



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN

“Desarrollo de un sistema web de comunicación para gasolineras mediante WebSockets en tiempo real en Gasintec S.A.”

Autor:

Juan José Pérez Jerez

Tutor:

Ing. Galárraga Hurtado, Juan Fernando

SANGOLQUÍ, AGOSTO 2022



*“El Software es una gran combinación
entre el arte y la ingeniería”*

Bill Gates



Agenda

- Introducción
- Definición del problema
- Justificación
- Objetivos
 - Objetivos Generales
 - Objetivos Específicos
- Marco Teórico
- Desarrollo
- Resultados
- Conclusiones
- Recomendaciones



Introducción

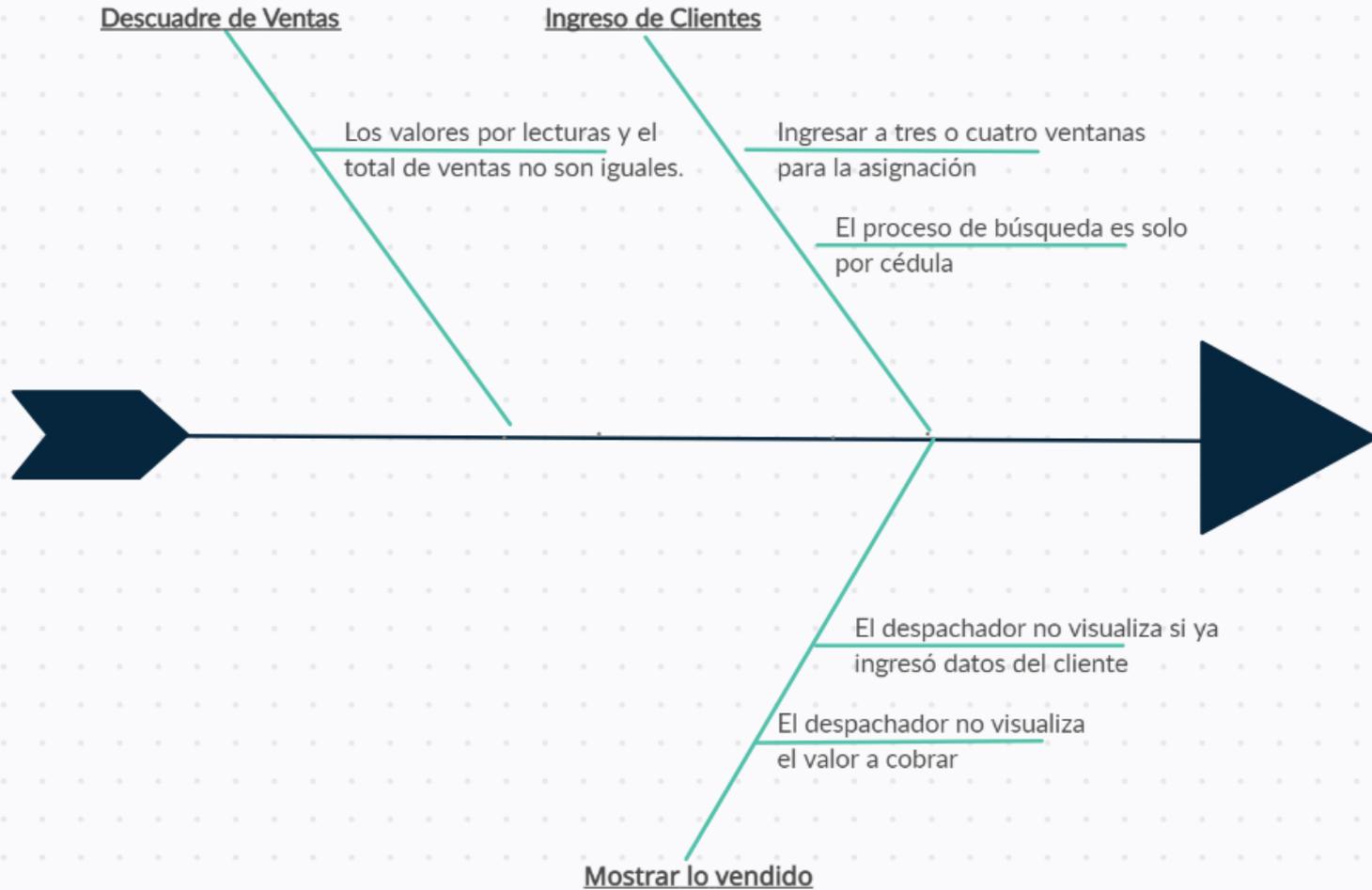
El presente trabajo de investigación realiza un estudio del software en tiempo real con la utilización del protocolo websocket en un sistema desarrollado para el control de ventas en una estación de servicio.

Para determinar la mejor herramienta se realizó una comparación de distintas tecnologías como lo son Poll y WebSockets.

Para el manejo de Poll se requiere intervalos definidos para su ejecución con el limitante de la latencia al presentar la información al usuario. El protocolo websocket es más seguro y no existe limitación alguna por el tiempo, la latencia que existe es aceptable gracias a la API Fetch de JavaScript se logró un control del manejo de memoria.



Definición del problema



(Pérez,2022)



Justificación

La empresa Gasintec S.A. justifica la necesidad de realizar un sistema web de comunicación para gasolineras mediante WebSockets en tiempo real, con el objetivo de tener un mejor control del proceso de venta de combustibles y proporcionando al usuario (despachador) una herramienta amigable e intuitiva que mejore el tiempo de venta.



Objetivo General

- Desarrollar un sistema web con comunicación WebSocket en tiempo real para el despacho de combustible en Estaciones de Servicio en Gasintec S.A.

