es

Page Title: Catálogo de conductores
Link: Salir

Navigation: Vendedores Revisores Inmovilizadores Conductores Costos de sistema Calles y Cajones Reportes

Buttons: Agregar nuevo

Nombre	Correo electrónico	Teléfono	Saldo
XXXXXXXXXX	XXXXXXXXXX@XXXXXX	XXXXXXXXXX	XXXXXXXXXX
XXXXXXXXXX	XXXXXXXXXX@XXXXXX	XXXXXXXXXX	XXXXXXXXXX
XXXXXXXXXX	XXXXXXXXXX@XXXXXX	XXXXXXXXXX	XXXXXXXXXX
XXXXXXXXXX	XXXXXXXXXX@XXXXXX	XXXXXXXXXX	XXXXXXXXXX

Total de registros: []
Download icon: Hoja de cálculo

En el módulo de conductor, el administrador solo tendrá permitido agregar nuevos conductores y visualizar su información.

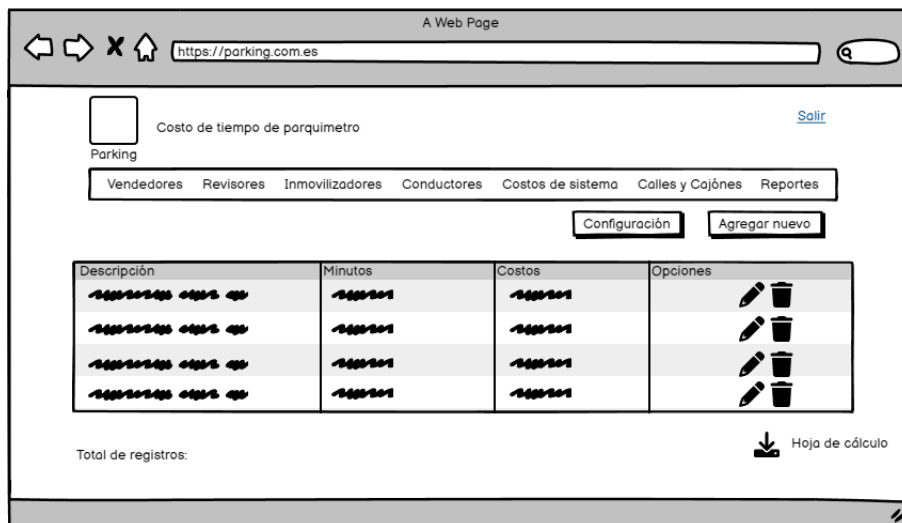
En la ventana de costos del sistema se muestra una tabla donde se almacenan los tiempos y su costo, el administrador cuenta con los permisos para agregar, edición y eliminación de la información.

En la parte superior se encuentran dos botones, en el primero servirá para configurar el costo de las multas y la tolerancia. La tolerancia es el tiempo que se tiene para pagar más tiempo, al término de este el revisor podrá colocar una multa. Por otro lado, siguiente botón sirve para agregar un nuevo costo.

Por último, se cuenta con la opción de descargar un informe con la información de la tabla.

Figura 51

Pantalla de baja fidelidad catálogo administrador sección costos del sistema

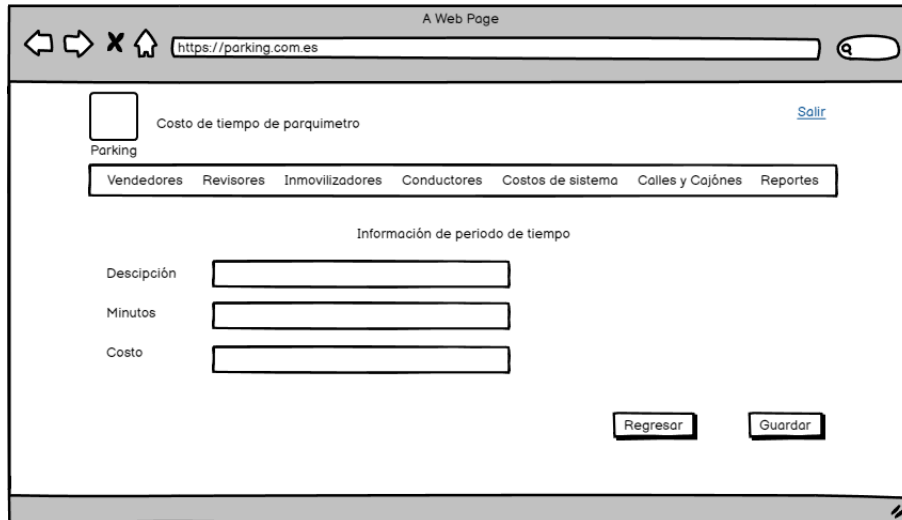


Al seleccionar la opción “Agregar nuevo” se visualizará un formulario donde tendrá campos para registrar la descripción, minutos y el costo de este.

Debajo contará con dos botones, uno para guardar la información y otro para cancelarlo en caso de requerirse.

Figura 52

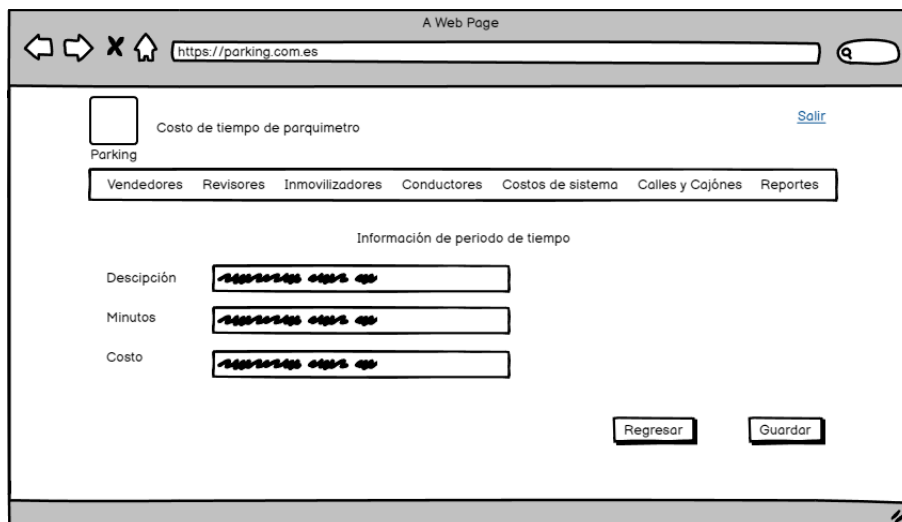
Pantalla de baja fidelidad nuevo costo del sistema



Al seleccionar la opción de editar se mostrará el formulario con la información, la parte inferior contará con dos botones, uno para guardar los cambios y otro para regresar.

Figura 53

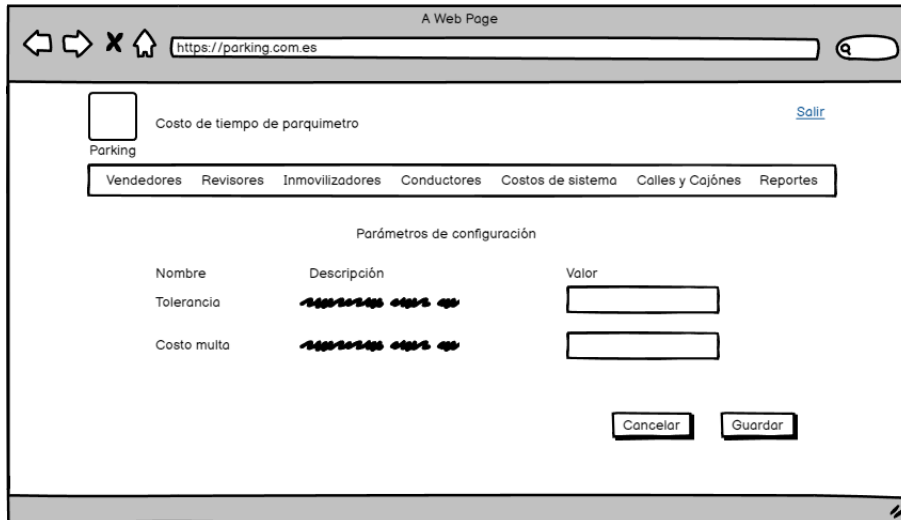
Pantalla de baja fidelidad editar información costos



Con la opción de "Configuración" se podrá cambiar el tiempo de tolerancia y el costo de las multas.

Figura 54

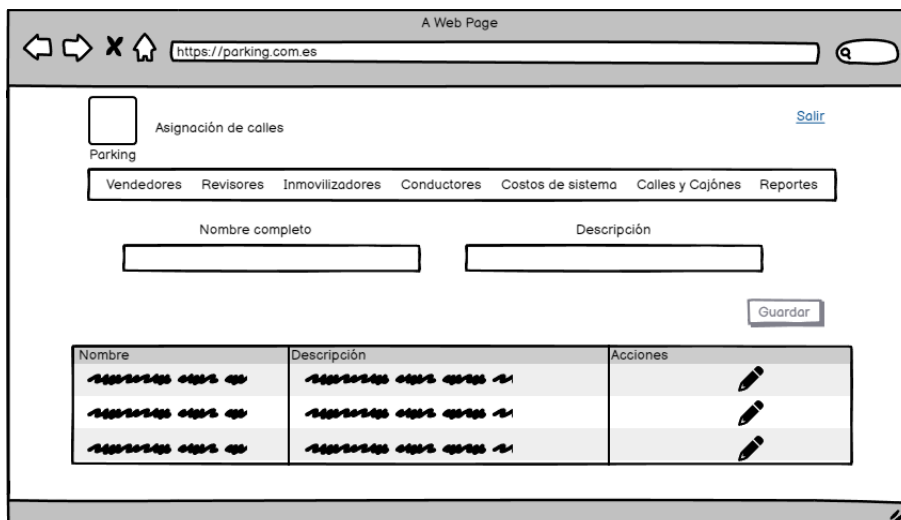
Pantalla de baja fidelidad de opción "Configuración"



En la ventana de calles el administrador agregará calles y una descripción, así mismo, podrá editar esta información.

Figura 55

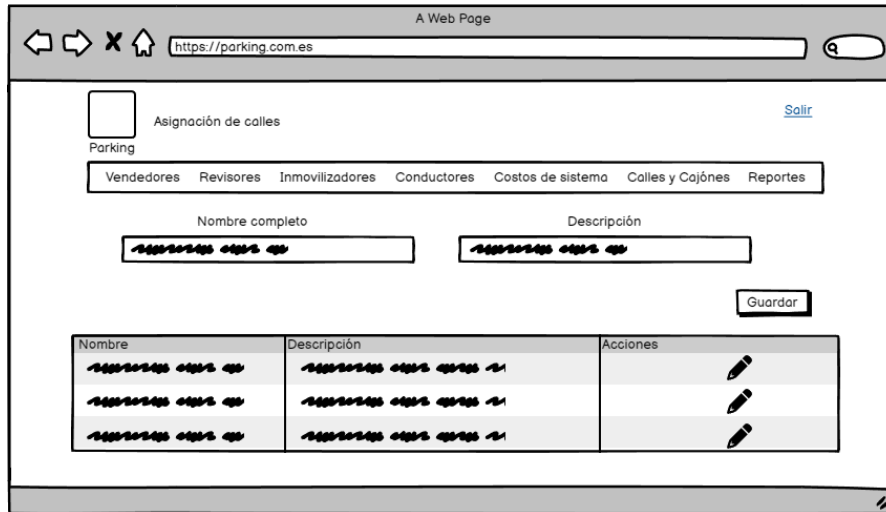
Pantalla de baja fidelidad catálogo administrador sección calles



Para agregar una nueva calle solo es necesario escribir la información en los recuadros que se encuentran en la parte superior de la tabla.

Figura 56

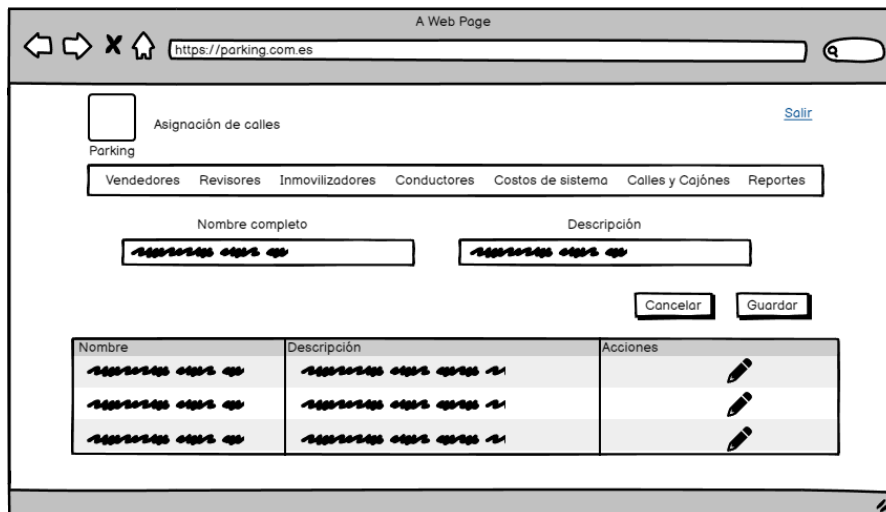
Pantalla de baja fidelidad nueva calle



Al seleccionar la opción editar llenará los recuadros de la parte superior con la información de la calle seleccionada, también tendrá dos botones, uno para cancelar y otro para guardar.

