

UNIVERSIDAD DE LAS AMÉRICAS
FACULTAD DE INGENIERÍA Y NEGOCIOS

MONITOREO INFRAESTRUCTURA TALLER AUTOMOTRIZ

Hernán Galleguillos Mella

2018

UNIVERSIDAD DE LAS AMÉRICAS
FACULTAD DE INGENIERÍA Y NEGOCIOS

MONITOREO INFRAESTRUCTURA TALLER AUTOMOTRIZ

Trabajo de titulación presentado en conformidad a los requisitos para obtener el título de Ingeniero de Ejecución en Informática.

Profesor: Sr. Mauricio Hormazábal.

Hernán Galleguillos Mella

2018

ÍNDICE

Introducción.....	1
1. Aspectos del Mercado	2
1.1 Visión de la Empresa.....	2
1.2 Misión de la Empresa	2
1.3 Antecedentes de la Empresa.....	2
1.4 Organigrama.....	3
1.5 Area Funcional.....	3
1.6 Descripción del Proceso a intervenir	3
2. Situación Actual del Proyecto	4
2.1 Descripción de la Situación Actual.....	4
2.2. Descripción del Problema.....	5
2.3 Propósito del proyecto	6
2.4 Estado del Arte	6
2.5 Solución Planteada.....	8
2.6 Alcances... ..	8
2.7 Restricciones	8
3. Planteamiento de Objetivos	8
3.1 Objetivos Generales	8
3.2 Objetivos Específicos	9
4. Estudio de Factibilidad y Gestión de Riesgos	9
4.1 Estudio de Factibilidad	9
4.1.1 Factibilidad Técnica.....	9
4.1.2 Análisis costo beneficio a un año.....	12
4.1.3 Factibilidad Operacional	13
4.1.4 Factibilidad Legal.....	14
4.1.5 Aspectos de Responsabilidad Legal.....	15
4.2 Identificación de Riesgos.....	15
4.2.1 Riesgos de Planeación	15
4.2.2 Riesgos de Desarrollo	15
4.2.3 Riesgos del Cliente.....	16

4.2.4 Riesgos de Implementación	16
4.2.5 Matriz de Riesgo.....	16
5. Planteamiento de la Solución.....	17
5.1 Análisis de la Solución.....	17
5.1.1 Descripción solución propuesta en detalle	17
5.1.2 Evaluación de Herramienta de desarrollo e Implementación.....	18
5.1.3 Diagrama de Arquitectura de Solución Propuesta.....	19
5.1.4 Diagrama Proceso Mejorado	20
5.2 Metodología Aplicada	20
5.2.1 Comparativa de metodologías asociadas al proyecto	20
5.2.2 Justificación de la metodología seleccionada.....	21
5.2.3 Metodología de la Administración.....	22
5.2.4 Carta Gantt	22
5.3 Obtención de Requerimientos.....	23
5.3.1 Metodología Aplicada a la Toma de Requerimientos	23
5.3.2 Requerimientos Funcionales	23
5.3.3 Requerimientos No Funcionales.....	24
5.3.4 Requerimientos de Seguridad	25
5.3.5 Requerimientos de Mantenimiento	26
5.3.6 Especificación de Requerimientos	27
6. Diseño del Sistema	26
6.1 Modelamiento UML	27
6.1.1 Diagrama Casos de Uso.....	27
6.1.2 Documentación Casos de Uso	28
6.1.3 Diagrama de Componentes.....	29
6.2 Modelamiento de Datos.....	31
7. Construcción del Prototipo Modelo Funcional	31
8. Diseño de Pruebas del Software	35
9. Conclusiones del Proyecto	36
9.1 Conclusiones.....	36
9.2 Perspectivas Futuras.....	37

INTRODUCCIÓN

Los servicios de mantenimiento ligados al mundo motor (vehículos menores, particulares, pesados o maquinaria) tienen acogida en los grandes concesionarios o mas pequeños Talleres de Mecánica, todos los servicios tienen sus propios sistemas administrativos (financieros, RRHH, inventarios, etc) pero no todos cuentan con un sistema de monitoreo que supervise la infraestructura física.

La infraestructura actual considera depósitos de lubricantes, residuos químicos, red húmeda, equipos de aire comprimido, colector de virutas o desechos metálicos, depósitos de aceites, etc, todo lo anterior en conjunto permiten desarrollar las funciones regulares al interior del taller, como son los distintos servicios prestados de distinta envergadura.

Como objetivo general se desea implementar un servicio de control y monitoreo que permita administrar y generar alertas tempranas frente a las distintas situaciones que se puedan presentar en el Taller, por ejemplo, humedad excesiva, detección de humo, detección de derrame o fuga de fluidos, aumento de temperatura en sectores de alto riesgo, depósitos aproximándose a niveles críticos, despositos de desechos solidos al máximo de capacidad, etc.

El resultado esperado es poder mejorar las gestión del taller, recibiendo alertas tempranas de modo de poder ejecutar las acciones pertinentes para evitar inconvenientes logrando así la reducción de los riesgos y mejorando el servicio de cara al cliente final.

En relación a la metodología a utilizar para el desarrollo de este proyecto será la denominada eXtreme Programming (XP) , la cual es considerada una metodología ágil ya que dentro de sus principales características podemos mencionar satisfacer la necesidad o al cliente por medio de entregas tempranas y continuas de software, centrarse en la solución más que en preparar documentación y desarrollo sostenible pro mencionar algunas.

Este proyecto tendrá como objetivo principal, mediante una aplicación, la captura de datos ambientales o variables del entorno físico, con el cuál se podrá realizar una gestión automatizada tipo industrial sobre los distintos eventos físicos que se presenten al interior del Taller.

Cabe mencionar que la solución podría haberse implementado con algún hardware más idóneo como los del tipo Arduino, pero considerando la flexibilidad y potencial para desarrollar aplicaciones más robustas que sólo las del tipo “electrónica” es que nuestra solución se desarrollará sobre los Ordenadores RaspBerry, y esto se explica en el capítulo de Factibilidad.

1. ASPECTOS DE LA EMPRESA.

1.1. Visión

En un mercado cada vez más desconforme “Taller Automotriz Ramon Osorio quiere estrechar su mano e invitarlo a que ud. disfrute del placer de dejar su vehículo en los mejores especialistas.”

1.2. Misión

Con nuestras buenas prácticas y espíritu de servicio queremos respaldar al mercado del servicio automotriz de Chile, brindando nuestros conocimientos, experiencia y asesoría técnica a todos nuestros clientes.

1.3. Antecedentes de la Empresa.

Taller Automotriz Ramón Osorio es una empresa que nace hace 28 años pensando en las necesidades de los clientes en el rubro de mecánica automotriz, a falta de servicios garantizados y de calidad por parte de los talleres automotrices.

Desde sus inicios la empresa se ha caracterizado por su responsabilidad, garantía y resultados destacados en cada una de sus servicios o proyectos especializados.

La empresa posee instalaciones propias tanto inmobiliarias como de infraestructura.