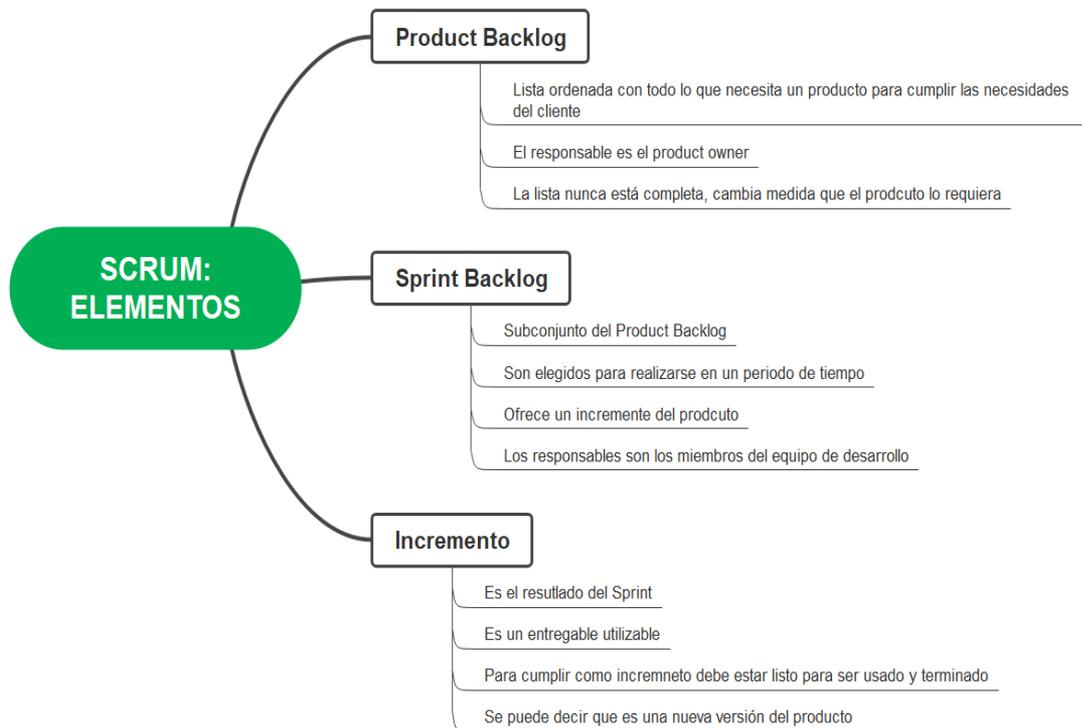
Objetivos Específicos

- Realizar una revisión bibliográfica de la tecnología a utilizar para emplearlo en el desarrollo del sistema.
- Analizar y desarrollar los módulos de conexión WebSocket e ingreso de información de clientes.
- Verificar el funcionamiento del sistema en tiempo real y análisis de resultados.



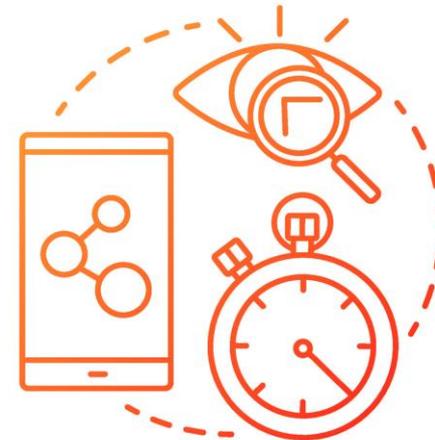
Marco Teórico

Metodología Scrum: Es un proceso ágil y ligero para gestionar y controlar el desarrollo de software. El desarrollo es iterativo e incremental. Cada ciclo o iteración finaliza con un software ejecutable que contiene una nueva funcionalidad.



Tiempo Real

- La web en tiempo real es una red web que utiliza tecnologías y prácticas que permiten a los usuarios recibir información tan pronto como es publicada por sus autores, en lugar de exigir que ellos o su software comprueben una fuente periódicamente para actualizaciones.
- Existen dos tipos, tiempo real duro y suave.
- Tiene cinco características:
 - Determinismo
 - Responsividad
 - Usuarios controladores
 - Confiabilidad
 - Operación a pruebas de fallas duras.



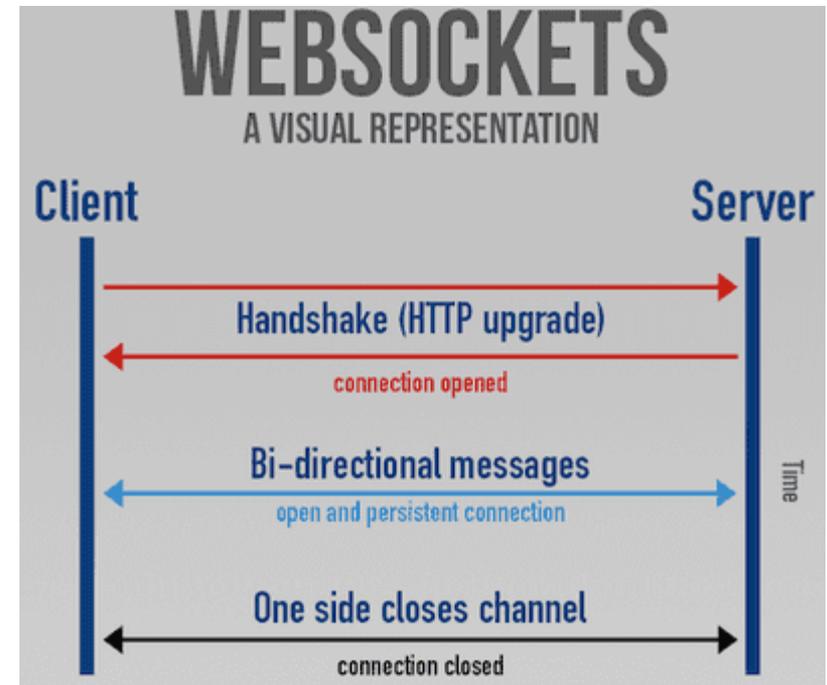
Real-time



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

WebSockets

- Proporciona canales de comunicaciones full-dúplex a través de una única conexión TCP
- Para implementarlo se establece un túnel de comunicación entre servidor-cliente.
- El protocolo está definido con los prefijos ws:// para WebSocket y wss:// para WebSocket Secure.