Figura 57

Pantalla de baja fidelidad editar información calle



En la ventana de cajones el administrador agregará el número de cajón de acuerdo a una calle, así mismo, editará esta información.

Figura 58

Pantalla de baja fidelidad catálogo administrador sección cajones

Asignación de Cajones [Salir](#)

Parking

Vendedores Revisores Inmovilizadores Conductores Costos de sistema Calles y Cajones Reportes

No. Cajón

Nombre de la calle

No. Cajón	Nombre de la calle	Acciones
XXXXXXXXXX	XXXXXXXXXX	
XXXXXXXXXX	XXXXXXXXXX	
XXXXXXXXXX	XXXXXXXXXX	

Para agregar un nuevo cajón solo es necesario escribir la información en el recuadro y seleccionar la calle.

Figura 59

Pantalla de baja fidelidad nuevo cajón

Asignación de Cajones [Salir](#)

Parking

Vendedores Revisores Inmovilizadores Conductores Costos de sistema Calles y Cajones Reportes

No. Cajón

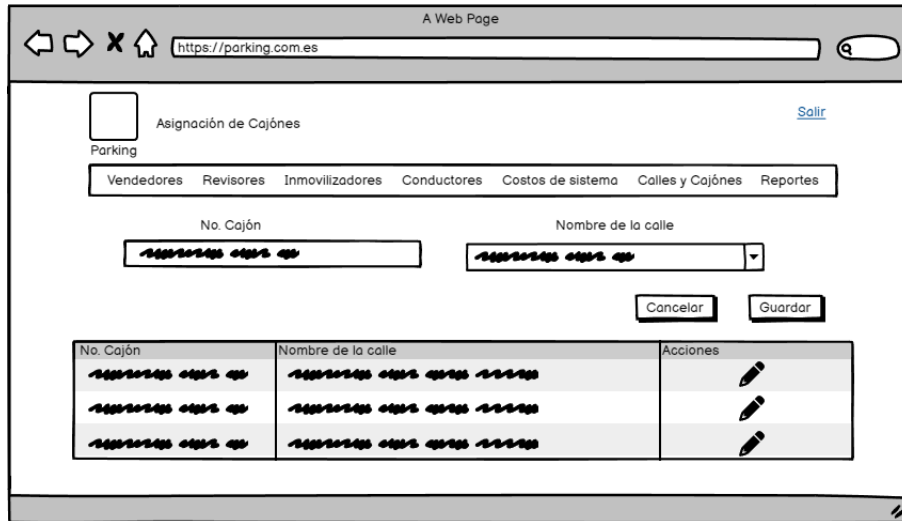
Nombre de la calle

No. Cajón	Nombre de la calle	Acciones
XXXXXXXXXX	XXXXXXXXXX	
XXXXXXXXXX	XXXXXXXXXX	
XXXXXXXXXX	XXXXXXXXXX	

Por último, para editar la información se debe seleccionar el lápiz de la derecha, su información se mostrará en los recuadros de arriba junto con dos botones, uno para guardar y otro para cancelar.

Figura 60

Pantalla de baja fidelidad editar información cajón



3.9. Burndown Chart Sprint 1

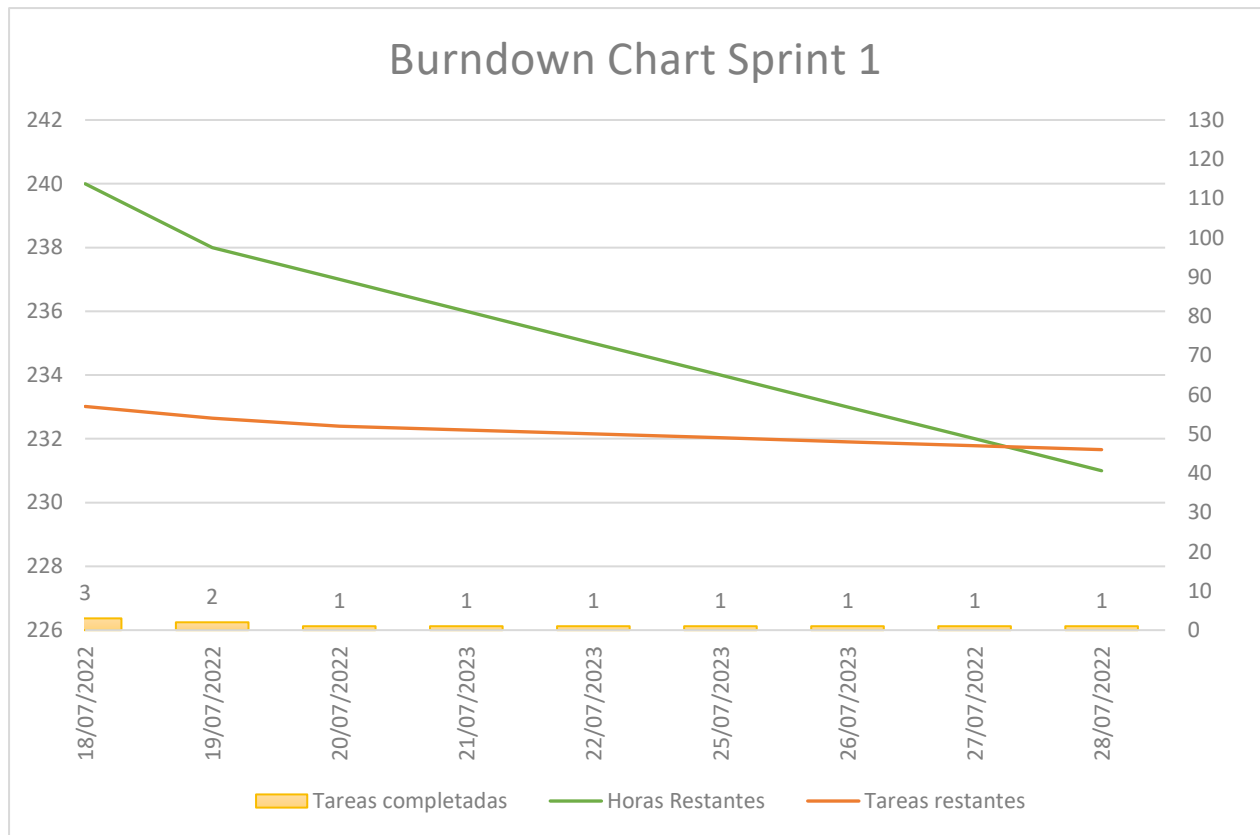
En la figura 61 se muestra el avance del sprint 1 en forma gráfica. Las barras de color amarillo representan la cantidad de tareas realizadas según la fecha indicada, la línea verde representa las horas restantes del proyecto y la línea naranja las tareas restantes.

En este sprint se avanzó con las bases del proyecto, instalando las herramientas necesarias junto con la creación de pantallas de baja fidelidad, con el fin de tener un boceto de las interfaces del proyecto.

En este sprint no hubo la necesidad de recorrer tareas para el sprint 2, debido a que todas las tareas se completaron, lo que significa que el proyecto va acorde a lo planeado.

Figura 61

Burndown Chart Sprint 1



3.10. Ejecución Sprint 2

En la segunda iteración se realizaron las siguientes 15 actividades, para la continuación del desarrollo del sistema. Si no se cumplen en su totalidad las actividades en el primer Sprint planeado se considerarán en el siguiente Sprint.

En la tabla 6 se presentan las tareas del Sprint 2 junto con la fecha programada, estado del desarrollo y el tiempo que se requiere. Se cuenta con un total de 15 tareas, de las cuales están enfocadas en el diseño de pantallas de alta fidelidad, creación del view y Controller del sistema.

Tabla 6*Sprint 2*

Sprint 2						
No.	Tarea	Responsable	Fecha	Sprint	Estado	Horas
1	Diseñar pantalla de alta fidelidad módulo Inicio de sesión	Mitzi Sollange Hernández Juárez	01/08/2022	2	Completado	2
2	Diseñar pantalla de alta fidelidad módulo registro	Mitzi Sollange Hernández Juárez	03/08/2022	2	Completado	2
3	Diseñar pantalla de alta fidelidad catálogo de conductor	Mitzi Sollange Hernández Juárez	05/08/2022	2	Completado	2
4	Diseñar pantalla de alta fidelidad catálogo de revisor	Mitzi Sollange Hernández Juárez	09/08/2022	2	Completado	2
5	Diseñar pantalla de alta fidelidad catálogo de inmovilizador	Mitzi Sollange Hernández Juárez	11/08/2022	2	Completado	2
6	Diseñar pantalla de alta fidelidad catálogo de vendedor	Mitzi Sollange Hernández Juárez	15/08/2022	2	Completado	2
7	Diseñar pantalla de alta fidelidad catálogo de administrador	Mitzi Sollange Hernández Juárez	17/08/2022	2	Completado	2
8	Crear View y Controller para Catálogo de Cajones	Mitzi Sollange Hernández Juárez	22/08/2022	2	Completado	3
9	Crear View y Controller para Catálogo de calles	Mitzi Sollange Hernández Juárez	24/08/2022	2	Completado	3
10	Crear View y Controller para Catálogo de usuarios	Mitzi Sollange Hernández Juárez	26/08/2022	2	Completado	7
11	Crear View y Controller para Catálogo de costo tiempos	Mitzi Sollange Hernández Juárez	30/08/2022	2	Completado	5

Sprint 2						
No.	Tarea	Responsable	Fecha	Sprint	Estado	Horas
12	Crear View y Controller para Catálogo de inmovilizaciones	Mitzi Sollange Hernández Juárez	01/09/2022	2	Completado	5
13	Crear View y Controller para Catálogo de revisiones	Mitzi Sollange Hernández Juárez	05/09/2022	2	Completado	5
14	Crear View y Controller para Catálogo de sesión	Mitzi Sollange Hernández Juárez	07/09/2022	2	Completado	3
15	Crear View y Controller para Catálogo de venta de saldo	Mitzi Sollange Hernández Juárez	09/09/2022	2	Completado	5

3.11. Resultados Obtenidos del Sprint 2

A continuación, se muestran los resultados obtenidos en el sprint 2, los cuales se describieron anteriormente en la tabla 6.

Debido a la carta de confidencialidad (Ver anexo A), solo se tiene permitido mostrar 6 capturas del sistema, por otro lado, no se tiene permitido mostrar la funcionalidad, código y lógica del proyecto, por lo tanto, se mostrarán ejemplos generales ajenos al proyecto sacados de páginas de internet, respecto a la creación del View y Controller.

3.11.1. Diseñar Pantalla de Alta Fidelidad Módulo Inicio de Sesión

En base a la pantalla de baja fidelidad desarrollada en el sprint pasado, se creó la pantalla de alta fidelidad de inicio de sesión. Se realizará la autenticación con el correo electrónico y la contraseña que han ingresado en su registro, debajo se encuentra el botón de inicio de sesión, una opción de registro y la política de privacidad otorgada por la empresa SmartSoft America Business Applications S.A. de C.V.

Esta pantalla de alta fidelidad se siguió y se cumplió en base a la pantalla de baja fidelidad antes desarrollada en el sprint 1.

En este módulo, se autenticarán los 6 diferentes tipos de usuarios del presente proyecto.

Figura 62

Pantalla de alta fidelidad inicio de sesión



La imagen muestra una interfaz de usuario para el inicio de sesión de un sistema web llamado "Parking". El diseño es limpio y moderno, con un fondo blanco y un borde naranja. En la parte superior, se encuentra el logo "Parking" con un ícono de un estacionamiento. Debajo del logo, hay un título "Inicio de sesión" centrado. A continuación, se ven dos campos de entrada: "Correo electrónico" y "Contraseña", ambos con bordes redondeados y un ícono de lupa a la izquierda. Debajo de los campos, hay un botón naranja con el texto "Iniciar sesión". En la parte inferior, hay dos enlaces: "Registrarse" y "Política de privacidad", ambos con un subrayado.

Nota. Captura de pantalla sistema web Parking, por SmartSoft America Business Applications S.A. de C.V., 2022.

3.11.2. Diseñar Pantalla de Alta Fidelidad Módulo Registro

Para la sección de registro es necesario el nombre, apellidos, dirección, código postal, colonia, municipio, estado, teléfono, correo electrónico y contraseña.