1.4. Organigrama.

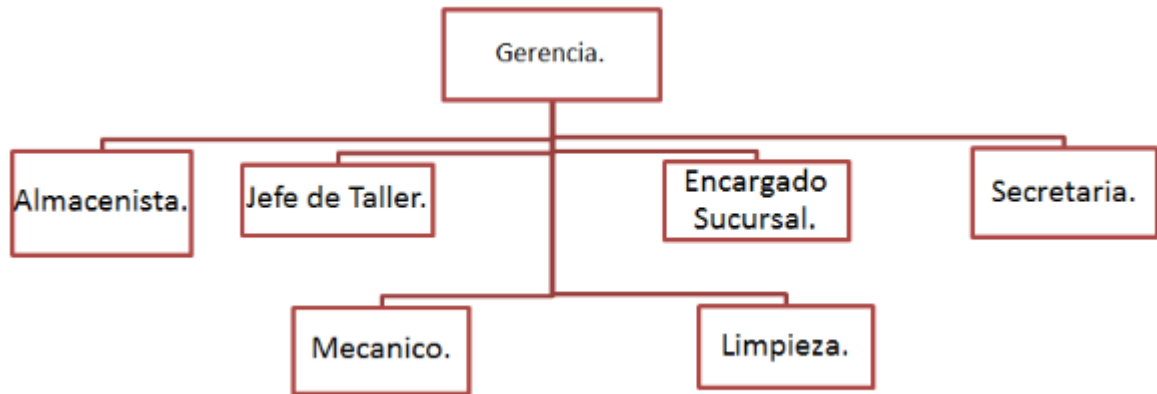


Figura 1 – Organigrama de la Empresa

1.5. Área Funcional.

El área funcional sobre la cual se focaliza este proyecto se limita al área infraestructura, suministros y control de residuos, la que se encarga de inspeccionar los equipos como así también las zonas de riesgo al interior del taller.

La inspección visual manual abarca todo el perímetro del taller lo que incluye de dos zonas de productos inflamables, una para residuos y otras tres que agrupan equipos eléctricos de diversa naturaleza.

1.6. Descripción del Proceso a intervenir.

El proceso a intervenir actual es totalmente manual, no considera herramientas ni tecnología de ningún tipo. El espacio físico tiene sectores determinados para cada propósito como por ejemplo oficina, repuestos, pozo y otros donde se agrupan lubricantes, residuos y otros destinados a equipamiento como tornos, compresores, servidores, equipos de alta presión, red húmeda por nombrar los más relevantes. El proceso como tal es llevado por la persona encargada de la limpieza que reporta al Jefe de Taller, la que semanalmente ejecuta

inspecciones visuales, y que frente a alguna situación irregular informa a su jefatura directa. Esta gestión carece de normas o protocolos, tampoco es rigurosa, siendo un chequeo visual no hay registro de su ejecución por lo cual el riesgo está presente a diario y tampoco se puede consultar ni hacer gestión de ningún evento al interior del taller.

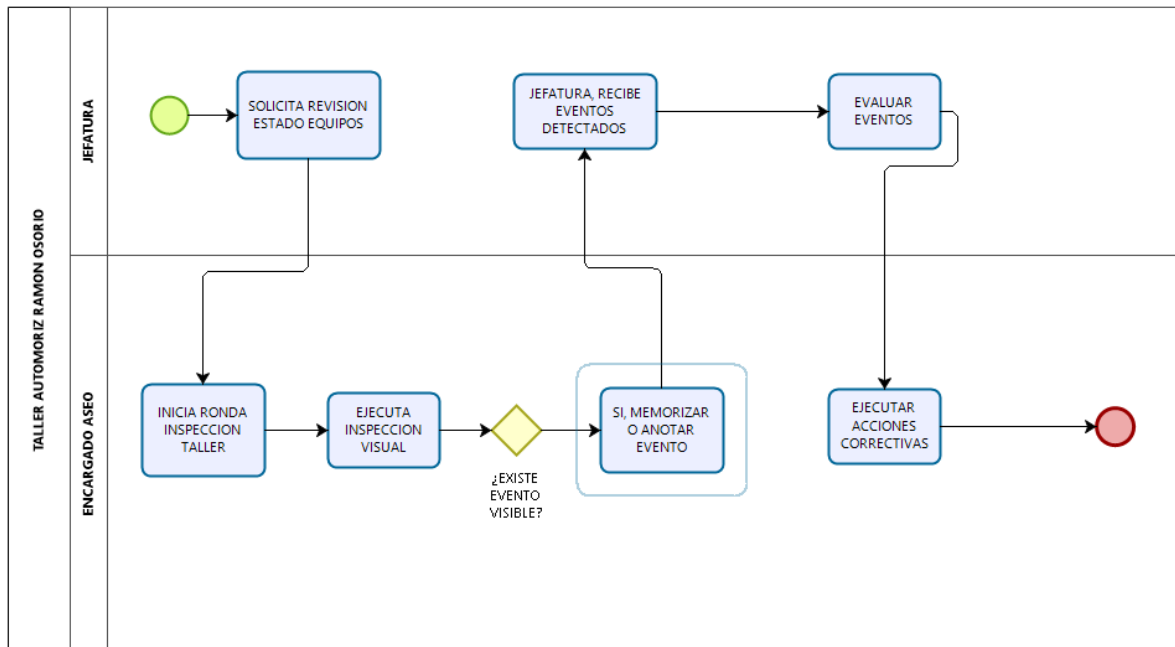


Figura 2 – Diagrama Proceso a Intervenir

2. SITUACIÓN ACTUAL DEL PROYECTO

2.1. Descripción de la Situación Actual

Dentro del escenario actual la empresa Taller Automotriz “Ramón Osorio” considera para dar sus servicios una serie de equipos eléctricos, depósitos de materiales peligrosos, residuos químicos e incluso desechos altamente tóxicos. Si bien los componentes antes mencionados tienen sus propias reglas de seguridad ellos no consideran ningún elemento que permita monitorear de manera automática, ya que la revisión es manual lo que inevitablemente a veces no ocurre.

Los depósitos de desechos son estanques que no disponen de ningún método de alerta lo que conlleva a que cuando se llega a su capacidad máxima recién se activa el proceso de eliminación lo cual genera un encolamiento de los desechos que se van generando.

Por otro frente están los lubricantes que por cada servicio que se deba atender con la extracción del mismo se provoca derrame lo cual a la larga genera suciedad, humedad y alto riesgo de inflamación.

También existen los restos o basura sólida, la que está alojada en un contenedor especial la cual tampoco dispone de ningún método de monitoreo y cuando se sobrepasa su capacidad máxima también genera inconvenientes.

En general existe varias situaciones las cuales dependen de un “chequeo visual” periódico el cual no siempre se lleva a cabo, lo que conlleva a situaciones que van en desmedro de la calidad de los trabajos contratados y generan situaciones de riesgo.

El monitoreo del estado del equipamiento eléctrico (sobrecarga), temperatura de los distintos sistemas de apoyo, el alojamiento de materiales inflamables, el oportuno manejo de los desechos, todo puede ser controlado por un sistema de sensores los que diseñados eficazmente pueden ser de gran ayuda al generar avisos o alertas de situaciones que deben ser atendidas.