Trabajos Relacionados

- “Análisis del uso de Websockets para implementar aplicaciones web en tiempo real” de José Guillermo Fierro Mendoza
- “WebSocket Adoption and the Landscape of the Real-Time Web” de Paul Marley, Zane Ma, Joshua Mason, Michael Bailey y Amin Kharraz.
- “Real-time monitoring using AJAX and WebSockets” de Yash Gupta y Alka Leekha.
- “Comunicación en tiempo real por medio de websockets para el control del prototipo Remington” de Diego Edinson Ramires Claros y Miguel Angel Hernandez Pedraza



DESARROLLO



Levantamiento de Requerimientos

Equipo SCRUM

Roles	Descripción	Persona
Product Owner	Gerente General Gasintec S.A.	Ing. David Herdoiza
Scrum Master	Director del proyecto	Juan José Pérez
Development Team	Desarrollador	Juan José Pérez



Levantamiento de Requerimientos

Requerimientos Funcionales

RF	NR	Características
RF-01	Login	El sistema ofrecerá al usuario una pantalla para ingresar su usuario y contraseña.
RF-02	Visualización de lados de los surtidores	Luego del login, el despachador observará todos los lados de cada surtidor de la gasolinera con la información de ventas realizadas o realizándose.
RF-03	Cambio de colores.	Se mostrará cada estado de la venta en distintos colores para diferenciarlos.
RF-04	Información del lado	En una ventana visualizar la información almacenada en la base de datos de un lado específico.
RF-05	Registro de Clientes	Al visualizar la información del lado, el despachador registrará o actualizará la información del cliente.
RF-06	Búsqueda de clientes nuevos	Si un cliente no está registrado en la base de datos de la gasolinera se procederá a consultar en una base de datos en común mediante un servicio web.



Levantamiento de Requerimientos

Requerimientos No Funcionales

RNF	NR	Características
RNF-01	Disponibilidad	El enlace por websocket desde la aplicación proporcionada por Gasintec debe estar siempre disponible para obtener los datos.
RNF-02	Operatividad	El sistema debe ser intuitivo y fácil de usar para los usuarios que en su mayoría no poseen conocimientos tecnológicos.
RNF-03	Fiabilidad	La información a mostrar en pantalla para el despachador debe ser la misma que se muestra en la pantalla del dispensador de combustible.



Product Backlog

Historias de Usuario

Historias de Usuario			
Número:	1	Usuario:	Despachador
Nombre historia:	Login		
Prioridad en negocio:	Alta	Riesgo en Desarrollo:	Baja
Puntos Estimados:	2	Iteración Asignada:	1
Programador Responsable:	Juan José Pérez		
Descripción:	El sistema ofrecerá al usuario una pantalla para ingresar su usuario y contraseña.		
Validación:	El despachador ingresará el usuario y la contraseña asignada e ingresarán al sistema y se mostrarán sus opciones.		



Product Backlog

Historias de Usuario

Historias de Usuario			
Número:	2	Usuario:	Despachador
Nombre historia:	Visualización de lados de los surtidores		
Prioridad en negocio:	Alta	Riesgo en Desarrollo:	Media
Puntos Estimados:	7	Iteración Asignada:	1
Programador Responsable:	Juan José Pérez		
Descripción:	Luego del <u>login</u> , el despachador observará todos los lados de cada surtidor de la gasolinera con la información de ventas realizadas o realizándose.		
Validación:	Visualización en tiempo real de la información de los surtidores de gasolina. La información a visualizar es: <ul style="list-style-type: none">- Volumen despachado- Precio del combustible- Valor total de venta		



Product Backlog

Historias de Usuario

Historias de Usuario			
Número:	3	Usuario:	Despachador
Nombre historia:	Cambio de colores.		
Prioridad en negocio:	Alta	Riesgo en Desarrollo:	Baja
Puntos Estimados:	4	Iteración Asignada:	1
Programador Responsable:	Juan José Pérez		
Descripción:	Se mostrará cada estado de la venta en distintos colores para diferenciarlos.		
Validación:	Los colores y estados son los siguientes: <ul style="list-style-type: none">○ Inactivo – Plomo○ Activo – Celeste○ Despachando – Morado○ Cerrado – Rojo		



Product Backlog

Historias de Usuario

Historias de Usuario			
Número:	4	Usuario:	Despachador
Nombre historia:	Información del lado		
Prioridad en negocio:	Alta	Riesgo en Desarrollo:	Media
Puntos Estimados:	2	Iteración Asignada:	2
Programador Responsable:	Juan José Pérez		
Descripción:	En una ventana visualizar la información almacenada en la base de datos de un lado específico.		
Validación:	Al presionar en un lado, este desplegará la siguiente información: <ul style="list-style-type: none">- Datos del cliente- Forma de pago- Tipo de venta- Placa del vehículo.		



Product Backlog

Historias de Usuario

Historias de Usuario			
Número:	5	Usuario:	Despachador
Nombre historia:	Búsqueda de Clientes		
Prioridad en negocio:	Alta	Riesgo en Desarrollo:	Media
Puntos Estimados:	2	Iteración Asignada:	2
Programador Responsable:	Juan José Pérez		
Descripción:	Al visualizar la información del lado, el despachador registrará o actualizará la información del cliente		
Validación:	Ingreso y búsqueda de la información del cliente y almacenamiento en la base de datos.		



Product Backlog

Historias de Usuario

Historias de Usuario			
Número:	6	Usuario:	Despachador
Nombre historia:	Búsqueda de clientes nuevos y registro de placas		
Prioridad en negocio:	Media	Riesgo en Desarrollo:	Media
Puntos Estimados:	3	Iteración Asignada:	2
Programador Responsable:	Juan José Pérez		
Descripción:	Si un cliente no está registrado en la base de datos de la gasolinera se procederá a consultar en una base de datos en común mediante un servicio web.		
Validación:	Al enviar la solicitud al servicio web este devolverá la información del cliente en <u>json</u> y se mostrará en una ventana emergente para registrarlo.		