En este formulario, solo se registrarán conductores, para el registro de los diferentes usuarios se tiene que realizar desde el módulo de administrador.

El diseño a comparación de la pantalla de baja fidelidad varía en la estructura, con el fin de hacerlo más práctico para el usuario, sin embargo, conserva todos los campos planteados.

Figura 63

Pantalla de alta fidelidad registro de conductor

T Parking

Registro de usuario

Nombre*

Apellidos

Dirección*

Código Postal*

Colonia*

Municipio*

Estado*

Teléfono*

Correo electrónico*

Contraseña*

Repite la contraseña*

Aceptar terminos y condiciones

Registrarme

[Iniciar sesión](#)

Nota. Captura de pantalla sistema web Parking, por SmartSoft America Business Applications S.A. de C.V., 2022.

3.11.3. Diseñar Pantalla de Alta Fidelidad Catálogo de Conductor

La pantalla de alta fidelidad del usuario conductor se dividirá en dos secciones, por el lado izquierdo encontramos toda su información personal y del lado contrario los vehículos registrados por este conductor.

Figura 64

Pantalla de alta fidelidad catálogo conductor



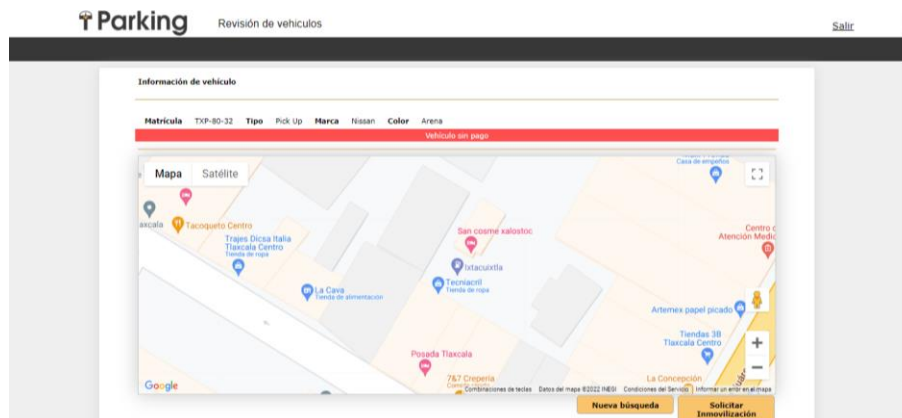
Nota. Captura de pantalla sistema web Parking, por SmartSoft America Business Applications S.A. de C.V., 2022.

3.11.4. Diseñar Pantalla de Alta Fidelidad Catálogo de Revisor

La pantalla de alta fidelidad de revisor consta de un menú de búsqueda, en el cual se debe ingresar las placas del vehículo para saber su ubicación, información general de vehículo y su estatus, debajo cuenta con un botón para solicitar la inmovilización y realizar nuevamente otra búsqueda.

Figura 65

Pantalla de alta fidelidad catálogo revisor



Nota. Captura de pantalla sistema web Parking, por SmartSoft America Business Applications S.A. de C.V., 2022.

3.11.5. Diseñar Pantalla de Alta Fidelidad Catálogo de Inmovilizador

En la ventana del usuario inmovilizador se visualizará una tabla con la información del vehículo, fechas de solicitud, colocación y pago de multa. Se visualizará la ubicación del vehículo y el histórico de todas las solicitudes realizadas.

Figura 66

Pantalla de alta fidelidad catálogo inmovilizador



Nota. Captura de pantalla sistema web Parking, por SmartSoft America Business Applications S.A. de C.V., 2022.

3.11.6. Diseñar Pantalla de Alta Fidelidad Catálogo de Vendedor

La ventana del usuario vendedor contiene una sección de búsqueda donde se ingresará el folio, nombre y/o apellidos para hacer la consulta, los resultados de esta se mostrarán en la tabla de la página.

Figura 67

Pantalla de alta fidelidad catálogo vendedor



Nota. Captura de pantalla sistema web Parking, por SmartSoft America Business Applications S.A. de C.V., 2022.

3.11.7. Diseñar Pantalla de Alta Fidelidad Catálogo de Administrador

El administrador tendrá acceso a toda la información, configuración del sistema, asignando costos de multas, tiempos, consulta, registro y eliminación de revisores, inmovilizadores, vendedores, calles, cajones.

Las ventanas de alta fidelidad se desarrollaron de acuerdo a lo planteado en las pantallas de baja fidelidad, desarrolladas en el sprint pasado, respetando el diseño y funcionalidades planteadas.

3.11.8. Creación de View

A continuación, en la tabla 7 se mostrará un ejemplo de una estructura de una View ajena al proyecto. La función principal de esta clase es instanciar todos los campos de una tabla de una base de datos. Solo se almacena la información de forma local para realizar las diferentes operaciones dentro del sistema. Dentro de esta se declararán todos los objetos que se utilizarán en la vista.

Tabla 7

Código de ejemplo de View

```
public class ChessGameView {
    @Id
    private Long id;
    private int gameRound;
    private String tournamentName;
    public Long getId() {
        return id;
    }
    public void setId(Long id) {
        this.id = id;
    }
    public int getGameRound() {
        return gameRound;
    }
}
```

```

    }
    public void setGameRound(int gameRound) {
        this.gameRound = gameRound;
    }
    public String getTournamentName() {
        return tournamentName;
    }
    public void setTournamentName (String tournamentName) {
        this.tournamentName = tournamentName;
    }
}

```

Nota. Adaptado de “Thorben Janssen”, por Janssen T., 2022, (<https://thorben-janssen.com/spring-data-jpa-views/>). Todos los derechos reservados 2022 por Licenciario. Reproducido con permiso del autor.

En el ejemplo de la tabla 7, se presenta una corta estructura de una View, esta se compone de declaración de campos, los getter y setter de los respectivos campos. Por medio de estos, se puede mapear y consultar información de las tablas de su base de datos.

3.11.9. Creación de Controller

Para el desarrollo del Controller se utilizará un ejemplo de su estructura, el cual se presenta en la tabla 8. La función del Controller es proporcionar funcionalidad al código, está integrado por métodos y conecta el Back-End con las solicitudes que se realizan desde afuera de la aplicación.

Tabla 8

Código de ejemplo de Controller

```

package com.studytonight.controllers;
import javax.validation.Valid;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.validation.BindingResult;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.ModelAttribute;

```

```

import org.springframework.web.bind.annotation.PostMapping;
import com.studytonight.models.User;
import com.studytonight.service.UserService;
@Controller
public class RegisterController{
    @Autowired
    private UserService userService;
    @GetMapping("/")
    public String index() {
        return "register";
    }
    @PostMapping("save")
    public String save(@Valid @ModelAttribute User user, BindingResult result) {
        if(result.hasErrors()) {
            return "register";
        }else {
            userService.register(user);
            return "successful";
        }
    }
}

```

Nota. Adaptado de “Spring MVC With JPA”, por Study Tonight, 2022, (<https://www.studytonight.com/spring-framework/spring-mvc-with-jpa>). Todos los derechos reservados 2022 por Licenciario. Reproducido con permiso del autor.

En la tabla 8 se presentó un ejemplo sobre la estructura del Controller. Este ejemplo cuenta dos métodos: index y save, el primero solo retorna una cadena y el segundo guarda la información en la base de datos.

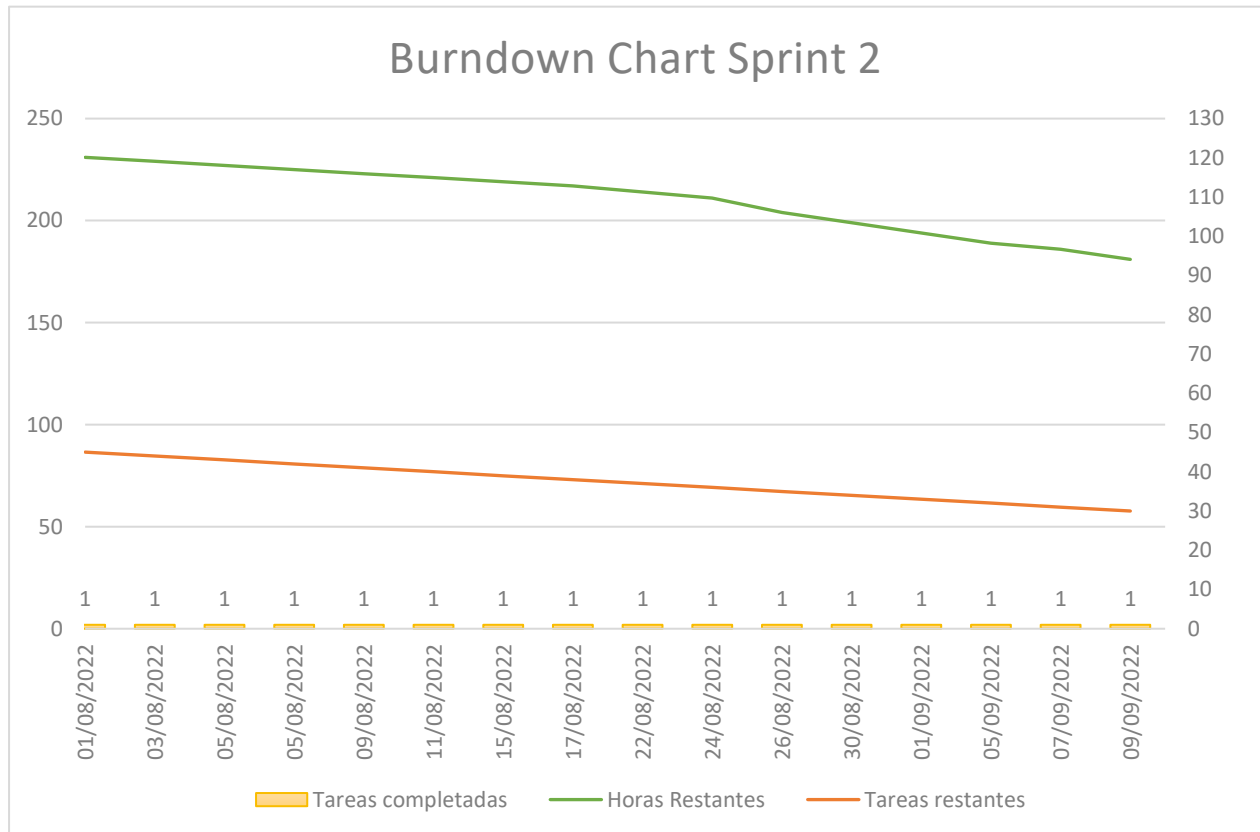
3.12. Burndown Chart Sprint 2

En la figura 68 se muestra el avance del sprint 2, las barras de color amarillo representan las tareas realizadas según la fecha indicada, la línea verde representa las horas restantes del proyecto y la línea naranja las tareas restantes

En este sprint no hubo la necesidad de recorrer tareas para el sprint 3, debido a que todas las tareas se completaron, lo que significa que el proyecto va acorde a lo planeado.

Figura 68

Burndown Chart Sprint 2



3.13. Ejecución Sprint 3

En la tercera iteración se realizaron las siguientes 13 actividades, para la continuación del desarrollo del sistema. Si no se cumplen en su totalidad las actividades en el Sprint planeado se considerarán en el siguiente.

En la tabla 9 se presentan las tareas del Sprint 3 junto con la fecha programada, estado del desarrollo y el tiempo que se requiere.

Tabla 9*Sprint 3*

Sprint 3						
No.	Tarea	Responsable	Fecha	Sprint	Estado	Horas
1	Crear Script SQL para la base de datos	Mitzi Sollange Hernández Juárez	14/09/2022	3	Completado	5
2	Crear Script de conexión Back - End con Front - End	Mitzi Sollange Hernández Juárez	16/09/2022	3	Completado	5
3	Crear Entity, DAO y Business para operación CRU en catálogo de cajones	Mitzi Sollange Hernández Juárez	20/09/2022	3	Completado	7
4	Crear Entity, DAO y Business para operación CRU en catálogo de calles	Mitzi Sollange Hernández Juárez	22/09/2022	3	Completado	7
5	Crear Entity, DAO y Business para operación CRUD en catálogo de usuarios	Mitzi Sollange Hernández Juárez	26/09/2022	3	Completado	7
6	Crear Entity, DAO y Business para operación CRUD en catálogo de costos tiempo	Mitzi Sollange Hernández Juárez	28/09/2022	3	Completado	7
7	Crear Entity, DAO y Business para operación CR en catálogo de venta saldo	Mitzi Sollange Hernández Juárez	30/09/2022	3	Completado	7
8	Crear Entity, DAO y Business para operación CR en catálogo de revisiones	Mitzi Sollange Hernández Juárez	04/10/2022	3	Completado	7
9	Crear Entity, DAO y Business para catálogo de sesión	Mitzi Sollange Hernández Juárez	06/10/2022	3	Completado	7
10	Crear Entity, DAO y Business para operación RU en parámetros de configuración	Mitzi Sollange Hernández Juárez	11/10/2022	3	Completado	7
11	Crear Entity, DAO y Business para operación R en catálogo de saldo utilizado	Mitzi Sollange Hernández Juárez	13/10/2022	3	Completado	7

Sprint 3						
No.	Tarea	Responsable	Fecha	Sprint	Estado	Horas
12	Crear Entity, DAO y Business para operación CRUD en catálogo de vehículos	Mitzi Sollange Hernández Juárez	17/10/2022	3	Completado	7
13	Implementación de generación de reportes en Excel y PDF	Mitzi Sollange Hernández Juárez	18/10/2022	3	Completado	7

3.14. Resultados Obtenidos del Sprint 3

A continuación, se muestran los resultados obtenidos en el sprint 3, los cuales se describieron anteriormente en la tabla 9.