Por todo lo anterior esta solución será un apoyo en mejorar la calidad del servicio lo que se traducirá en mejores utilidades gracias en parte a la eliminación de problemas de suministros, equipos dañados, humedad y riesgos de diversa índole.

2.2. Descripción del problema.

El hecho de disponer de equipos eléctricos, sectores expuestos a humedad, depósitos de elementos tóxicos e inflamables hace necesario el poder controlar y supervisar el estado y condición operacional a diario de cada elemento, pero el control es precario, manual, poco riguroso, no permite hacer mejoras ni gestión sobre los eventos que se presenten, generando mermas económicas, daños y riesgos.

2.3. Propósito del proyecto.

El propósito de este proyecto es implementar una solución tecnológica que permita mejorar el monitoreo, control y alerta temprana a eventos que pongan en riesgo al servicio, a la infraestructura o peor aún a la integridad física de las personas.

2.4. Estado del Arte.

El control automático de procesos es una de las disciplinas que se ha desarrollado a una velocidad vertiginosa, dando las bases a lo que hoy algunos autores llaman la segunda revolución industrial. El uso intensivo de las técnicas del control automático de procesos tiene como origen la evolución y tecnificación de las tecnologías de medición y control aplicadas al ambiente industrial.

(Patricio Abarca. (2015). El ABC de la Automatización. Septiembre 2018, de Asociación de la Industria Eléctrica Chile Sitio web:

<http://www.aie.cl/files/file/comites/ca/abc/sistemas-de-control-automatico.pdf>)

El principio de todo sistema de control automático es la aplicación del concepto de realimentación o feedback (medición tomada desde el proceso que entrega información del estado actual de la variable que se desea controlar) cuya característica especial es la de mantener al controlador central informado del estado de las variables para generar acciones correctivas cuando así sea necesario. Este mismo principio se aplica en campos tan diversos como el control de procesos químicos, control de hornos en la fabricación del acero, control de máquinas herramientas, control de variables a nivel médico e incluso en el control de trayectoria de un proyectil militar.

(Patricio Abarca. (2015). El ABC de la Automatización. Septiembre 2018, de Asociación de la Industria Eléctrica Chile Sitio web:

<http://www.aie.cl/files/file/comites/ca/abc/sistemas-de-control-automatico.pdf>)

Las dos técnicas básicas de control automático, es decir, el ingreso al controlador de las variables aleatorias y de las variables controladas, dan lugar,

respectivamente, a los sistemas de lazo abierto y de lazo cerrado. La distinción entre ambos tipos de sistemas se basa en la existencia o ausencia de un camino de realimentación mediante el cual las variables controladas ingresen al controlador; si existe este camino, el sistema será de lazo cerrado; en caso contrario; será de lazo abierto.

Profesor Molina. (2015). Teoría con Automatización. Septiembre 2018, de Automatización Sitio web:

http://www.profesormolina.com.ar/tecnologia/teoria_con_aut.htm

Actualmente esta tecnología base es usada principalmente para el entretenimiento, automatización y control al interior de las viviendas pero puede ser usada en diversos proyectos de manera más accesible a nivel de costos, implementación como su mantenimiento.

A nivel de implementaciones existen soluciones disponibles que incluyen el monitoreo y alerta básica frente a eventos determinados, tanto en el ámbito de manejo de materiales tóxicos como también en talleres de servicios automotriz. Nuestra solución considera el uso de sensores de alta especialización que poseen atributos y características que permiten esta implementación ya que están diseñados pensando en cubrir diversos propósitos y usos, los que conectados a un ordenador entregarán más inteligencia en la gestión al interior del Taller Automotriz.

Existen disponibles tecnologías similares que permiten implementar distintas soluciones automatizadas de control, pero nuestro proyecto está basado en una arquitectura más robusta que permite ampliar sus funciones más que un simple sensor conectado a algún dispositivo de alerta.

La tecnología escogida considera un amplio espectro de implementaciones exitosas, biblioteca de software libre y componentes de Hardware de costo muy accesible.

2.5. Solución Planteada.

La solución planteada en este proyecto considera el diseño de un Sistema compuesto de ordenadores, interfases y sensores los que interconectados permitirán monitorear el estado de salud de la infraestructura generando alertas y gestión sobre la operación del servicio consiguiendo mejoras y reducción de los riesgos a la infraestructura y daño a las personas.

2.6. Alcances.

Para este proyecto se definen los siguientes alcances :

- Se controlarán equipos industriales eléctricos
- Adicionalmente se monitorearán los depósitos de materiales tóxicos
- El tiempo de desarrollo y puesta en marcha deberían ser unos 12 meses.
- No se invertirá en equipamiento adicional o nuevo propio del taller.
- El acceso al sistema WEB (usuario) y consola SSH (Adm.)
- El proyecto inicialmente no podrá operar sobre dispositivos móviles.

2.7. Restricciones.

- Los costos del desarrollo no podrán exceder al presupuesto autorizado
- Las pruebas pilotos deben ser ejecutadas en el taller
- No se intervendrá la infraestructura actual del taller

3. PLANTEAMIENTO DE OBJETIVOS.

3.1. Objetivo General.

Desarrollar un sistema especializado que permita hacer un control al interior del taller, apoyando así en la administración y seguridad frente a eventos en el Taller Automotriz Ramón Osorio.

3.2. Objetivos Específicos.

- 1) Revisar el proceso actual analizando la operación manual, personas y las tareas que se ejecutan.
- 2) Establecer nuevo proceso, determinar la mejora al proceso manual implementando un sistema automatizado permitiendo que el proceso manual conviva en un estado pasivo o usable sólo en caso de que el sistema no se encuentre operativo.
- 3) Permitir la visualización de los valores de cada uno de los sensores, registrar bitácora de eventos, generar alertas, permitir acceso remoto de consulta sobre toda la plataforma.

4. ESTUDIO DE FACTIBILIDAD Y GESTIÓN DE RIESGOS

4.1. Estudio de Factibilidad.

4.1.1. Factibilidad Técnica.

La necesidad está enmarcada en desarrollar un sistema que permita controlar una serie de eventos de infraestructura, para lo cual se necesita evaluar diversos criterios técnicos y así diseñar sobre la tecnología disponible más apropiada, a continuación abordaremos un paralelo entre Arduino y Raspberry.

A pesar de que tanto Arduino como Raspberry Pi pueden parecer muy similares en aspecto, su funcionalidad es muy distinta. Lo que si comparten es que ambas son grandes opciones para iniciarse en el campo de la programación y creación de proyectos, por lo que vamos a ver cuáles son las ventajas de elegir una u otra en función del proyecto o tarea que queramos realizar.

Para empezar, la principal diferencia entre Arduino y Raspberry Pi es que el primero es un microcontrolador, mientras que el segundo es básicamente un miniordenador completo. Un microcontrolador es una pequeña parte de lo que forma un ordenador concreto, por lo que a nivel de funcionalidades una placa Arduino es sólo una pequeña parte de lo

que puede ofrecer una Raspberry Pi. Mientras en una placa Arduino se pueden programar, por ejemplo, pequeñas aplicaciones en C++, en Raspberry Pi podemos ejecutar un sistema operativo completo.