Sprint Planning

Descripción	Prioridad	Complejidad	Sprint	Duración(días)
Arquitectura del sistema	5	4	0	10
Login	5	2	1	5
Visualización de lados de los surtidores	5	5	1	30
Cambio de estados	5	5	1	10
Información del lado	4	4	2	5
Búsqueda de Clientes	4	3	2	5
Búsqueda de clientes nuevos y registro de placas	4	3	2	5
			Total	70



Sprint Backlog 0

ID	Responsable	Descripción	Tareas	Esfuerzo empleado en días
1	Juan José Pérez	Necesito entender el proceso de venta	Realizar el diagrama del funcionamiento	2
			Revisión del diagrama	1
			Realizar el diagrama de acciones	2
			Revisión del diagrama	1
2	Juan José Pérez	Necesito preparar el entorno de desarrollo	Diseño y revisión de la vista principal de la aplicación	2
			Configurar el framework de desarrollo	2
			Total:	10

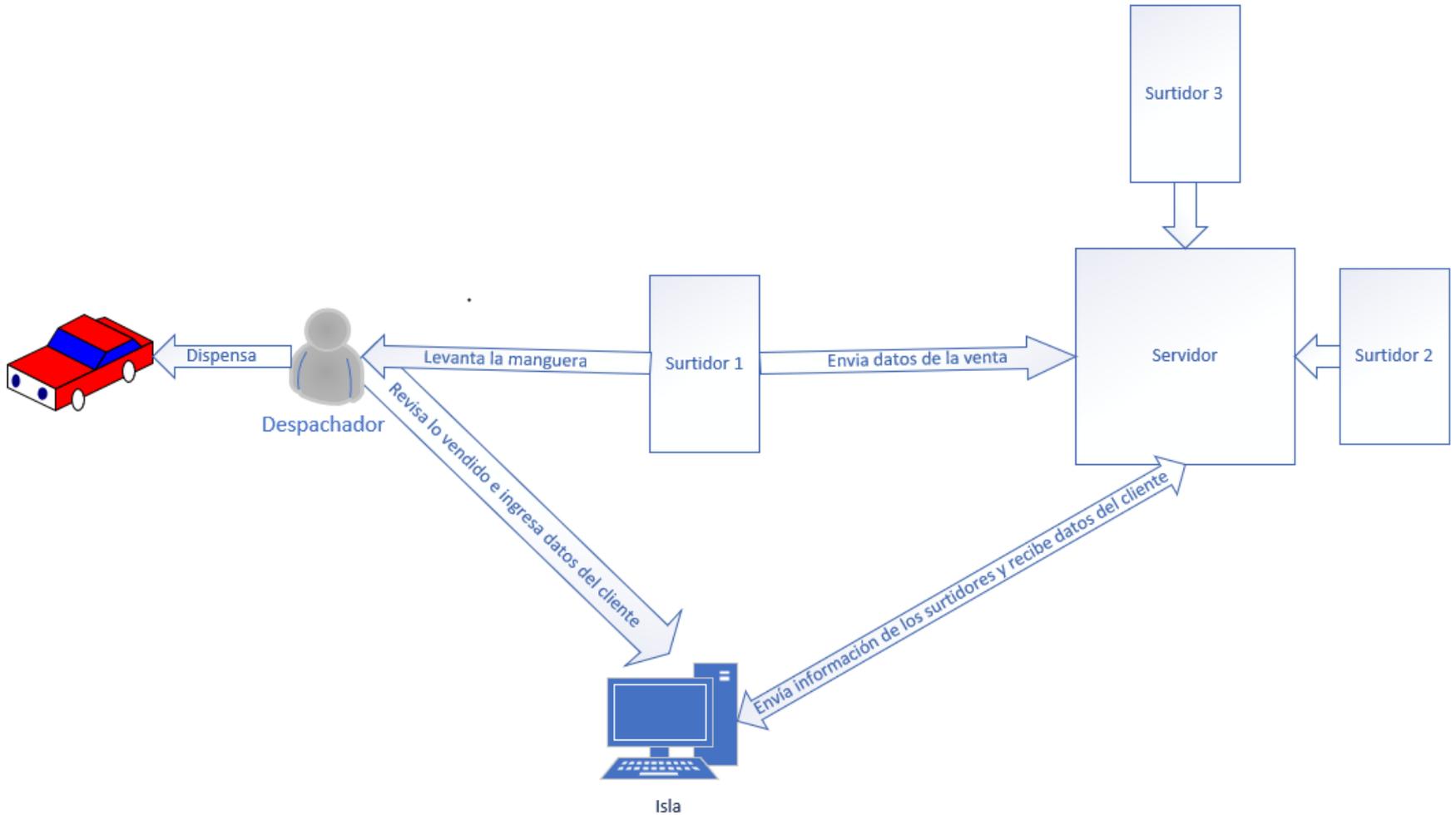


Sprint Review 0

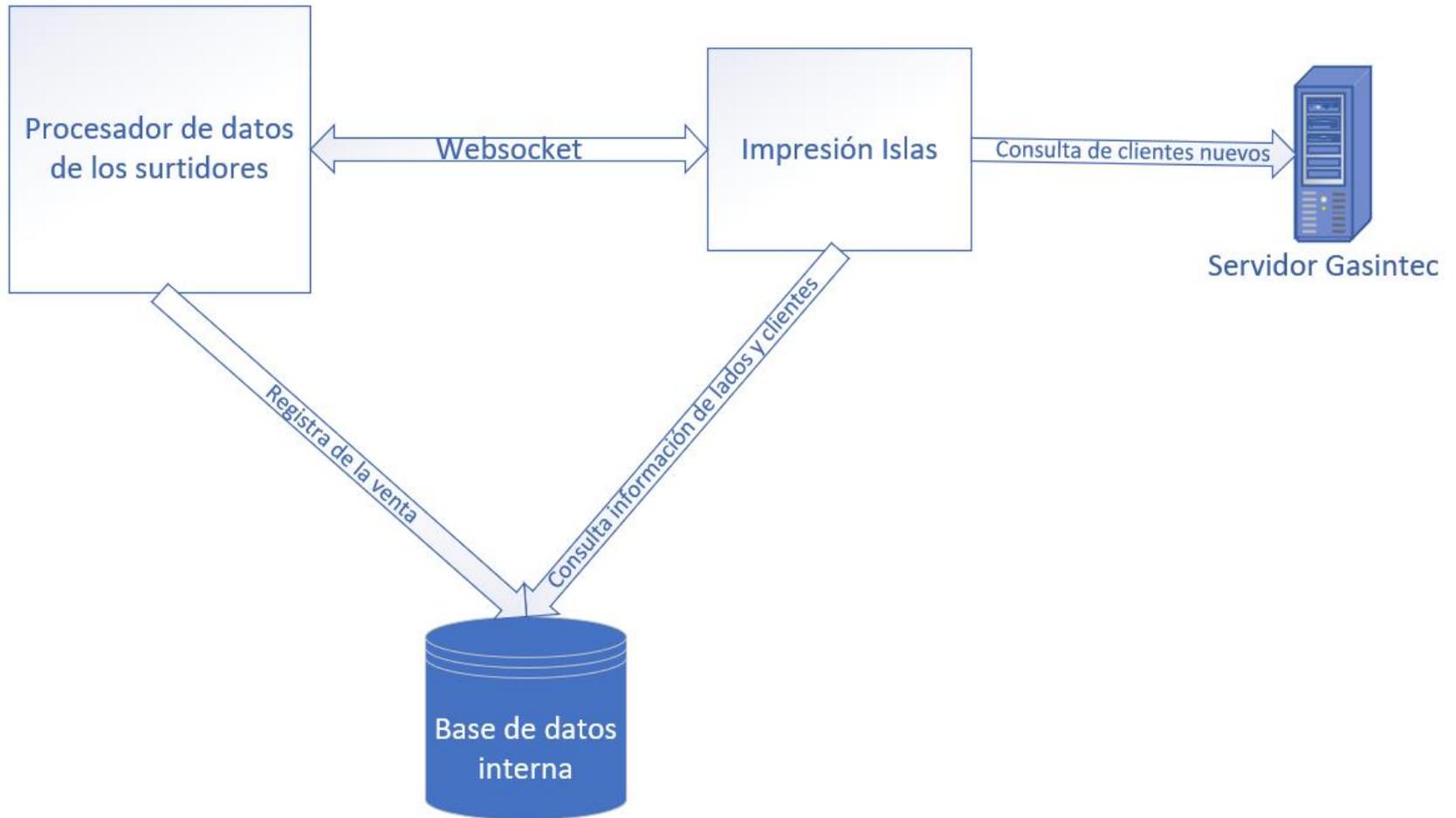
ID	Tarea	Iteración	Avance
1	Diagrama de Funcionamiento	0	100%
	Diagrama de Acciones	0	100%
2	Vista principal de la aplicación	0	100%
	Configuración del framework de desarrollo	0	100%