Debido a la carta de confidencialidad (Ver anexo A), no se tiene permitido mostrar la funcionalidad del proyecto, por lo tanto, se mostrarán ejemplos generales ajenos al proyecto sacados de internet, respecto a la creación del Business, DAO, Entity y la generación de reportes.

3.14.1. Creación de Entity

El desarrollo del Entity se utilizará un ejemplo de su estructura, el cual se presenta en la tabla 10. La función del Entity es identificar los objetos de las respectivas tablas de una base de datos, en esta se declaran tanto claves primarias como foráneas.

Tabla 10

Código de ejemplo de Entity

```
package com.studytonight.models;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.Id;
@Entity
@Table(name = "usuarios")
@XmlRootElement
@NamedQueries({
    @NamedQuery(name = "Usuarios.findAll", query = "SELECT u FROM Usuarios u")
})
```

```

})
public class User {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Basic(optional = false)
    @Column(name = "id")
    private Long id;
    @Column(name = "name")
    private String name;
    @Column(name = "mobile")
    private Long mobile;
    public Long getId() {
        return id;
    }
    public void setId(Long id) {
        this.id = id;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public Long getMobile() {
        return mobile;
    }
    public void setMobile(Long mobile) {
        this.mobile = mobile;
    }
}

```

Nota. Adaptado de “Spring MVC With JPA”, por Study Tonight, 2022, (<https://www.studytonight.com/spring-framework/spring-mvc-with-jpa>). Todos los derechos reservados 2022 por Licenciatarario. Reproducido con permiso del autor.

Dentro de la estructura del Entity se encuentran algunas consultas ya establecidas, declaración de los objetos de las tablas y los métodos getter y setter, los cuales nos servirán para tomar o guardar valores.

3.14.2. Creación de DAO

Para el desarrollo del DAO se utilizará un ejemplo de su estructura, el cual se presenta en la tabla 11. El DAO permite la persistencia de datos, establece la conexión entre las consultas definidas del Entity y la información que pide el Business.

Tabla 11

Código de ejemplo de DAO

```
@Repository
public class OtherSuperheroDAO {
    private static Logger logger = Logger.getLogger(OtherSuperheroDAO.class);
    @PersistenceContext
    private EntityManager entityManager;
    public List<Superhero> findAll() {
        logger.info("En Other superhero DAO");
        Query query = getEntityManager().createQuery(
            "select entity from Superhero");
        return query.getResultList();
    }
    public EntityManager getEntityManager() {
        return entityManager;
    }
    public void setEntityManager(EntityManager entityManagerArg) {
        entityManager = entityManagerArg;
    }
}
```

Nota. Adaptado de “Creación de DAOs con Spring JPA”, por Luis Alberto G., 2015, (<https://www.studytonight.com/spring-framework/spring-mvc-with-jpa>). Todos los derechos reservados 2016 por Licenciatario. Reproducido con permiso del autor.

En el ejemplo de la tabla 11, se crea y ejecuta la consulta a una tabla llamada “Superhero” en la cual se tomará el campo “entity”, por último, retorna la información de la consulta.

3.14.3. Creación de Business

Para el desarrollo del Business se utilizará un ejemplo de su estructura, el cual se presenta en la tabla 12. El Business se desarrolla la funcionalidad del sistema, en esta se crean los métodos que sean necesarios, desde aquí se llaman las consultas por medio del DAO y manda información a la parte de la vista.

Tabla 12

Código de ejemplo de Business

```
package org.primefaces.showcase.service;
import java.time.LocalDateTime;
import java.util.ArrayList;
import java.util.Date;
import java.util.List;
import javax.ejb.Stateless;
import javax.inject.Inject;
import javax.persistence.EntityManager;
import javax.persistence.EntityManagerFactory;
@Stateless
public class Business {
    @Inject
    private EmpleadoDAO dao;
    public List<Business> obtenerAll(Integer id){
        return dao.obtenerAll(id);
    }
    public Empleado createEmpleado(int id, String nombre, long salario) {
        comprobanteComienzoTx();
        Empleado emp = new Empleado(id);
        emp.setNombre(nombre);
        emp.setSalario(salario);
        em.persist(emp);
        comprobanteFinTx();
        return emp;
    }
    public Empleado findEmpleado(int id) {
```

```

        return em.find(Employee.class, id);
    }
    public void removeEmpleado(Empleado emp) {
        pruebaComienzoTx();
        em.remove(emp);
    }
    public void empleadoAddProyecto(Empleado emp, Proyecto proy) {
        pruebaComienzoTx();
        emp.getProyectos().add(proy);
        proy.getEmpleados().add(this);
        pruebaFinTx();
    }
    public Empleado subeSueldoEmpleado(Empleado emp, long aumento) {
        pruebaComienzoTx();
        emp.setSueldo(emp.getSueldo + aumento);
        pruebaFinTx();
    }
    public List<Empleado> findEmpleadosSueldoMayorQue(long sueldo) {
        List<Empleado> result = em.createQuery("SELECT e FROM Empleado e " +
            "WHERE e.sueldo > :sueldo").setParameter("sueldo",
            20000).getResultList();
        return result;
    }
    private void pruebaComienzoTx() {
        transaccionActiva = false;
        if (!em.getTransaction().isActive()) {
            em.getTransaction().begin();
            transaccionActiva = true;
        }
    }
    private void pruebaFinTx() {
        if (transaccionActiva) em.getTransaction().commit();
    }
    public String validar(int id, String nombre, long salario) {
        if (id == null) {
            return "Valor requerido";
        }
    }

```

```

    }
    if (nombre == null) {
        return " Valor requerido ";
    }
    if (salario == null) {
        return " Valor requerido ";
    }
    return null;
}
}

```

Nota. Adaptado de “Diseño de un DAO para JPA”, por CCIA, 2010, (<http://www.jtech.ua.es/j2ee/2007-2008/restringido/jpa/sesion06-apuntes.html>). Todos los derechos reservados 2010 por Licenciario. Reproducido con permiso del autor.

3.14.4. Descargas de Documentos Excel

Para las descargas de documentos Excel se ocupará el comando DataExporter, la cual pertenece a la librería PrimeFaces. Por medio de esta se pueden exportar tablas a archivos de tipo XAML, CSV, PDF y XLS.

Una vez que se tenga lista la tabla en la que se tiene la información a exportar, es necesario agregar dependencias para el correcto funcionamiento de la librería, para esto se agregarán las dependencias de la tabla 13 en el archivo “pom.xml” del proyecto.

Tabla 13

Dependencias para usar DataExporter

```

<dependency>
  <groupId>xmlbeans</groupId>
  <artifactId>xmlbeans</artifactId>
  <version>2.3.0</version>
</dependency>
<dependency>
  <groupId>com.lowagie</groupId>
  <artifactId>itext</artifactId>
  <version>2.1.7</version>
</dependency>

```



```
<dependency>
  <groupId>org.apache.poi</groupId>
  <artifactId>poi</artifactId>
  <version>3.13</version>
</dependency>
```

Nota. Adaptado de "PrimeFaces - DataExporter Example", por LogicBig, 2018, (<https://www.logicbig.com/tutorials/misc/primefaces/data-exporter.html>). Todos los derechos reservados 2022 por Licenciatario. Reproducido con permiso del autor.

Una vez agregadas las dependencias, se podrá realizar la exportación. En la tabla 14 se muestra un ejemplo del código para la implementación de las opciones de exportación, en este se muestra cómo se exporta a XML, CSV y XLS.

Tabla 14

Código de ejemplo de descargas de documentos Excel con DataExporter

```
<h:form>
  <p:dataTable id="employeeTable"
    var="employee"
    value="#{employeeBean.employeeList}"
    scrollable="true"
    scrollHeight="300">
    <p:column headerText="Id">
      <h:outputText value="#{employee.id}"/>
    </p:column>
    <p:column headerText="Name">
      <h:outputText value="#{employee.name}"/>
    </p:column>
    <p:column headerText="Phone Number">
      <h:outputText value="#{employee.phoneNumber}"/>
    </p:column>
    <p:column headerText="Address">
      <h:outputText value="#{employee.address}"/>
    </p:column>
  </p:dataTable>
  <br/>
  <p:commandButton value="Export as XML" ajax="false" >
```

```

        <p:dataExporter type="xml" target="employeeTable" fileName="employees"/>
    </p:commandButton>
    <p:commandButton value="Export as CSV" ajax="false" >
        <p:dataExporter type="csv" target="employeeTable" fileName="employees"/>
    </p:commandButton>
    <p:commandButton value="Export Table as Excel" ajax="false" >
        <p:dataExporter type="xls" target="employeeTable" fileName="employees"/>
    </p:commandButton>
    <p:commandButton value="Export Table as PDF" ajax="false" >
        <p:dataExporter type="pdf" target="employeeTable" fileName="employees"/>
    </p:commandButton>
</h:form>

```

Nota. Adaptado de “PrimeFaces - DataExporter Example”, por LogicBig, 2018, (<https://www.logicbig.com/tutorials/misc/primefaces/data-exporter.html>). Todos los derechos reservados 2022 por Licenciatario. Reproducido con permiso del autor.

3.14.5. Descargas de Documentos PDF

En el caso de los informes en PDF se utilizará la librería IText. En la tabla 15 se presenta un ejemplo del código para la creación de dichos reportes, una de las principales ventajas de esta librería es la rapidez de su creación y menos consumo de recursos.

Tabla 15

Código de ejemplo de descargas de documentos PDF con IText

```

import java.io.FileOutputStream;
import java.io.IOException;
import com.lowagie.text.Document;
import com.lowagie.text.DocumentException;
import com.lowagie.text.Paragraph;
import com.lowagie.text.pdf.PdfWriter;
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("http://www.seraphinux.com/");
        Document documento = new Document();
        try {
            PdfWriter.getInstance(documento,

```

```
        new FileOutputStream("seraphinux.pdf"));
        documento.open();
        documento.add(new Paragraph("http://www.seraphinux.com/"));
    } catch (DocumentException de) {
        System.err.println(de.getMessage());
    } catch (IOException ioe) {
        System.err.println(ioe.getMessage());
    }
    documento.close();
}
}
```

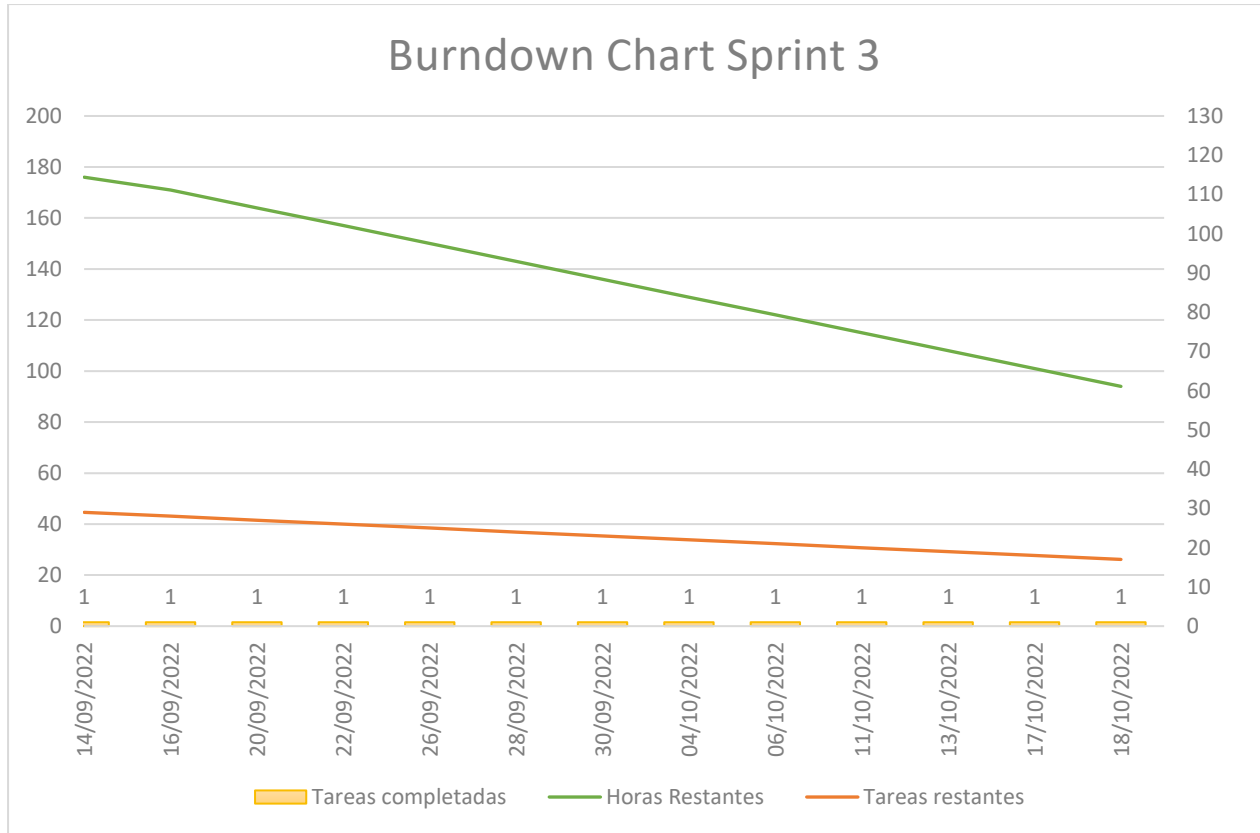
Nota. Adaptado de "IText", por Marco de Desarrollo de la Junta Andalucía, 2018, (<https://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/239>). Todos los derechos reservados 2018 por Licenciario. Reproducido con permiso del autor.