Entonces, si una Raspberry Pi es mucho más completa que un Arduino, ¿qué sentido tiene comprar una placa de Arduino? Más del que podría creerse. Arduino es ideal para todo tipo de proyectos de electrónica. Con sus conectores se puede enchufar directamente a componentes y sensores, siendo más rápida a la hora de realizar cambios o proyectos sencillos que no requieran tareas complejas.



Figura 3 – Arduino Vs Raspberry PI

En conclusión si bien Arduino es poderoso en proyectos de electrónica no cumple con las necesidades que pretendemos cubrir con este sistema ya que no sólo esperamos controlar sensores sino además contar con una plataforma de registro de eventos, módulo de consultas, alertas, envío de correos, acceso local y remoto, en otras palabras que disponga de un sistema operativo y una amplia gama de software disponible.

Ahora estableceremos el paralelo entre Raspberry y Orange PI



Figura 4 –Raspberry PI Vs Orange PI

Es cierto que ambos dispositivos son muy similares y, a grandes rasgos, van a funcionar de una forma muy similar. Sin embargo, si de verdad queremos saber cuál es mejor y, sobre todo, qué diferencias hay entre ambos, debemos empezar comparando el hardware de los dos.

En el caso del Raspberry Pi 3, este cuenta con un procesador ARM Cortex-A53 de 64 bits con 4 núcleos a 1.2Ghz junto con 1 GB de memoria RAM compartidos con la GPU. El Orange Pi Prime, en este aspecto, cuenta con un procesador muy similar Allwinner H5 con cuatro núcleos Cortex A-53, aunque sí que cuenta con una GPU bastante superior a la del Raspberry Pi 3, una Mali450 y, además, con 2 GB de memoria RAM. En cuanto a la conectividad inalámbrica ambas placas son iguales, Bluetooth y Wi-Fi n, aunque la Orange Pi ofrece mejor conectividad por cable gracias a su tarjeta de red Gigabit Ethernet, frente a la Fast Ethernet del Raspberry Pi 3. Si buscamos conectividad, en este aspecto puede fallar un poco, ya que el Orange Pi, aunque tiene un receptor IR, solo tiene 3 puertos USB, frente a los 4 que lleva de base su rival. En ambos mini-ordenadores estamos hablando de USB 2.0.

En cuanto al precio, parece que, una vez más, los de Orange Pi han conseguido superar a su rival. Partiendo de una misma página web como

es Aliexpress, el Orange Pi Prime tiene un precio en torno a 30 dólares, mientras que el Raspberry Pi 3 no se encuentra por debajo de 40 dólares. Como podemos ver, a nivel de hardware el Orange Pi es superior. Si comparamos el software, Raspberry Pi tiene un soporte imposible de igualar y un gran apoyo por parte de la comunidad y los desarrolladores, siendo, además, Raspbian, uno de los mejores sistemas operativos para mini-ordenadores.

Por todo lo anterior nuestra solución será desarrollada sobre Raspberry PI, ya que de acuerdo a la naturaleza del sistema y funciones a soportar está muy por sobre de Arduino. Por otro lado a nivel de arquitectura de hardware está levemente bajo Orange PI pero la característica que resolvió la incógnita fue el gran soporte que tiene Raspberry tanto es su comunidad de desarrolladores como su robusto sistema operativo Raspbian.

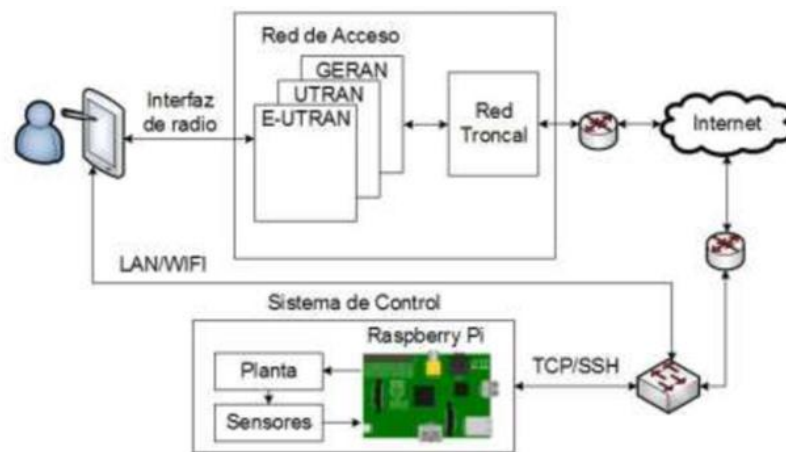


Figura 5 – Ejemplo Diagrama Factibilidad Raspberry

A continuación se mencionan las herramientas, software base y estación de trabajo requeridas para desarrollar esta solución. El acceso al ordenador Raspberry se logra accediendo por medio de un cliente SSH desde una estación Windows o directamente por modo consola al ordenador.

Herramienta	Versión
Debian for RASPBERRY Pi	3.1
MySQL	5.6
Apache WebServer	2
PHP	5
Python language	6
Estación de trabajo desarrollo Lenovo	64 bits

Figura 6 – Herramientas y software base

4.1.2. Análisis costo beneficio a un año.

De acuerdo al siguiente cuadro se pueden observar tanto los costos como ingresos relacionados con el proyecto durante los 12 meses que durará el proyecto, en este se puede observar que el valor del VAN es positivo por lo que podemos concluir que el proyecto es factible o viable de ejecutar.