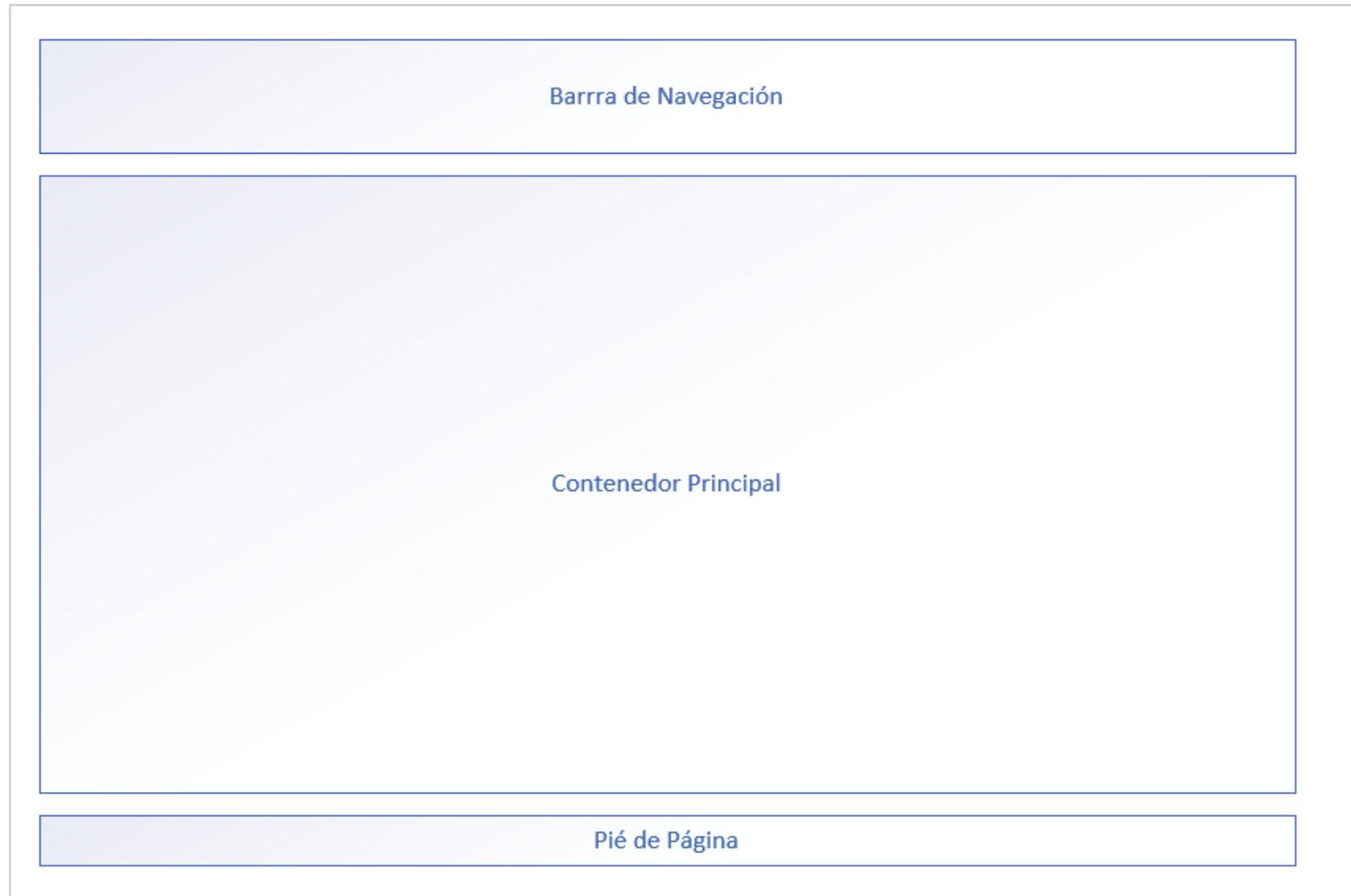
Entregables Sprint 0



Entregables Sprint 0



Entregables Sprint 0



Entregables Sprint 0

- **IDE:** Apache Netbeans IDE 12.1
- **Backend:** Java 8
- **Frontend:** Primefaces 10
- **Base de datos:** PostgreSQL 13
- **Servidor web:** Wildfly 13
- **Sistema Operativo:** Windows 10



Sprint Backlog 1

ID	Responsable	Descripción de la historia de usuario	Tareas	Esfuerzo empleado en días
3	Juan José Pérez	Necesito ingresar a la aplicación con un usuario y contraseña	Crear una vista para el ingreso	2
			Validar campos de ingreso	1
			Crear un controlador para datos de ingreso	2