3.15. Burndown Chart Sprint 3

En la figura 69 se muestra el avance del sprint 3, las barras de color amarillo representan las tareas realizadas según la fecha indicada, la línea verde representa las horas restantes del proyecto y la línea naranja las tareas restantes. En este sprint no hubo la necesidad de recorrer tareas para el sprint 4, debido a que todas las tareas se completaron, lo que significa que el proyecto va acorde a lo planeado.

Figura 69

Burndown Chart Sprint 3



3.16. Ejecución Sprint 4

En la cuarta iteración se realizaron las siguientes 16 actividades, para la continuación del desarrollo del sistema.

En la tabla 16 se presentan las tareas del Sprint 4 junto con la fecha programada, estado del desarrollo y el tiempo que se requiere.

Tabla 16

Sprint 4

Sprint 4						
No.	Tarea	Responsable	Fecha	Sprint	Estado	Horas
1	Prueba de Módulo inicio de sesión	Mitzi Sollange Hernández Juárez	20/10/2022	4	Completado	5

Sprint 4						
No.	Tarea	Responsable	Fecha	Sprint	Estado	Horas
2	Prueba permisos para tipos de usuarios	Mitzi Sollange Hernández Juárez	25/10/2022	4	Completado	5
3	Operación CRU en Catálogo de Cajones	Mitzi Sollange Hernández Juárez	27/10/2022	4	Completado	6
4	Operación CRU en Catálogo de Catálogo de calles	Mitzi Sollange Hernández Juárez	27/10/2022	4	Completado	6
5	Operación CRUD en Catálogo de usuarios	Mitzi Sollange Hernández Juárez	28/10/2022	4	Completado	6
6	Operación CRUD en Catálogo de costos tiempo	Mitzi Sollange Hernández Juárez	28/10/2022	4	Completado	6
7	Operación CR en Catálogo de venta saldo	Mitzi Sollange Hernández Juárez	31/10/2022	4	Completado	6
8	Operación CR en Catálogo de revisiones	Mitzi Sollange Hernández Juárez	31/10/2022	4	Completado	6
9	Operación RU en Catálogo de parámetros de configuración	Mitzi Sollange Hernández Juárez	01/11/2022	4	Completado	6
10	Operación R en Catálogo de catálogo de saldo utilizado	Mitzi Sollange Hernández Juárez	01/11/2022	4	Completado	6
11	Operación CRUD en Catálogo de catálogo de vehículos	Mitzi Sollange Hernández Juárez	02/11/2022	4	Completado	6
12	Prueba módulo de generación de reportes	Mitzi Sollange Hernández Juárez	02/11/2022	4	Completado	6
13	Prueba descargas Excel y PDF	Mitzi Sollange Hernández Juárez	03/11/2022	4	Completado	6

Sprint 4						
No.	Tarea	Responsable	Fecha	Sprint	Estado	Horas
14	Mensajes de advertencia y confirmación	Mitzi Sollange Hernández Juárez	04/11/2022	4	Completado	6
15	Formato de reportes	Mitzi Sollange Hernández Juárez	08/11/2022	4	Completado	6
16	Formato de descargas Excel y PDF	Mitzi Sollange Hernández Juárez	10/11/2022	4	Completado	6

3.17. Resultados Obtenidos del Sprint 4

A continuación, se muestran los resultados obtenidos en el sprint 4, los cuales se describieron anteriormente en la tabla 16.

Debido a la carta de confidencialidad (Ver anexo A), no se tiene permitido mostrar la funcionalidad del proyecto, por lo tanto, se mostrarán ejemplos generales ajenos al proyecto sacados de internet.

3.17.1. Mensajes

Para mandar mensajes de advertencia al usuario, se hará uso de la librería PrimeFaces, en la tabla 17 y 18 se presenta un ejemplo del código que permite mostrar mensajes al usuario. En este ejemplo se muestran 4 mensajes diferentes, de información, advertencia y de errores.

Tabla 17

Código de ejemplo de mensajes en sección XHTML

```
<h:form>
  <p:messages id="messages" showDetail="true" closable="true">
    <p:autoUpdate />
  </p:messages>
  <p:commandButton value="Info" action="#{messagesView.info}" />
```

```

    <p:commandButton value="Warn" action="#{messagesView.warn}" />
    <p:commandButton value="Error" action="#{messagesView.error}" />
    <p:commandButton value="Fatal" action="#{messagesView.fatal}" />
</h:form>
<p:separator />
<h:form>
    <h:panelGrid id="grid" columns="3" cellpadding="5">
        <p:outputLabel for="default" value="Default:" />
        <p:inputText id="default" required="true" />
        <p:message for="default" />
        <p:outputLabel for="txt" value="Text:" />
        <p:inputText id="txt" required="true" />
        <p:message for="txt" display="text" />
        <p:outputLabel for="icon" value="Icon:" />
        <p:inputText id="icon" required="true" />
        <p:message for="icon" display="icon" />
        <p:outputLabel for="tt" value="Tooltip:" />
        <p:inputText id="tt" required="true" />
        <p:message for="tt" display="tooltip" />
    </h:panelGrid>
    <p:commandButton id="submitButton" value="Submit" update="grid" />
</h:form>

```

Nota. Adaptado de “Messages”, por PrimeFaces, 2019, (<https://www.primefaces.org/showcase-v8/ui/message/messages.xhtml>). Todos los derechos reservados 2019 por Licenciario. Reproducido con permiso del autor.

Tabla 18

Código de ejemplo de mensajes en sección View

```

@Named
@RequestScoped
public class MessagesView {
    public void info() {
        FacesContext.getCurrentInstance().addMessage(null, new
FacesMessage(FacesMessage.SEVERITY_INFO, "Info", "PrimeFaces Rocks."));
    }
    public void warn() {

```

```

        FacesContext.getCurrentInstance().addMessage(null, new
FacesMessage(FacesMessage.SEVERITY_WARN, "Warning!", "Watch out for PrimeFaces.));
    }
    public void error() {
        FacesContext.getCurrentInstance().addMessage(null, new
FacesMessage(FacesMessage.SEVERITY_ERROR, "Error!", "Contact admin.));
    }
    public void fatal() {
        FacesContext.getCurrentInstance().addMessage(null, new
FacesMessage(FacesMessage.SEVERITY_FATAL, "Fatal!", "System Error"));
    }
}

```

Nota. Adaptado de “Messages”, por PrimeFaces, 2019, (<https://www.primefaces.org/showcase-v8/ui/message/messages.xhtml>). Todos los derechos reservados 2019 por Licenciatario. Reproducido con permiso del autor.

3.17.2. Formato de Reportes

La generación de reportes en el sistema, se realiza con el fin de ver información general, el administrador será el único usuario que podrá generar reportes de toda la información del sistema. De acuerdo al módulo, en el reporte se encontrará la siguiente información:

- Vendedores: Nombre completo, correo electrónico, teléfono y saldo.
- Revisores: Nombre completo, correo electrónico y teléfono.
- Inmovilizadores: Nombre completo, correo electrónico y teléfono.
- Conductores: Nombre completo, correo electrónico, teléfono y sado.
- Costos del sistema: Descripción, minutos y costo.

3.18. Burndown Chart Sprint 4

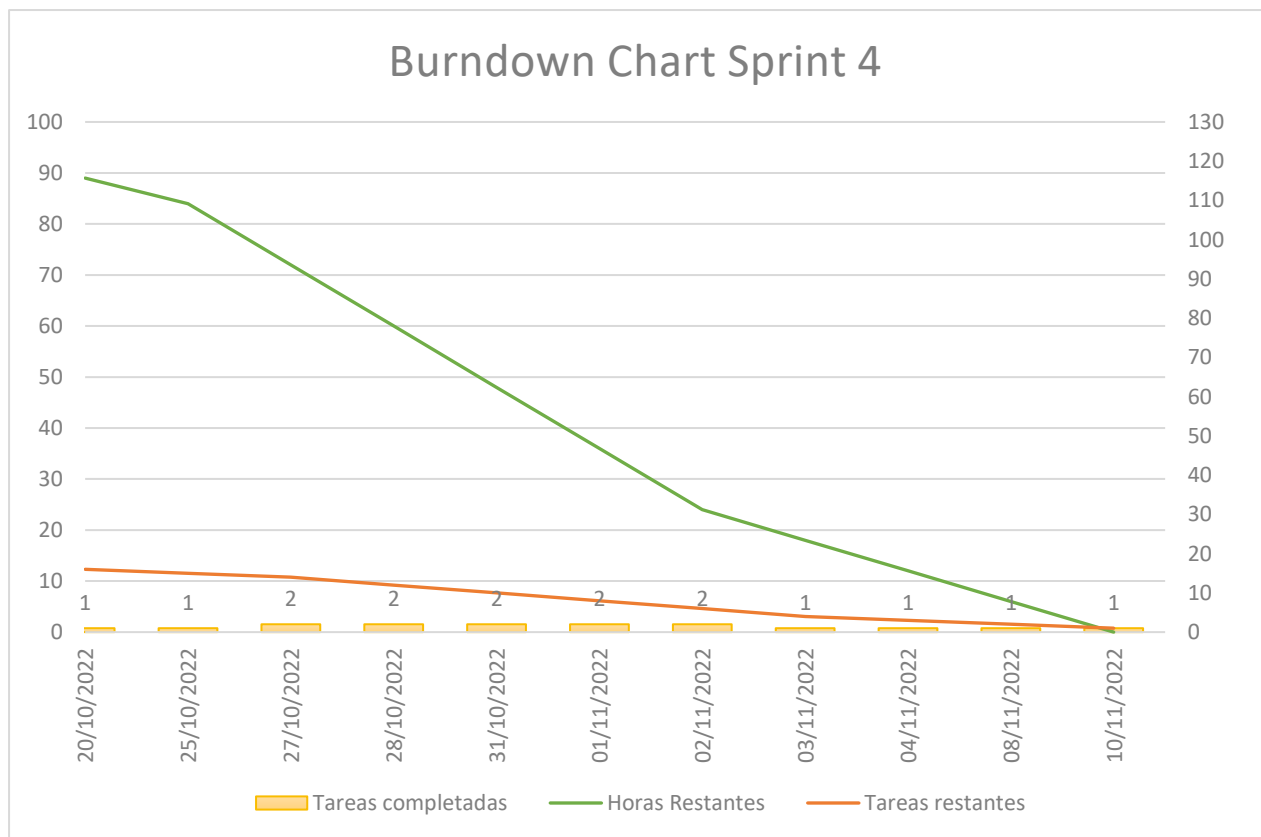
En la figura 70 se muestra el avance del sprint 4, las barras de color amarillo representan la cantidad de tareas realizadas según la fecha indicada, la línea verde representa las horas restantes del proyecto y la línea naranja las tareas restantes.

En este sprint se avanzó con las pruebas a los diferentes módulos y funcionalidades del sistema, como la autenticación de los diferentes usuarios, reportes, mensajes y verificación de los permisos de cada usuario a sus respectivos módulos.

Al final de este sprint se han desarrollado todas las tareas planeadas desde un inicio, cumpliendo en tiempo y forma con cada actividad.