	Mes 0	Mes 1	Mes 2	Mes 3	Mes 4	Mes 5	Mes 6	Mes 7	Mes 8	Mes 9	Mes 10	Mes 11	Mes 12
Ingresos													
Ingeniero en Informática		300.000	300.000	300.000	300.000	300.000	300.000	300.000	300.000	300.000	300.000	300.000	300.000
Ingeniero en Informática Senior		500.000	500.000	500.000	500.000	500.000	500.000	500.000	500.000	500.000	500.000	500.000	500.000
Total (\$Pesos)	0	800.000	800.000	800.000	800.000	800.000	800.000	800.000	800.000	800.000	800.000	800.000	800.000
	Mes 0	Mes 1	Mes 2	Mes 3	Mes 4	Mes 5	Mes 6	Mes 7	Mes 8	Mes 9	Mes 10	Mes 11	Mes 12
Egresos													
Locomoción(x2)	0	-60.000	-60.000	-60.000	-60.000	-60.000	-60.000	-60.000	-60.000	-60.000	-60.000	-60.000	-60.000
Internet	0	-54.000	-54.000	-54.000	-54.000	-54.000	-54.000	-54.000	-54.000	-54.000	-54.000	-54.000	-54.000
Luz Eléctrica	0	-54.000	-54.000	-54.000	-54.000	-54.000	-54.000	-54.000	-54.000	-54.000	-54.000	-54.000	-54.000
Agua	0	-29.000	-29.000	-29.000	-29.000	-29.000	-29.000	-29.000	-29.000	-29.000	-29.000	-29.000	-29.000
Teléfono	0	-75.000	-75.000	-75.000	-75.000	-75.000	-75.000	-75.000	-75.000	-75.000	-75.000	-75.000	-75.000
Papelería		-60.000	-60.000	-60.000	-60.000	-60.000	-60.000	-60.000	-60.000	-60.000	-60.000	-60.000	-60.000
Total (\$Pesos)	0	-332.000	-332.000	-332.000	-332.000	-332.000	-332.000	-332.000	-332.000	-332.000	-332.000	-332.000	-332.000
Total Ingresos	0	800.000	800.000	800.000	800.000	800.000	800.000	800.000	800.000	800.000	800.000	800.000	800.000
Total Egresos	0	-332.000	-332.000	-332.000	-332.000	-332.000	-332.000	-332.000	-332.000	-332.000	-332.000	-332.000	-332.000
Total neto	0	468.000	468.000	468.000	468.000	468.000	468.000	468.000	468.000	468.000	468.000	468.000	468.000
RAI (Renta Antes del Impuesto)	0	268.000	268.000	268.000	268.000	268.000	268.000	268.000	268.000	268.000	268.000	268.000	268.000
Impuesto (19%)	0	50.920	50.920	50.920	50.920	50.920	50.920	50.920	50.920	50.920	50.920	50.920	50.920
RDI (Renta Despues del Impuest)	0	217.080	217.080	217.080	217.080	217.080	217.080	217.080	217.080	217.080	217.080	217.080	217.080
Depreciación		8.511	8.511	8.511	8.511	8.511	8.511	8.511	8.511	8.511	8.511	8.511	8.511
Inversión Inicial	500.000	0	0	0	0	0	0	0	0	0	0	0	0
Flujo de Caja	-500.000	225.591	225.591	225.591	225.591	225.591	225.591	225.591	225.591	225.591	225.591	225.591	225.591
Tasa de Descuento	0,027												
VAN		2.286.284											
TIR		44,26%											

Figura 7 – Análisis Costo/Beneficio

También se deben considerar otros beneficios valorizables, por mencionar algunos cuantificables : costos en pólizas de seguro, por incendio y almacenaje de materiales inflamables, costos por accidentes personales, costos por demandas civiles o penales en caso de muerte, valorizar la imagen , en general el aumento de estos beneficios obtenidos por la implementación de esta solución.

4.1.3. Factibilidad Operacional.

La solución contemplará un capa de administración encargada de la configuración, seguridad y mantenimiento y otra capa de usuario que explotará el sistema, ejecutará consultas, revisará los eventos reportados, y podrá preparar pequeños informes que facilitarán la gestión en los niveles más altos de la organización.

Para la capa de administración se requieren conocimientos más avanzados en cambio para la explotación solo se requieren conocimientos generales de ambientes Microsoft Windows y visores de páginas WEB (browsers).

Este sistema no interactuará con datos de otros sistemas existentes como Inventario, Clientes ni tampoco datos personales de los empleados.

4.1.4. Factibilidad Legal.

La administración, gestión y desarrollo del proyecto contará con las licencias correspondientes de uso y/o distribución, ya sea de software libre (free software) o de código abierto (open source), por lo tanto, no infringirá ninguna política y/o normativa de derecho de autor.

Respecto de la construcción del sistema y sus componentes, este corresponderá a un producto nuevo elaborado en base a los requerimientos y necesidades planteadas por el usuario, por lo tanto no transgredirá ninguna ley relacionada con la copia o piratería de software.

La plataforma donde se desarrollará el sistema debe contar con las licencias de software requeridas, si se opta por desarrollar a través de un acceso SSH al ordenador, este equipo local requerirá, en el caso de Microsoft Windows,

de la licencia al menos del S.O., si se opta por preparar documentación a la licencia de Sistema Operativo se deberá agregar las de productos tipo Office. En los ordenadores productivos se podrá instalar y usar sin licenciamiento pagado el sistema operativo especial para estos propósitos (Raspian, versión Linux DEBIAN para Raspberry), más servidor Web APACHE, BD MySql y Lenguaje Python.

4.1.5. Aspectos de responsabilidad Legal.

Se establecerá un acuerdo dentro de un marco legal en donde se estipule que frente a una falla del sistema que genere una condición de riesgo o desencadene en un siniestro solo será responsabilidad del taller y no de los propietarios de la implementación o construcción del sistema.

4.2. Identificación de Riesgos.

4.2.1. Riesgos de Planeación

Los riesgos que podrían tener efectos negativos en la planeación de este proyecto están ligados a las siguientes variables:

- Alza en los precios ofrecidos por el mercado (ordenadores).
- Mala toma de requerimientos.
- Falta de supervisión en el diseño.
- Incumplimiento de las fechas acordadas para cada fase.

4.2.2. Riesgos de Desarrollo

Los riesgos que podrían tener afecto contrario en el desarrollo:

- Falta de experiencia en la construcción de la solución.
- No disponer de los recursos necesarios.
- Falta de supervisión de los desarrolladores en la fase de construcción o programación.
- Incumplimiento de las fechas acordadas para cada fase.

4.2.3. Riesgos del Cliente

Los riesgos relativos a cliente que podrían afectar:

- Persona poco idónea para entregar requerimientos.
- No entrega de la información apropiada y necesaria
- Rotación o cambio de la contraparte responsable.

4.2.4. Riesgos de Implementación

Los riesgos que podrían tener un efecto adverso en la implementación:

- Falta de prolijidad en el desarrollo.
- Falta de información para alguna función
- Presencia de alguna incompatibilidad de HW/SW.
- Errores de diseño.

4.2.5. Matriz de Riesgo

La siguiente matriz de riesgos se basa en el método de Análisis y para ellos se utiliza siguiente gráfico de riesgos :

Riesgo = Probabilidad x Magnitud del Daño

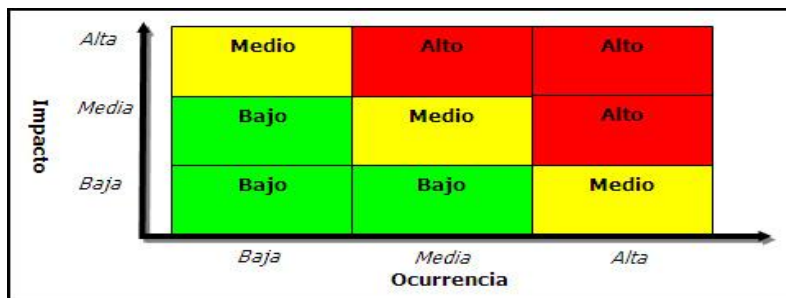


Figura 8 – Matriz de Riesgo, Probabilidad e Impacto

La posibilidad de una amenaza y su impacto pueden tomar los siguientes valores:

1 = Baja, insignificante.

2 = Media.

3 = Alta.