4	Juan José Pérez	Necesito visualizar los lados de la estación de servicio	Crear una vista para listar los lados de los surtidores	3
			Crear un controlador para consultar los lados	2
			Conectar el websocket con la aplicación de Gasintec	10
			Mostrar la información del websocket en la vista	10
			Verificar que la información sea en tiempo real	5

5	Juan José Pérez	Necesito visualizar la información cada lado con colores distintivos	Definir los estados y colores	1
			Crear controlador de colores por estado	8
			Verificar la información en la vista.	1



Sprint Review 1

ID	Tarea	Iteración	Avance
3	Crear una vista para el ingreso	1	100%
	Validar campos de ingreso	1	100%
	Crear un controlador para datos de ingreso	1	100%
4	Crear una vista para listar los lados de los surtidores	1	100%
	Crear un controlador para consultar los lados	1	100%
	Conectar el websocket con la aplicación de Gasintec	1	100%
	Mostrar la información del websocket en la vista	1	100%
	Verificar que la información sea en tiempo real	1	100%
5	Definir los estados y colores	1	100%
	Crear controlador de colores por estado	1	100%
	Verificar la información en la vista.	1	100%



Entregables Sprint 1



INGRESAR

Control System Evolution v1.00

Gasintec S.A. © Copyright 2020-2022



Información: Usuario no registrado



Información: Usuario y/o Clave incorrecta



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

Entregables Sprint 1

	3.38		10.00
	1.408		4.167
Lado 1	2.400	Lado 2	2.400
	0,00		0,00
	0,000		0,000
Lado 3	0,000	Lado 4	0,000
	14.00		75.02
	7.998		42.870
Lado 5	1.750	Lado 6	1.750
	20.00		10.00
	0.000		4.167



Entregables Sprint 1

 Lado 1	18.02 7507 2400 FACTURA	 Lado 2	25.01 10.419 2400 FACTURA
 Lado 3	0,00 0,000 0,000 CALIBRACION	 Lado 4	0,00 0,000 0,000 CREDITO
 Lado 5	755 4.914 1.750	 Lado 6	32.00 18.289 1.750 FACTURA
 Lado 7	19.99 11.423 1.750	 Lado 8	60.00 34.287 1.750



Sprint Backlog 2

ID	Responsable	Descripción de la historia de usuario	Tareas	Esfuerzo empleado en días
6	Juan José Pérez	Necesito que al presionar en el lado se visualice la información registrada.	Crear una vista para el ingreso de la información.	2
			Al presionar sobre verificar si existe información	1
7	Juan José Pérez	Necesito buscar la información registrada de un cliente	Crear buscadores por código, placa e identificación	2
			Crear un controlador para consultar los clientes	1
			Validar la información según la normativa del SRI.	1
			Mostrar la información en la vista	1



Sprint Backlog 2

ID	Responsable	Descripción de la historia de usuario	Tareas	Esfuerzo empleado en días
8	Juan José Pérez	Necesito buscar clientes no registrados	Validar la información de la identificación según el Registro Civil antes de consultar	2
			Crear una vista para la creación de clientes	1
			Crear un controlador de clientes nuevos	1
			Validar la información del cliente	1
		Necesito registrar placas	Crear una vista para registrar placas	1
			Crear un controlador para registrar placa	1
			Validar la placa a ingresar según el Servicio de Rentas Internas para la autorización de comprobantes electrónicos.	1

Sprint Review 2

ID	Tarea	Iteración	Avance
6	Crear una vista para el ingreso de la información.	2	100%
	Al presionar sobre verificar si existe información.	2	100%
7	Crear buscadores por código, placa e identificación	2	100%
	Crear un controlador para consultar los clientes	2	100%
	Validar la información según la normativa del SRI.	2	100%
	Mostrar la información en la vista	2	100%
8	Validar la información de la identificación según el Registro Civil antes de consultar	2	100%
	Crear una vista para la creación de clientes	2	100%
	Crear un controlador de clientes nuevos	2	100%
	Validar la información del cliente	2	100%
	Crear una vista para registrar placas	2	100%
	Crear un controlador para registrar placa	2	100%
	Validar la placa a ingresar según el Servicio de Rentas Internas para la autorización de comprobantes electrónicos.	2	100%



Entregables Sprint 2

REGISTRO DEL CLIENTE

Guardar

Cancelar

Información del Cliente

Lado 2

Código

Placa

Cédula

Nombre/Razón Social



Actualizar

Seleccione una Placa



Registrar Placa

Factura

Crédito

Prepago

Calibración

Forma de Pago

Efectivo



Billetes



Entregables Sprint 2

REGISTRO DEL CLIENTE

Guardar

Cancelar

Información del Cliente

Lado 2

Código

Placa

ZZZ9999

Cédula

Nombre/Razón Social

Actualizar

Factura

Crédito

Forma de Pago

Efectivo

Billetes

- 1, AGENCIA DE REGULACION Y CONTROL HIDROCARBURIFERO ARCH
- 45, CHICAIZA LOPEZ LILIA PATRICIA
- 95, VARGAS QUINTANILLA EDUARDO BOLIVAR
- 106, CEDENO MACIAS ROBINSON ORLEY
- 139, CHUSIN GRAY LUIS FELIPE
- 200, HY-LINE ECUADOR S.A.
- 233, OTO TONATO MANUEL MARIA
- 258, ERAS HIDALGO MARIA FRANCELINA
- 263, RODRIGUEZ GALLO ANDERSON JAVIER
- 301, ENERMAX S.A