Figura 70

Burndown Chart Sprint 4



CAPÍTULO IV RESULTADOS

4.1. Resultados de Encuesta de Sistema Implementado

Con el fin de medir la opinión de los usuarios al nuevo sistema se ocupará el método Likert, donde el usuario puede estar de acuerdo o no. El propósito será medir el sistema según el punto de vista del conductor, revisor e inmovilizador.

Primero, se enlistarán las preguntas para el conductor:

1. ¿Fue fácil acceder al sistema web Parking?
2. ¿Fue fácil e intuitivo la compra de tiempo de parquímetro?
3. ¿Fue fácil e intuitivo la recarga de tiempo de parquímetro?
4. ¿Fue fácil e intuitivo el pago de multas en el sistema web?
5. ¿La navegación dentro del sistema web es fácil de entender?
6. ¿Se redujo el tiempo de compra de tiempo del parquímetro?
7. ¿Se redujo el tiempo de recarga de tiempo del parquímetro?
8. ¿Se redujeron los tiempos de espera para el retiro de la araña inmovilizadora?

Como segunda encuesta tenemos al revisor, las preguntas para este usuario serán las siguientes:

1. ¿Fue fácil acceder al sistema web Parking?
2. ¿Fue fácil e intuitivo la búsqueda de placas dentro del sistema?
3. ¿Fue fácil e intuitivo la solicitud de inmovilización?
4. ¿La navegación dentro del sistema web es fácil de entender?
5. ¿Se redujo el tiempo de revisión de vehículos?
6. ¿Se redujo el tiempo de solicitar la inmovilización?
7. ¿Consideras que hubo una mejora de tiempos entre el inmovilizador y tú?

Por último, se tienen las preguntas para la encuesta del inmovilizador, son las siguientes:

1. ¿Fue fácil acceder al sistema web Parking?
2. ¿Fue fácil e intuitivo la verificación del pago de multas dentro del sistema?
3. ¿Fue fácil e intuitivo la aceptación de solicitudes de inmovilización?
4. ¿La navegación dentro del sistema web es fácil de entender?
5. ¿Se redujo el tiempo de verificación del pago de multas?
6. ¿Se redujo el tiempo de atención a solicitudes de inmovilización?
7. ¿Consideras que hubo una mejora de tiempos entre el revisor y tú?

Para la evaluación de todas las encuestas se ocupa el método de Likert, el cual tiene una escala de 5 niveles para conocer su experiencia respecto al sistema, siendo los niveles:

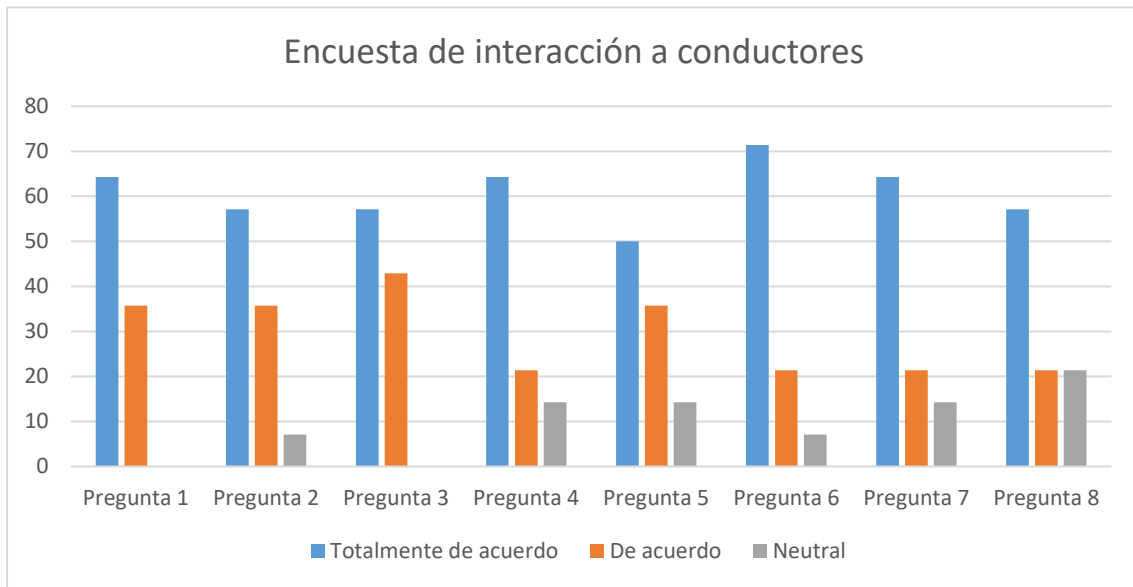
- Totalmente de acuerdo
- De acuerdo
- Neutral
- En desacuerdo
- Totalmente en desacuerdo

Para tener una mejor visión de los resultados de las encuestas se representarán gráficamente por medio de una gráfica de barras.

En la figura 71 se presentan los resultados de las encuestas realizadas para los conductores, en esta se puede observar que las opciones que tuvieron más votos fueron “Totalmente de acuerdo” y “De acuerdo”, esto nos muestra que el conductor se encuentra satisfecho con este nuevo sistema, mejorando procesos para reducir sus tiempos.

Figura 71

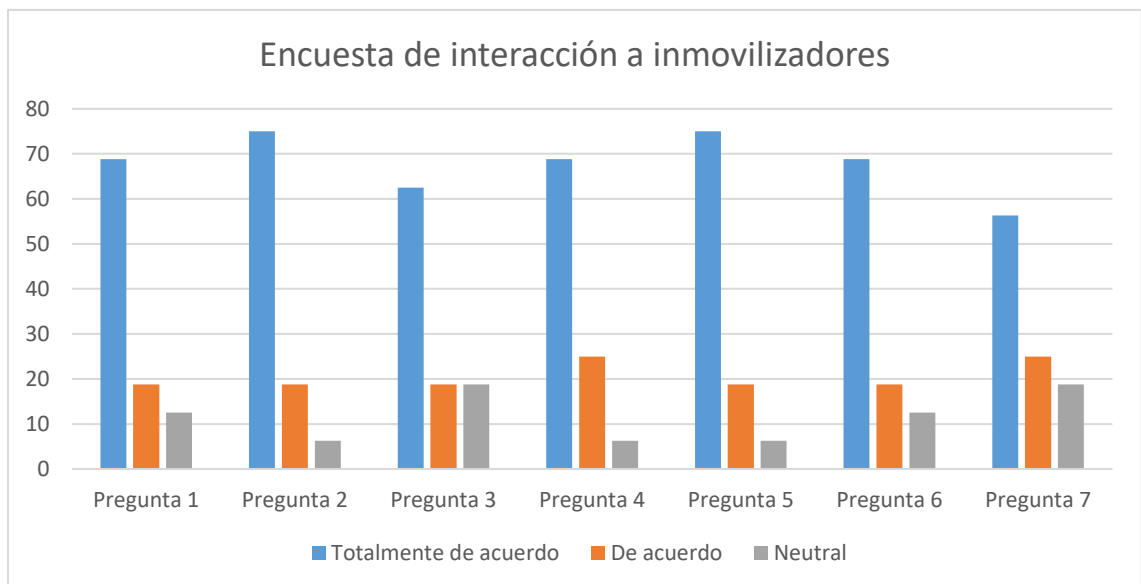
Resultados de encuesta a conductores



En la figura 72 se presentan los resultados de las encuestas realizadas para los inmovilizadores, en esta se puede observar que el resultado que tuvo más votos fue la opción “Totalmente de acuerdo”, indicando la total conformidad de los inmovilizadores respecto al sistema.

Figura 72

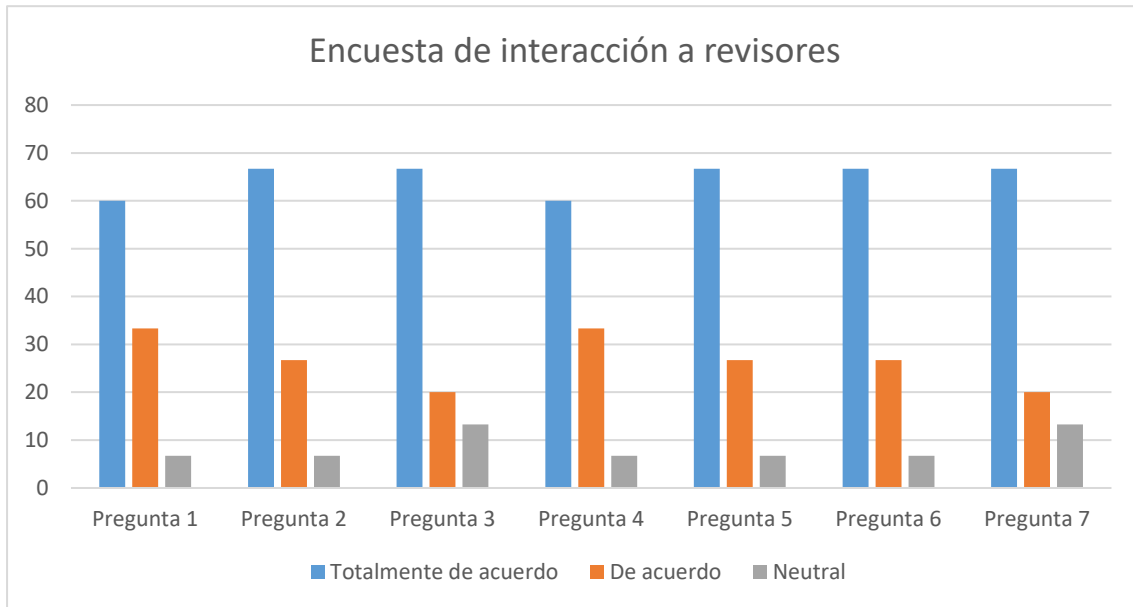
Resultados de encuesta a inmovilizadores



En la figura 73 se presentan los resultados de las encuestas realizadas para los revisores, en esta se puede observar que el resultado que tuvo más votos fue la opción “Totalmente de acuerdo”, indicando la total conformidad de los revisores respecto al sistema.

Figura 73

Resultados de encuesta a revisores



4.2. Comparación de Tiempos Entre Sistemas

Se realizaron pruebas en las diferentes actividades de los conductores, revisores e inmovilizadores, con el fin de comparar el tiempo entre el sistema actual y este nuevo sistema, para esto se crearon diferentes tablas en las cuales cada una engloba todas las tareas realizadas por ese usuario. Por último, se calculó el porcentaje de tiempo optimizado, esta información ayudará a comprobar la hipótesis.

En primer lugar, tenemos al conductor, los procesos a evaluar es la compra y recarga se tiempo, pago de multas y el retiro de araña. Se tiene en cuenta un total de 10 iteraciones para cada tarea.

Tabla 19

Tiempos de sistemas para conductor

ANÁLISIS DE TIEMPO PARA EL CONDUCTOR					
Proceso	Iteración	Tiempo en minutos/ sistema anterior	Tiempo en minutos/ sistema nuevo	Diferencia de tiempo en minutos	Porcentaje de tiempo optimizado
Compra de tiempo	1	5	2.5	2.5	50 %
	2	3.5	1.5	2	57.1 %
	3	3	2.5	0.5	16.7 %
	4	2.5	1.5	1	40 %
	5	4	2	2	50 %
	6	6	0.5	5.5	91.7 %
	7	5.5	2	3.5	63.6 %
	8	4	2.5	1.5	37.5 %
	9	6.5	3	3.5	53.8 %
	10	2	2	0	0.0 %
Recarga de tiempo	1	5	1	4	80 %
	2	3	1.5	1.5	50 %
	3	6	2	4	66.7 %
	4	3.5	2	1.5	42.9 %
	5	5.5	1.5	4	72.7 %
	6	4	3	1	25 %
	7	2.5	2	0.5	20 %
	8	5	2	3	60 %
	9	3.5	1	2.5	71.4 %
	10	5.5	2.5	3	54.5 %
Pago de multas	1	6	3	3	50 %
	2	10	2.5	7.5	75 %
	3	18	2	16	88.9 %
	4	9	1	8	88.9 %
	5	7	1.5	5.5	78.6 %
	6	8.5	2	6.5	76.5 %
	7	10	3	7	70 %
	8	8	2.5	5.5	68.8 %
	9	15	2	13	86.7 %
	10	16	3	13	81.3 %
Retiro de araña	1	20	5	15	75 %
	2	13	8	5	38.5 %
	3	5	3	2	40 %
	4	10	6	4	40 %
	5	8	6	2	25 %
	6	9	5	4	44.4 %

ANÁLISIS DE TIEMPO PARA EL CONDUCTOR					
Proceso	Iteración	Tiempo en minutos/ sistema anterior	Tiempo en minutos/ sistema nuevo	Diferencia de tiempo en minutos	Porcentaje de tiempo optimizado
	7	16	9	7	43.8 %
	8	11	5	6	54.5 %
	9	18	6	12	66.7 %
	10	12	7	5	41.7 %

Una vez se terminaron las iteraciones de todas las tareas, se sacó un promedio del tiempo y porcentajes de cada una de estas. En total se tiene una mejora del 55.9% de tiempo respecto a las actividades del conductor.