Se identificará el grado de riesgo según color asignado a cada rango de acuerdo a la siguiente tabla de magnitud de impacto :

Descripción	Plan de mitigación	Probabilidad	Impacto	Riesgo
Aumento en los precios del equipamiento necesario	Aplicar margen de pérdida incluido en los detalles del presupuesto	1	2	2
Indisponibilidad de equipamiento para el desarrollo	Se accederá via SSH directo al ordenador	1	2	2
Falta de las competencias técnicas	Supervisión permanente para detectar falta de experiencia	1	3	3
Cambios en los requerimientos	Cada cambio debe quedar registrado	2	2	4
Incumplimiento en las entregas	Revisión de avances diarios	2	3	6
Rotación de los ingenieros desarrolladores	Crear equipos de desarrollo donde se intercambien los requerimientos	3	2	6

Figura 9 – Matriz de Riesgos

5. PLANTEAMIENTO DE LA SOLUCIÓN

5.1 Análisis de la Solución.

5.1.1 Descripción solución propuesta en detalle.

La solución propuesta se basa en diseñar un sistema de control que opere local y remotamente que permita apoyar el control de eventos al interior de un taller automotriz.

El sistema estará compuesto por una red de sensores controlados por ordenadores Raspberry y ellos comunicados sobre una red LAN. Cada sensor operará de manera independiente y adicionalmente registrará en el servidor central dichos eventos. Se dispondrá de un ordenador central que administrará los eventos generados por cada sensor los que serán centralizados en una base de datos para posterior análisis o gestión. Por cada evento crítico se activará por medio de un servicio de monitoreo una instancia de alertas, despliegue en un panel de control, señales auditivas o avisos por correo electrónico.

Cada ordenador el cual controlará sólo un sensor contendrá el software apropiado y configuración adecuada de acuerdo a la naturaleza o función del sensor.

El acceso local y remoto sólo estará permitido para el servidor central, el cual permitirá consultar en tiempo real el estado de la infraestructura o analizar alguna alerta de monitoreo que se haya recibido.

El software estará compuesto por el Sistema Operativo (Versión especial Debian, cliente de conexión base de datos (MySQL), intérprete de script Python y algunos servicios TCP/IP para la comunicación de la red.

La construcción del sistema propiamente tal será desarrollada en el mismo ordenador el cual tendrá todo el software necesario instalado como también las interfaces de su correspondiente sensor.

La escritura del código será usando las herramientas libres disponibles o incluidas en el sistema operativo, el programador podrá escoger usar entre editor nano, el clásico vi o algún otro que no vulnere derechos de autor o que requiera licencia pagada.

En conclusión esta solución aporta un gran valor añadiendo, no sólo el apoyo al control de la infraestructura sino también al confort y seguridad de las personas que trabajan al interior del taller, siendo un gran apoyo para las actividades cotidianas que se llevan a cabo día a día. Ello es debido, como ya se ha mencionado, al mayor control que brindan este tipo de sistemas.

5.1.2 Evaluación de Herramienta de desarrollo e Implementación.

El proceso de desarrollo de aplicaciones de software normalmente involucra varias etapas. Los programas de software usan muchos lenguajes y tecnologías diferentes, con las herramientas típicamente creadas para tecnologías específicas. El desarrollo de software puede ser una actividad compleja y larga, por lo que las herramientas disponibles pueden reducir el estrés y aumentar el desempeño tanto de desarrolladores como de las aplicaciones resultantes. Hay herramientas disponibles para cada etapa en el proceso de desarrollo de software.

Las principales herramientas de desarrollo son las siguientes :

Herramienta	Versión	Licencia
Debian for RASPBERRY Pi	3.1	Libre
MySQL	5.6	Libre
Apache WebServer	2	Libre

PHP		5	Libre
Python language		6	Libre
Estación de trabajo desarrollo Lenovo	64 bits		Licencia pagada

Figura 10 – Herramientas para el desarrollo

5.1.3 Diagrama de Arquitectura de Solución Propuesta.

La arquitectura a implementar será el modelo Cliente-Servidor, en dónde cada función será dividida por cada cliente disponible y el Servidor centralizará la información y gestionará los eventos por cada condición, además este último es quién tendrá el rol de atender las solicitudes tanto locales como remotas.

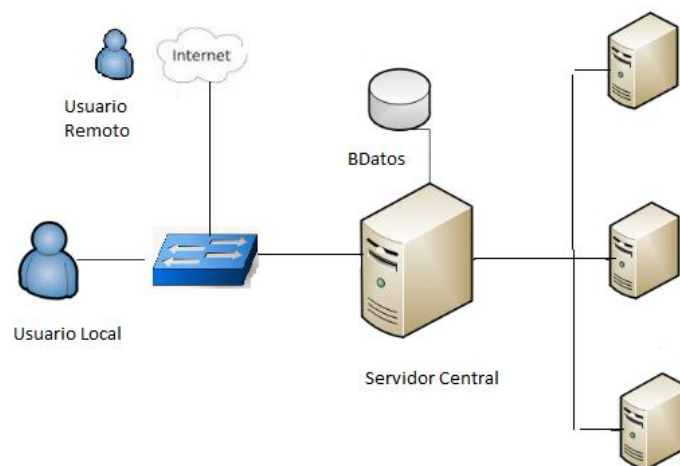


Figura 11 – Diagrama de Arquitectura

5.1.4 Diagrama proceso mejorado (BPMN)

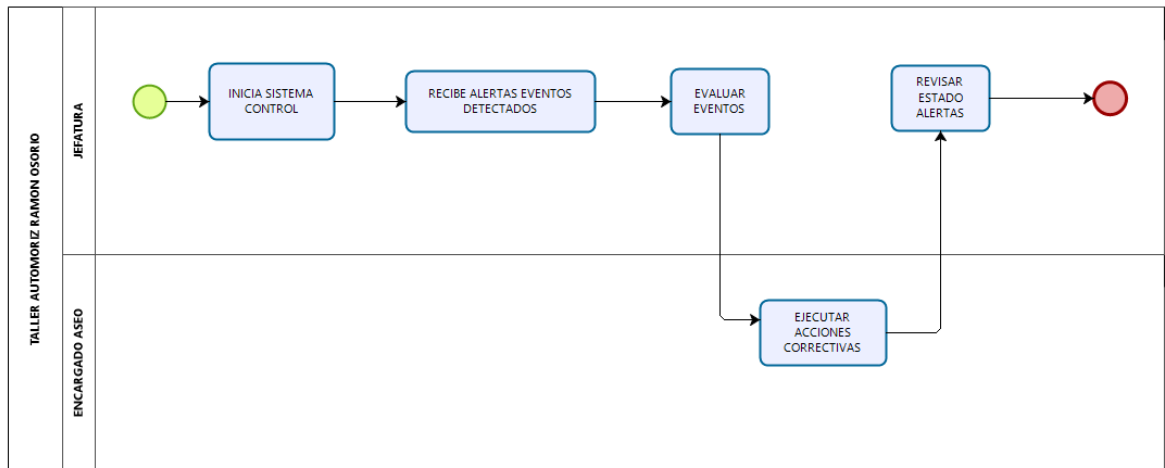


Figura 12 – Diagrama Proceso Mejorado

Etapas del Proceso Mejorado

Jefatura Inicia sistema : La jefatura activa o ejecuta el sistema de monitoreo.