Entregables Sprint 2

Lado 2

Código

1

Placa

ZZZ9999

Cédula

1768158680001

Nombre/Razón Social

AGENCIA DE REGULACION Y CONTROL HIDROCARBURIFERO ARCH



Actualizar

ZZZ9999



Registrar Placa

Factura

Crédito

Prepago

Calibración



Lado 2

10.00

4.166

2.400

CALIBRACION



Entregables Sprint 2

Registrar Cliente



Cedula Ciudadania RUC Pasaporte

Identificación

1804474466

Nombre

PEREZ JEREZ JUAN JOSE

Dirección

Teléfono

Correo

✓ Guardar

✗ Cancelar



Entregables Sprint 2

Registrar Placa ×

Placa

 Información: La placa ingresada no tiene el formato correcto



ANÁLISIS DE RESULTADOS



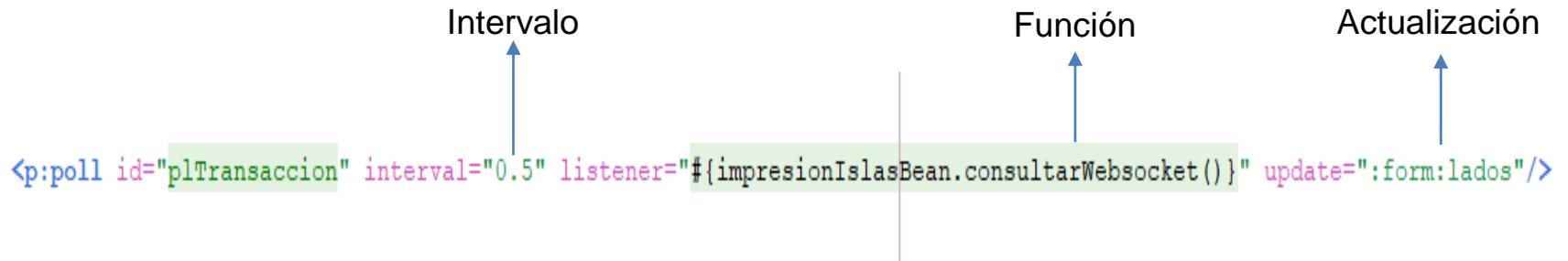
Tiempo Real

- Poll con WebSockets
- JavaScript con WebSockets
- JavaScript con Websockets y API Fetch



Tiempo Real

Poll con WebSocket



Trama: Lado-producto-total-volumen-precio-estado



Tiempo Real

Poll con WebSocket

Característica	Resultado
Determinismo	<ul style="list-style-type: none">- El sistema no garantiza la ejecución de otras tareas al mismo tiempo.
Responsividad	<ul style="list-style-type: none">- La ejecución de la petición no permite la ejecución de otras.- No se puede ejecutar más de una tarea.
Usuarios controladores	<ul style="list-style-type: none">- El proceso prioriza solo su ejecución
Confiabilidad	<ul style="list-style-type: none">- A pesar de no ejecutarse otras tareas, luego del tiempo deseado si se visualiza el valor de las ventas.
Operación a prueba de fallas duras	<ul style="list-style-type: none">- El sistema no es estable.- Si existe un fallo el sistema si preserva los datos.



Tiempo Real

JavaScript con WebSockets

Característica	Resultado
Determinismo	<ul style="list-style-type: none">- El sistema garantiza la ejecución de otras tareas al mismo tiempo.- El tiempo en ejecutar el procesamiento de la información es mínimo en un principio ralentizando con el transcurrir de los días.
Responsividad	<ul style="list-style-type: none">- La ejecución de la petición se procesa en background, por lo que no interfiere con los otros procesos, hasta que el aumento de la memoria RAM ralentiza el sistema.- El usuario al solicitar la ejecución de otras tareas no tiene efecto secundario en el procesamiento de los datos de los surtidores hasta que aumenta la memoria RAM del navegador.

Usuarios controladores	<ul style="list-style-type: none">- El procesamiento de los datos de ventas se ejecuta en segundo plano, permitiéndole al usuario priorizar la tarea deseada.- El proceso no tiene control sobre la memoria RAM a utilizar y con el transcurso del tiempo almacena abundante memoria caché.
Confiability	<ul style="list-style-type: none">- Con el transcurso del tiempo la calidad del sistema se degrada.
Operación a prueba de fallas duras	<ul style="list-style-type: none">- Si el registro de un lado está incompleto no afecta en la visualización de los otros lados, y cuando recepta el siguiente registro lo actualiza.- El sistema no es estable.



Tiempo Real

JavaScript con Websockets y API Fetch

```
const API_URL = window.location.protocol + "://" + window.location.host + "/ruta";
```

```
fetch(API_URL).then(function (response) {  
    openWebSocketClient(hostname);  
});
```

```
websocketURL = protocol + "://" + hostname + ":" + port + endpoint;
```



Tiempo Real

JavaScript con Websockets y API Fetch

```
socket = new WebSocket(webSocketURL);
socket.onopen = function (openEvent) {
    socket.send("Hi");
};
socket.onclose = function (event) {
    if (event.wasClean) {
        console.log('websocket cerrado');
    } else {
        console.log('websocket no cerrado');
    }
};
socket.onerror = function (event) {
    console.log("Error en la ejecución del websocket: " + e);
};
socket.onmessage = function (event) {
    let data = event.data;
    try {
        mostrartiemposeal(data);
    } catch (e) {
        console.error(e);
        data = null;
    }
    data = null;
};
```



Tiempo Real

JavaScript con Websockets y API Fetch

Característica	Resultado
Determinismo	<ul style="list-style-type: none"> - El sistema garantiza la ejecución de otras tareas al mismo tiempo. - El tiempo en ejecutar el procesamiento de datos sobre la venta es mínimo.
Responsividad	<ul style="list-style-type: none"> - La ejecución de la petición se procesa desde <u>le</u> navegador, por lo que no interfiere con los otros procesos que necesitan transaccionar desde el servidor. - El usuario al solicitar la ejecución de otras tareas no tiene efecto secundario en el procesamiento de los datos de los surtidores.

Usuarios controladores	<ul style="list-style-type: none"> - El procesamiento de los datos de ventas se ejecuta en segundo plano, permitiéndole al usuario priorizar la tarea deseada. - El proceso no tiene control sobre la memoria RAM a utilizar, pero lo usado por esta no afecta los otros procesos de la aplicación y/o servidor.
Confiabilidad	<ul style="list-style-type: none"> - Con el transcurso del tiempo la calidad del sistema no se degrada. - Si la trama recibida por el websocket está incompleta no afecta el funcionamiento de la aplicación, ni de los otros procesos, se actualiza lo recibido.
Operación a prueba de fallas duras	<ul style="list-style-type: none"> - Si el registro de un lado está incompleto no afecta en la visualización de los otros lados, y cuando recepta el siguiente registro lo actualiza. - El sistema es estable con el tiempo.