Tabla 20

Tiempo promedio de sistemas para conductor

ANÁLISIS DE TIEMPO PROMEDIO PARA EL CONDUCTOR			
Proceso	Tiempo promedio en minutos/ sistema anterior	Tiempo promedio en minutos/ sistema nuevo	Porcentaje promedio de tiempo optimizado
Compra de tiempo	4.2	2	46.0
Recarga de tiempo	4.35	1.85	54.3
Pago de multas	10.75	2.25	76.4
Retiro de araña	12.2	6	47.0
Promedio optimización de tiempo			55.9

El segundo usuario el cual realizará pruebas es el revisor, este es el encargado de verificar los tiempos de los vehículos, por lo tanto, una de sus tareas es la búsqueda de placas en el sistema, dependiente el estatus puede solicitar su inmovilización. Se tienen en cuenta 10 iteraciones donde se comparan los tiempos del sistema actual y el nuevo sistema.

Tabla 21

Tiempos de sistemas para revisor

ANÁLISIS DE TIEMPO PARA EL REVISOR					
Proceso	Iteración	Tiempo en minutos/ sistema anterior	Tiempo en minutos/ sistema nuevo	Diferencia de tiempo en minutos	Porcentaje de tiempo optimizado
Tiempo de respuesta al ingresar la placa	1	0.5	0.2	0.3	60.0
	2	0.5	0.1	0.4	80.0
	3	0.6	0.5	0.1	16.7
	4	0.4	0.3	0.1	25.0
	5	0.6	0.6	0	0.0
	6	1	0.5	0.5	50.0
	7	0.4	0.2	0.2	50.0
	8	0.3	0.3	0	0.0
	9	1	0.5	0.5	50.0
	10	0.6	0.6	0	0.0
Tiempo de solicitud de inmovilización	1	1	1	0	0.0
	2	3	0.5	2.5	83.3
	3	2.5	1.5	1	40.0
	4	1.5	1.5	0	0.0
	5	3.5	2	1.5	42.9
	6	5	2	3	60.0
	7	4	2	2	50.0
	8	2.5	1.5	1	40.0
	9	1	1	0	0.0
	10	2	0.5	1.5	75.0
Tiempo de respuesta de aceptación de solicitud	1	2	1	1	50.0
	2	3	2	1	33.3
	3	2	1.5	0.5	25.0
	4	1.5	1.5	0	0.0
	5	3	2.5	0.5	16.7
	6	5	3	2	40.0
	7	2	2	0	0.0
	8	3.5	1	2.5	71.4
	9	4	2	2	50.0
	10	2.5	1.5	1	40.0

Para resumen de los tiempos y la optimización de las pruebas, se toma el tiempo y porcentajes promedio de las tareas. Con la información se demuestra que se el revisor optimiza los tiempos de sus tareas en un 35%.

Tabla 22

Tiempo promedio de sistemas para revisor

ANÁLISIS DE TIEMPO PROMEDIO PARA EL REVISOR			
Proceso	Tiempo promedio en minutos/ sistema anterior	Tiempo promedio en minutos/ sistema nuevo	Porcentaje promedio de tiempo optimizado
Tiempo de respuesta al ingresar la placa	0.59	0.38	33.2
Tiempo de solicitud de inmovilización	2.6	1.35	39.1
Tiempo de respuesta de aceptación de solicitud	2.85	1.8	32.6
Promedio optimización de tiempo			35.0

Por último, se tiene al usuario inmovilizador, la tarea a realizar pruebas es la verificación de multas, se realizan 10 iteraciones y se anotan los tiempos del sistema anterior y el nuevo, finalmente se calcula el porcentaje de optimización.

Tabla 23

Tiempos de sistemas para inmovilizador

ANÁLISIS DE TIEMPO PARA EL INMOVILIZADOR					
Proceso	Iteración	Tiempo en minutos/ sistema anterior	Tiempo en minutos/ sistema nuevo	Diferencia de tiempo en minutos	Porcentaje de tiempo optimizado
Verificación de multa	1	3	2	1	33.3
	2	3.5	1	2.5	71.4
	3	2	1.5	0.5	25.0
	4	5.5	2	3.5	63.6
	5	4	2.5	1.5	37.5
	6	3	2	1	33.3

	7	4.5	2.5	2	44.4
	8	2	1.5	0.5	25.0
	9	2.5	1	1.5	60.0
	10	5	3	2	40.0

Para resumir la información de la tabla pasada, se calculan sus promedios en la tabla 24, en esta se calcula el porcentaje total de tiempo para el usuario, siendo el 43.4%.

Tabla 24

Tiempo promedio de sistemas para inmovilizador

ANÁLISIS DE TIEMPO PROMEDIO PARA EL INMOVILIZADOR			
Proceso	Tiempo promedio en minutos/ sistema anterior	Tiempo promedio en minutos/ sistema nuevo	Porcentaje promedio de tiempo optimizado
Verificación de multa	3.5	1.9	43.4
Promedio optimización de tiempo			43.4

4.3. Verificación de Hipótesis

Respecto a los tiempos de ejecución de tareas del conductor, se optimizan los tiempos en un 55.9%, el cual abarca los tiempos de compra, mejorando un 46% y los tiempos de recarga, que se mejora en un 54.3%, dando un promedio de 50.1%. Por otro lado, se analiza los tiempos de espera del conductor, el cual mejora un 47% de tiempo.

En base a la hipótesis planteada al inicio, se pronostica una mejora del 50% en los tiempos de compra y venta, de acuerdo a los estudios realizados, está comprobado se mejora en un 50.1%, quedando ligeramente arriba de lo planteado.

Finalmente, se planteó una mejora del 30% en los tiempos de espera del conductor, por medio de los estudios realizados se mejoró en un 47%, quedando por arriba de la hipótesis planteada.

Con esto se concluye que el sistema realizado cumple con la hipótesis planteada al inicio del documento, mejorando los procesos, el tiempo de respuesta y optimizando tiempos, con el fin de que este sea un buen servicio para todos los usuarios de este sistema.

4.4. Respuesta a Preguntas de Investigación

¿Cómo mejorar y optimizar el proceso para la compra y revisión de tiempo en parquímetros e inmovilización de vehículos del municipio de Tlaxcala de Xicohtécatl? El evitar que el usuario se dirija a las casetas de los parquímetros reduce un tiempo significativo a la hora de usarlos, al ingresar al sitio web puede realizar la compra de su tiempo al instante que se estaciona. Por otro lado, en la revisión de los vehículos, el solo ingresar la placa del vehículo y que despliegue el estatus actual de este, hace que el proceso de inmovilización sea más ágil, con lo cual se reduce el tiempo de espera del conductor que es lo que se busca.

¿Qué tan factible es utilizar una aplicación web para el control de parquímetros en comparación del proceso actual? De acuerdo a la experiencia de los diferentes usuarios, estaban totalmente de acuerdo que era fácil acceder al sistema web, negando complicaciones a la hora de navegar a través de este y de poder realizar sus actividades con éxito.

CAPÍTULO V DISCUSIÓN

5.1. Resultados Obtenidos & Esperados

Recapitulando, el objetivo general del proyecto es desarrollar un sistema web de parquímetros Parking en el municipio de Tlaxcala de Xicohtécatl, utilizando Java EE, para la mejora del proceso de compra y venta de tiempo, pago de multas, revisión del tiempo, solicitud e inmovilización del vehículo, así como la generación de reportes.

El objetivo antes planteado se logrará cumplir por medio del desarrollo de los objetivos específicos, los cuales fueron descritos al inicio del documento. Por medio de los requerimientos recabados por la empresa SmartSoft America Business Applications S.A. de C.V., se logrará identificar las necesidades del cliente y a su vez los objetivos del proyecto.

Una vez identificados las necesidades del cliente se establecerá una metodología de desarrollo, en este caso será la metodología ágil Scrum. El primer paso será realizar el product backlog, donde se definirán y dividirán las tareas de acuerdo a los objetivos, como el desarrollo de interfaces de baja y alta fidelidad, desarrollo de módulos del sistema, por otro lado, se agregarán las fechas en las que se llevará acabo cada tarea y el encargado del desarrollo de estas.

Una vez definido el product backlog, se logrará identificar la cantidad de sprint que se ejecutarán a lo largo del proyecto, así como fechas y tiempos de desarrollo. A lo largo de estos se deberá contar con una óptima organización para cumplir adecuadamente con los plazos de entrega y terminar el proyecto cumpliendo con los objetivos planteados.

Al término de todos los sprints, se espera que se cumpla el objetivo general del proyecto, esto por medio de los objetivos específicos que a su vez fueron plasmados en el product backlog.

Por último, el porcentaje de optimización en los tiempos de compra y recarga de tiempo por parte del conductor se espera mejorar en un 50%, el tiempo de espera se mejora espera una mejora del 30%.

Una vez tras realizar las pruebas en el nuevo sistema, se observa que se cumple con la hipótesis planteada con una estimación del 50% en el tiempo de compra y recarga de tiempo y un 47% en el tiempo de mejora de espera, superando un 17% respecto al tiempo de espera.

5.2. Conclusiones

El uso de una metodología ágil, en este caso Scrum, dentro de un ambiente profesional, fue de suma importancia, ya que con esta se buscaba crear y desarrollar una solución tecnológica de un sistema web de parquímetros, para el municipio de Tlaxcala de Xicohtécatl.

Por medio de cada iteración de sprint, el sistema ha cumplido con el objetivo plantado al inicio del proyecto, cumpliendo las necesidades del cliente y a su vez con los objetivos específicos.

El sistema web permite que los diferentes usuarios, como el administrador, revisor, vendedor, inmovilizador y conductor, logren tomar su rol de forma óptima, cumpliendo sus tareas correspondientes sin necesidad de empalmar su trabajo como se hacía anteriormente.

La generación de reportes por parte del administrador, logra tener un mejor control y consulta de información tanto de los trabajadores como el cliente, donde se visualiza información básica como su nombre completo, correo electrónico y número telefónico.

Por otro lado, mejora la comunicación entre el revisor y el inmovilizador, logrando que se agregue la araña inmovilizadora al conductor en caso de que se requiera, también se mejora el tiempo del pago de multa y el retiro de la araña antes mencionada.

Por último, mejora la experiencia del conductor para el pago de tiempo, tiempo extra y el pago de multas, siendo permitido todo esto por medio de un dispositivo a su alcance, a comparación de antes que se tenía que dirigir a la caseta de parquímetros de la calle.

De esta forma se concluye con el presente proyecto, obteniendo un amplio conocimiento en el manejo de librerías, herramientas, lenguajes de programación y metodologías de desarrollo.

5.3. Recomendaciones Y Aportación

Los resultados del proyecto cumplen con los objetivos planteados al inicio del documento, sin embargo, tras el desarrollo de cada sprint se identificaron aspectos por mejorar en el sistema.

En la sección del conductor, este puede visualizar la información de todos los pagos de tiempos y multas, sin embargo, no cuenta con un tiempo de filtrado, es decir, el conductor no puede realizar una consulta de sus pagos por un intervalo de fechas, tipo de pago o vehículo.

En la sección del revisor, solo es mostrado un buscador, en este solo se ingresa la placa del vehículo para saber su estatus. Al inicio se podría agregar una consulta por calle, permitiendo ver los cajones disponibles en esta y en caso de estar ocupados mostrar la información del vehículo que hace uso de ese cajón.

En la sección del inmovilizador, solo visualiza una tabla donde se presentan las solicitudes realizadas por el revisor y cuenta con un histórico, dentro de este se presenta todas las inmovilizaciones que ha realizado, sin embargo, no cuenta con un filtrado, es decir, no puede ver la información por vehículos que no han pagado multa, búsquedas por un rango de fechas entre otros.