Sistema : Queda activo monitoreando en tiempo real la infraestructura apoyado en los ordenadores/sensores configurados.

Jefatura : Recibe en tiempo real alertas por cambios de estado o umbrales que hayan llegado a su valor configurado.

Jefatura : Ejecuta acciones de acuerdo al tipo de alerta recibida.

5.2 Metodología Aplicada.

5.2.1 Comparativa de metodologías asociadas al proyecto.

	Metodologia Incremental	Metodologia en Espiral	Metodologia Extreme Programming
Definición	Reemplazar un componente antiguo por uno nuevo	Combinación de metodologías prototipos y casacada	Enfoque orientado al usuario final
Desventajas	Los requisitos cambian permanentemente	Se podrian generar multiples prototipos del sistema	Generación de muchas entregas parciales
Características	Se desarrolla para cumplir requisitos específicos y posteriores	Evaluación, planificación y desarrollo del prototipo	Entregas tempranas y continuas de Software

Figura 13 – Cuadro Comparativo Metodologías

5.2.2 Justificación de la metodología seleccionada.

En relación a la metodología a utilizar para el desarrollo de este proyecto será la denominada eXtreme Programming (XP) , la cual es considerada una metodología ágil ya que dentro de sus principales características podemos mencionar satisfacer la necesidad o al cliente por medio de entregas tempranas y continuas de software, centrarse en la solución más que en preparar documentación y desarrollo sostenible por mencionar algunas.

Para mayor claridad esta metodología es altamente aplicable ya que el desarrollo no implica la construcción de un inmenso sistema informático, se dispondrá de un repositorio de módulos pre-desarrollados los cuales de acuerdo a los indicado por los clientes (o usuario final) será adaptado o ajustado según se especifique.

El universo de dispositivos es bastante amplio, pero los estados para cada uno son finitos o bien acotados, a excepción de los equipos que manejen un rango de valores posible (ejemplo un sensor de temperatura).

Por lo anterior de acuerdo a lo indicado o especificado se buscará el componente más apropiado y se harán los ajustes necesarios si es que ellos fueran necesarios, ya que por ejemplo detectar cambios de temperatura no requerirá de un desarrollo específico ya que lo existente cubriría la necesidad.

Gracias a una metodología ágil el desarrollo, entregas parciales y correcciones rápidas podrán ser entregadas en tiempo y forma.

5.2.3 Metodología de la Administración.

Para una eficiente administración del proyecto y así poder asegurar el correcto avance , manteniendo los plazos y calidad se acordaron revisiones cada 10 días. En estas reuniones se revisan los problemas, inquietudes o puntos de mejora, los cuales podrían involucrar el mantener o alterar la carta Gantt del proyecto.

Con lo anterior se conseguirá la transparencia del proyecto, ya que se irán informando las distintos eventos que se vaya presentando, avances y contratiempos que aparte de mantener informado al cliente se conseguirá ir despejando incógnitas y conservar una buena atmósfera al interior de los equipos de trabajo ya que todos dispondrán del estado detallado del proyecto.

5.2.4 Carta Gantt.

ACTIVIDADES				Dias de Duración
ID	Nombre de la Tarea	Inicio	Término	
1	Inicio actividades	01-01-2019	01-01-2019	1
	ANALISIS			
2	Definición de requerimientos	02-01-2019	31-01-2019	30
	DISEÑO			
3	Modelamiento Base de Datos	02-02-2019	15-02-2019	15
4	Modelamiento Funciones	15-02-2019	28-02-2019	13
	CONSTRUCCION			
5	Configuración ordenador	01-03-2019	10-03-2019	10
6	Preparación prototipo	10-03-2019	10-04-2019	30
7	Entrega prototipo	15-04-2019	20-04-2019	5
8	Pruebas con usuario	25-04-2019	10-05-2019	15
	ENTREGA FINAL			
9	Entrega prototipo definitivo	10-05-2019	15-05-2019	5
10	Pruebas con usuario	15-05-2019	30-05-2019	15
	PUESTA EN MARCHA			
11	Entrega versión final	01-06-2019	10-06-2019	10
12	Entrega documentación	10-06-2019	10-07-2019	30
13	CIERRE PROYECTO			

Figura 14 – Cuadro Carta Gantt

5.3 Obtención de Requerimientos.

5.3.1 Metodología Aplicada a la Toma de Requerimientos.

La metodología utilizada considerando la naturaleza del proyecto, no requiere intervención adicional de otras entidades, ya que bastaría con que una persona especializada, para nuestro caso será el Jefe de Taller quien posee la experiencia en la operación del taller en el tema levante el requerimiento para después analizarlo y derivarlo al área de mantenimiento.

Las características de la metodología son las siguientes:

- Es una tarea precisa para obtener información clara y oportuna de diversas fuentes.
- Es importante que los requerimientos sean abordados con las personas más adecuadas logrando así la mayor cantidad de información , clara y oportuna.
- Se utilizará una plantilla en donde queden registrados los requerimientos como así los detalles relevantes.
- Se utilizará un catálogo donde se muestre una plantilla de requerimientos ya desarrollados de modo de confirmar y estandarizar las necesidades.

5.3.2 Requerimientos Funcionales.

La solución propuesta se basa en diseñar un sistema de acceso y control de dispositivos de diversa naturaleza para un taller automotriz, el que claro debe contar con las condiciones mínimas necesarias. (energía, temperatura, internet, etc.)

5.3.2.1 Requerimientos Funcionales de Acceso

- Controlar el acceso permitiendo o denegando la conexión con la plataforma.
- Autorizar acceso de acuerdo a los privilegios asignados.
- Registrar en bitácora el historial de accesos.
- Administración de seguridad.

Número de requisito	REQ-FUNCIONAL-001		
Nombre de requisito	Habilitar acceso remoto		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Características	Poder ejecutar eventos sobre dispositivos se debe acceder a la red.		
Descripción del requisito	Por las características básicas el acceso debe ser remoto, localmente no requiere autenticación.		
Prioridad del requisito	<input type="checkbox"/> Alta/Esencial	<input checked="" type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/Opcional

Número de requisito	REQ-FUNCIONAL-002		
Nombre de requisito	Habilitar registro bitácora accesos		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Características	Registrar en bitácora cada acceso al sistema,		
Descripción del requisito	Por seguridad y control se habilita el registros de los accesos..		
Prioridad del requisito	<input type="checkbox"/> Alta/Esencial	<input checked="" type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/Opcional

5.3.3 Requerimientos No Funcionales.

Los requerimientos no funcionales que debe cubrir este sistema se entregan a continuación.

Número de requisito	REQ-NO-FUNCIONAL-001		
Nombre de requisito	Nuevo Ordenador		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Características	Habilitación nuevo ordenador		
Descripción del requisito	Configurar nuevo ordenador de acuerdo a una nueva necesidad.		

Prioridad del requisito	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
	Alta/Esencial	Media/Deseado	Baja/Opcional

- El uso de esta aplicación está asociada exclusivamente a cada usuario o personal debidamente autorizado.
- Debe ser compatible con dispositivos móviles (WEB) y navegadores.
- El acceso al sistema debe operar tanto local como remotamente.