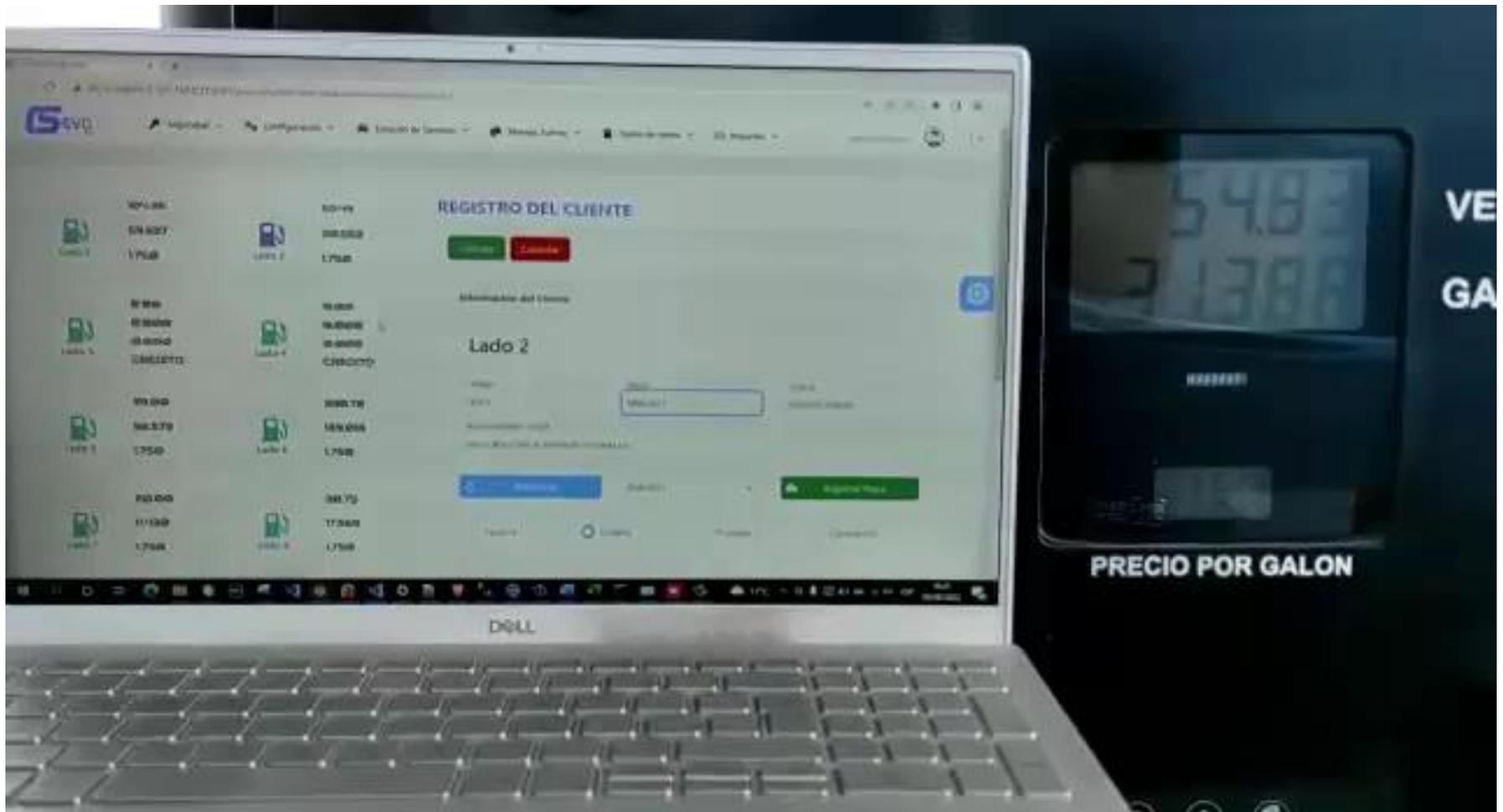
Control System V2.0 Establecimiento: ESTACION DE SERVICIO ALOAG CIA. LTDA. [Status: NUEVA]

Controlador
 Información de Turnos
 Factura Electrónica

Borrar Venta Terminada
 Limpiar Datos Cliente
 Cerrando Turno Vigente

LADO	Ap	Li	Li
LADO 1	203.42	116.240	1.750
LADO 2	40.00	22.855	1.750
LADO 3	12.00	6.861	1.750





CONCLUSIONES



- El protocolo websocket con el API Fetch de JavaScript facilita la visualización de la información en tiempo real y optimiza los recursos de hardware del servidor. Minimiza la latencia y permite que el usuario priorice las acciones a ejecutar por la aplicación.
- No se puede definir cuanto tiempo es tiempo real, puesto que depende de la percepción del usuario, el software en tiempo real se basa en la aplicación de todas sus características sin excepción.
- El software en tiempo real facilita al despachador de una gasolinera el control sobre el combustible surtidor y valor vendido en efectivo, así como conocer el estado de su venta mediante una respuesta gráfica de la pantalla.
- WebSocket es mejor para el manejo de tiempo real que Poll de Primefaces, Websocket abre un túnel de comunicación bidireccional segura en el cual el cliente no necesita realizar una petición para obtener información, mientras que Poll depende del intervalo de tiempo ingresado para su ejecución.



RECOMENDACIONES



- La optimización de recursos es fundamental para el funcionamiento correcto de un sistema en tiempo real, el tener en pruebas un hardware potente puede dar una falsa ilusión de que se llegó al objetivo, se recomienda que antes de salir a producción se realice pruebas en el hardware mínimo que puede tener.
- Existen varias APIs de JavaScript para el manejo del tiempo real, y constantemente hay nuevas librerías, APIs que facilitan su ejecución. Antes de empezar el proyecto se recomienda revisar cuál sería la más óptima según lo deseado.
- Realizar una revisión bibliográfica actualizada, las inquietudes de desarrollo o entendimiento de un problema, alguien ya lo puede haber resuelto.
- Analizar el usuario principal del sistema a realizar para la creación de las diferentes vistas y facilitar su uso. Por ejemplo, el sistema es vendido al dueño de una gasolinera, pero él no es el usuario del sistema, el que está en contacto siempre es el despachador, en algunos casos adultos mayores, o personas con un mínimo de estudio. un software no amigable es posible que lo desechen.