Referencias Bibliográficas

Blinkay (2023). Blinkay paso a paso. Blinkay. <https://www.blinkay.app/es/how-it-works/>

Cuéntame INEGI (2020). Número de habitantes. <https://cuentame.inegi.org.mx/monografias/informacion/tlax/poblacion/default.aspx?tema=me&e=29>

devCamp (2020). Qué es una librería en programación. devCamp. <https://devcamp.es/que-es-libreria-programacion/>

Digité (s.f.). What Is Scrum Methodology? & Scrum Project Management. Digité. <https://www.digite.com/agile/Scrum-methodology/>

GeeksforGeeks (9 de noviembre de 2022). SQL | Comandos DDL, DQL, DML, DCL y TCL. GeeksforGeeks. <https://www.geeksforgeeks.org/sql-ddl-dql-dml-dcl-tcl-commands/>

GeeksforGeeks (4 de julio de 2022). XHTML | Introduction. GeeksforGeeks. <https://www.geeksforgeeks.org/xhtml-introduction/>

Genos (2022). Soporte GlassFish. Genos. <https://genos.es/glassfish/>

Gustavo B. (2 de marzo de 2022). ¿Qué es AJAX y cómo funciona? Hostinger Tutoriales. <https://www.hostinger.mx/tutoriales/que-es-ajax>

Hernández, J., Molina, J., & Gómez, A. (2018). Análisis de la regulación del estacionamiento en la vía pública mediante parquímetros. Revista de Estudios Urbanos, 33(2), 57-72.

IBM (27 de septiembre de 2022). Java Persistence API (JPA). IBM. <https://www.ibm.com/docs/es/was-liberty/nd?topic=overview-java-persistence-api-jpa>

INEGI (2021). Vehículos de motor registrados en circulación. INEGI.
https://www.inegi.org.mx/sistemas/olap/consulta/general_ver4/MDXQueryDatos.asp?#Regreso&c=

iParkMe (31 de marzo de 2023). iParkMe parquímetro y parking. Aptoide.
<https://iparkme.mx.aptoide.com/app>

Loshin P. (febrero de 2022). Structured Query Language (SQL). TechTarget.
<https://www.techtarget.com/searchdatamanagement/definition/SQL>

MDN web docs (21 de octubre de 2022). CSS. MDN web docs.
<https://developer.mozilla.org/es/docs/Web/CSS>

MDN web docs (20 de septiembre de 2022). Developer Tools. MDN web docs.
https://developer.mozilla.org/en-US/docs/Glossary/Developer_Tools

MDN web docs (20 de septiembre de 2022). Google Chrome. MDN web docs.
https://developer.mozilla.org/en-US/docs/Glossary/Google_Chrome

Montiel A. (13 de julio de 2020). Reanudarán servicio de parquímetros en la capital. Tlaxcala digital. <http://www.tlaxcaladigital.com/2020/07/13/reanudaran-servicio-de-parquimetros-en-la-capital/>

Morales, A., López, D., & Pérez, E. (2017). Análisis del efecto disuasorio de la araña inmovilizadora en el estacionamiento ilegal. Revista de Seguridad Vial, 21(3), 27-42.

Mueve ciudad (2020). ¿Cómo funciona? Mueve ciudad.
<https://www.mueveciudad.com/ComoFunciona>

MySQL (2022). MySQL Workbench. MySQL.
<https://www.mysql.com/products/workbench/>

ORACLE (2020). Java Programming Language. ORACLE.
<https://docs.oracle.com/javase/7/docs/technotes/guides/language/>

ORACLE (2022). NetBeans IDE. ORACLE.
<https://www.oracle.com/mx/tools/technologies/netbeans-ide.html>

ORACLE (2022). What Is a Database? ORACLE. <https://www.oracle.com/database/what-is-database/>

Parkimovil (2020). Soluciones. Parkimovil. <https://parkimovil.com/soluciones-parkimovil.html>

Parkum (2019). Sobre Parkum. Parkum. <https://parkum.mx/sobre-parkum/>

Peek S. (29 de junio de 2022). What Is Agile Scrum Methodology? Business News Daily. <https://www.businessnewsdaily.com/4987-what-is-agile-Scrum-methodology.html>

Pedrosa S. (19 de octubre de 2020). Economipedia. Pago. <https://economipedia.com/definiciones/pago.html>

Rus E. (1 de noviembre de 2020). Economipedia. Tipos de investigación. <https://economipedia.com/definiciones/tipos-de-investigacion.html>

Souza I. (14 de junio de 2019). ¿Qué es un servidor web y para qué sirve en Internet? Rockcontent. <https://rockcontent.com/es/blog/que-es-un-servidor/>

Techopedia (11 de julio de 2022). Data Definition Language (DDL). Techopedia. <https://www.techopedia.com/definition/1175/data-definition-language-ddl>

Tutorialspoint (2022). JPA – Arquitectura. Tutorialspoint. https://www.tutorialspoint.com/es/jpa/jpa_architecture.htm#

Vogel L. (29 de julio de 2021). Java persistence API – Tutorial. Vogel/a. <https://www.vogella.com/tutorials/JavaPersistenceAPI/article.html>

Yáñez J. (2022). PRIMEFACES. Código Xules. <https://codigoxules.org/primefaces/>

GLOSARIO

A

AJAX

Acrónimo del inglés *Asynchronous JavaScript and XML*, son tecnologías empleadas para el desarrollo para páginas web asíncronas.

API

Acrónimo de *Interfaz de Programación de Aplicaciones*, el cual permite la comunicación de dos componentes de software.

B

Base de datos

Almacena datos de forma estructurada en tablas, con el fin de que la información pueda ser consultada, eliminada o editada.

C

Consulta (Query)

Accede a la información guardada previamente en una o varias tablas dentro de la base de datos.

CSS

Acrónimo del inglés *Cascading Style Sheets* que significa hojas de estilo en cascada. Se encarga del estilo y diseño de la página web.

CSV

Acrónimo del inglés *Comma Separated Values*, es un tipo de archivo que permite representar información en forma de tabla, donde los campos son separados por comas.

D

DAO

Acrónimo del inglés Data Access Object, se encarga de manejar la conexión entre la aplicación y la base de datos, creando un objeto para encapsular la información.

DCL

Acrónimo del inglés Data Control Language o en español Lenguaje de Control de Datos, controla el acceso del usuario a la base de datos y su contenido.

DDL

Acrónimo del inglés Data Definition Language o en español Lenguaje De Definición De Datos, describe los datos y su relación en una base de datos.

DML

Acrónimo del inglés Data Manipulation Language o en español Lenguaje de Manipulación de Datos, son instrucciones para insertar, borrar, modificar y consultar datos de una base de datos.

DTD

Acrónimo del inglés Document Type Definition o en español Definición De Tipo De Documento, el cual define la estructura de un documento XML o SGML.

DQL

Acrónimo del inglés Data Query Language o en español lenguaje de Consulta De Datos, recupera información almacenada en la base de datos.

E

Excel

Hoja de cálculo para manipular datos y analizar información.

F

Framework

Es un marco de trabajo que proporciona herramientas ya desarrolladas que agilizan el desarrollo de un proyecto.

G

H

HTML

Acrónimo del inglés HyperText Markup Language o en español Lenguaje de Marcas de Hipertexto, es usado para dar estructura una página web.

HTTP

Acrónimo del inglés Hypertext Transfer Protocol, es un protocolo que permite la transferencia de información a través de internet.

I

IDE

Acrónimo del inglés Integrated Development Environment, es una aplicación para el desarrollo de un proyecto de software.

J

JAR

Acrónimo del inglés Java ARchive, es un tipo de archivo comprimido que almacena diferentes archivos de Java.

JPA

Acrónimo del inglés Java Persistence API, es un modelo de programación de persistencia.

JSF

JavaServer Faces Es un framework del lado del servidor para aplicaciones web basadas en Java.

K

L

M

MDB

Manejador de Base de Datos o Sistema Gestor De Base De Datos (SGBD), es un programa que permite la gestión de una base de datos.

Metodología

Técnicas o herramientas que son utilizadas para llevar a cabo una investigación o estudio.

Menú

Es un conjunto de opciones que permiten al usuario la navegación dentro del proyecto.

N

Navegador web

Permite al usuario visualizar el contenido de una página web, este interpreta el lenguaje de marcado y permite al usuario interactuar con su contenido.

Ñ

O

P

PDF

Acrónimo del inglés Portable Document Format o en español Formato De Documento Portátil, es un archivo usado para mostrar información, independientemente del hardware, software o sistema operativo.

Q

R

RFT

Acrónimo del inglés Revisable Form Text, es un tipo de archivo de textos y gráficos.

ROI

Acrónimo del inglés Return On Investment o en español Retorno De La Inversión, se trata del dinero obtenido por la implementación de un software.

S

Sistema

Conjunta las partes del hardware, software y usuario que almacena y procesa información.

T

TCL

Acrónimo del inglés Transaction Control Language o en español Lenguaje de Control Transaccional, son comandos que manejan las transacciones dentro de una base de datos relacional.

U

V

W

Web

Acrónimo del inglés World Wide Web, funciona por medio de internet y se transfiere diferentes tipos de datos por medio de HTTP.

X

XHTML

Acrónimo del inglés eXtensible HyperText Markup Language, es un archivo que reformula a HTML usando el lenguaje de marcado XML.

XLS

Es la extensión de un archivo de las hojas de cálculo de Microsoft Excel.

XML

Acrónimo del inglés Extensible Markup Language es un lenguaje de marcado que sirve para la codificación de documentos.

XSLT

Acrónimo del inglés eXtensible Stylesheet Language for Transformations, es un lenguaje que permite la transformación de XML a XML, HTML o un archivo de texto plano.

Y

Z

ANEXO A CONVENIO DE CONFIDENCIALIDAD

Santa Ana Chiautempan, Tlax., 25 de Agosto del 2022

Asunto: Convenio de Confidencialidad

SMARTSOFT AMERICA BUSINESS APPLICATIONS S.A. de C.V.
M.C. OSCAR ATRIANO PONCE
DIRECCIÓN GENERAL

PRESENTE

El que suscribe **Ing. Mitzi Sollange Hernández Juárez** bajo protesta de decir verdad, como residente profesional del *Instituto Tecnológico Superior de San Martín* en la empresa **SmartSoft America Business Applications S.A. de C.V.** con un periodo inicial a partir del 16 de julio al 16 noviembre del 2022, me permito manifestar a usted mi compromiso profesional y personal para obligarme antes, durante y al término de mi residencia profesional, obligándome a no utilizar, disponer, copiar, transmitir y/o divulgar la información proporcionada a la que tenga acceso, ni la forma y procedimientos de operación, ni los datos y resultados obtenidos, para fines diferentes a los establecidos por la empresa, ya sea a través de publicaciones, conferencias, informaciones ó de cualquier otra forma o medio, sin la autorización expresa y por escrito de **SmartSoft America Business Applications S.A. de C.V. por un periodo de tiempo de 5 años posterior a la fecha de la presente.**

SmartSoft America Business Applications S.A. de C.V. podrá ejercer las acciones penales que se deriven de la violación del contenido de esta carta, en cualquier tiempo, sin perjuicio de las sanciones administrativas ó civiles a las que haya lugar.

ATENTAMENTE

Recibi Original

25 / 08 / 22



Ing. Mitzi Sollange Hernández Juárez

Santa Ana Chiautempan, Tlax., 08 de Septiembre del 2022

Asunto: Carta autorización uso de información

SMARTSOFT AMERICA BUSINESS APPLICATIONS S.A. de C.V.
M.C. OSCAR ATRIANO PONCE
DIRECCIÓN GENERAL

PRESENTE

Por medio de la presente se da respuesta a la solicitud de no inconveniencia para utilización de información recibida con fecha del 08 de septiembre del 2022 donde la interesada **C. Mitzi Sollange Hernández Juárez** quien participa en el proyecto "**Sistema Web de Parquímetros Parking para el Estado de Tlaxcala**" bajo esquema de Residencias Profesionales de la Ingeniería en Sistemas Computacionales, con N° de Control **18240013** de acuerdo a la naturaleza de la información y el contrato de confidencialidad (**NDA**) convenida con el Cliente **SmartSoft America Business Applications S.A. de C.V.** se extiende la autorización bajo los siguientes términos:

No se autoriza mostrar, publicar, divulgar o mencionar en medios impresos, electrónicos o de cualquier índole; Diagramas, Documentos del diseño, Algoritmos, Esquemas de Base de datos, Código Fuente y Pruebas unitarias y otros artefactos de dicho Proyecto.

Se le autoriza mencionar la metodología utilizada, herramientas y versiones usadas y hasta 6 capturas de pantalla del sistema de dicho Proyecto.

SmartSoft America Business Applications S.A. de C.V. podrá ejercer las acciones penales que se deriven de la violación del contenido de esta carta, en cualquier tiempo, sin perjuicio de las sanciones administrativas ó civiles a las que haya lugar.

A petición de la/el interesada/o y para fines de Residencia Profesional exclusivamente; se extiende la presente carta a los 08 días del mes de septiembre del año en curso.

ATENTAMENTE



SmartSoft America Business Applications S.A. de C.V.
Calle Adolfo López Mateos 131 A. Texcacoac
Chiautempan, Tlaxcala, México C.P. 90806
RFC: SAB090522FB7
Tel: +52 (246) 117 5228
www.smartsoftamerica.com.mx

LIC. Leslie Allison Rodríguez Rivera
Responsable de Recursos Humanos y Trabajo
SmartSoft America Business Applications S.A. de C.V.