Número de requisito	REQ-NO-FUNCIONAL-002		
Nombre de requisito	Productos Base		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Características	Instalación de productos de software		
Descripción del requisito	Instalar Software a una nueva necesidad.		
Prioridad del requisito	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> Baja/Opcional
	Alta/Esencial	Media/Deseado	

Número de requisito	REQ-NO-FUNCIONAL-003		
Nombre de requisito	Crear servidor de base de datos MySql.		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Características	Acceso a la información de BD MySql.		
Descripción del requisito	Habilitar servidor de BD MySql dentro de un nuevo odenador		
Prioridad del requisito	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/> Baja/Opcional
	Alta/Esencial	Media/Deseado	

5.3.4 Requerimientos de Seguridad.

El sistema contará con acceso controlado, un módulo de administración que permita mantener los usuarios autorizados para así garantizar el seguro ingreso e interacción con el sistema.

Número de requisito	REQ-SEGURIDAD-001		
Nombre de requisito	Control de acceso.		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Características	Se requiere la creación de cuentas de acceso al sistema.		
Descripción del requisito	El sistema controlará el acceso y lo permitirá solamente a usuarios autorizados. Los usuarios deben ingresar al sistema con un nombre de usuario y contraseña.		
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/Opcional
Número de requisito	REQ-SEGURIDAD-002		
Nombre de requisito	Alerta de seguridad.		
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción	
Características	Se enviará al administrador semanalmente nómina de accesos válidos y fallidos.		
Descripción del requisito	El sistema generará nomina detallada de accesos.		
Prioridad del requisito	<input checked="" type="checkbox"/> Alta/Esencial	<input type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/Opcional

5.3.5 Requerimientos de Mantenición.

- Analizar tabla de sucesos.
- Analizar log de accesos.
- Analizar estado de las interfaces y sistema en general.

Número de requisito	REQ-FUNCIONAL-004	
Nombre de requisito	Habilitar opciones de análisis y gestión	
Tipo	<input checked="" type="checkbox"/> Requisito	<input type="checkbox"/> Restricción

Características	Habilitar funciones que permitan gestión.		
Descripción del requisito	Crear opciones de administración.		
Prioridad del requisito	<input type="checkbox"/> Alta/Esencial	<input checked="" type="checkbox"/> Media/Deseado	<input type="checkbox"/> Baja/Opcional

5.3.6 Especificación de Requerimientos.

Para el control de los requerimientos se dispondrá de una plantilla estándar la cual incluirá los datos necesarios para la clara comprensión y exactitud en el desarrollo de la solicitud.

6. DISEÑO DEL SISTEMA

6.1. Modelamiento UML

6.1.1. Diagrama Casos de Uso.

Un caso de uso es una técnica que permite capturar requisitos potenciales de un sistema nuevo o de la actualización de un software. Es una secuencia de interacciones que se van a desarrollar entre el sistema y sus actores.

Se utiliza para capturar el comportamiento deseado del sistema sin necesidad de especificar un lenguaje técnico en la forma en que se planea implementar ese comportamiento.

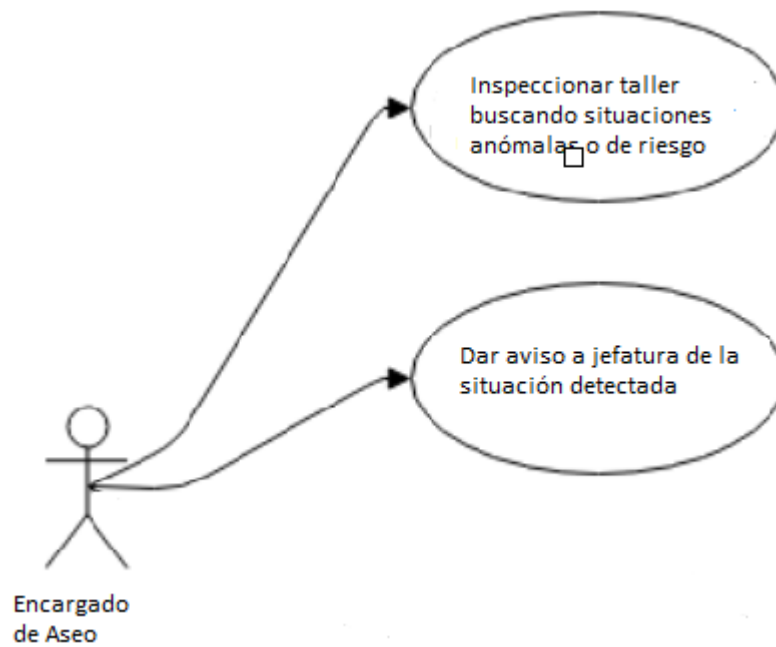


Figura 15 – Diagrama de Uso Verificación Visual

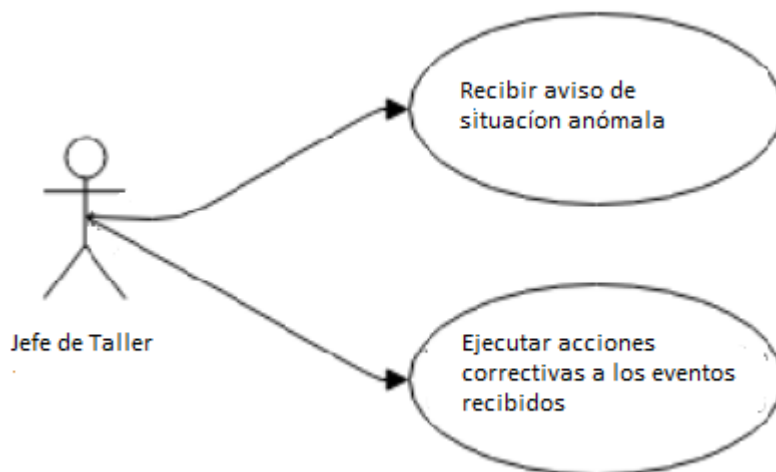


Figura 16 – Diagrama de Uso Ejecutar acciones

6.1.2. Documentación Casos de Uso.

Sistema de Monitoreo

DESCRIPCION DE CASOS DE USO

Nombre:	Inspección Visual
Alias:	
Actores:	Encargado de Aseo
Función:	Ejecutar Revisión Manual.
Descripción:	El encargado del aseo debe ejecutar con cierta periodicidad rondas visuales con el fin de detectar situaciones anormales, de riesgo o que puedan generar fallas en la infraestructura o riesgo a las personas.
Referencias:	

Figura 17 – Función del Diagrama de Uso Verificación Visual

Sistema de Monitoreo	
DESCRIPCION DE CASOS DE USO	
Nombre:	Ejecución de acciones
Alias:	
Actores:	Jefe de Taller
Función:	Ejecutar acciones correctivas.

Descripción:	El Jefe de Taller recibido el aviso con la situación anómala detectada deberá ejecutar determinadas acciones que apunten a corregir el evento que podría estar generando el riesgo.
Referencias:	

Figura 18 – Función del Diagrama de Uso Ejecutar Acciones

6.1.3. Diagrama de Componentes.

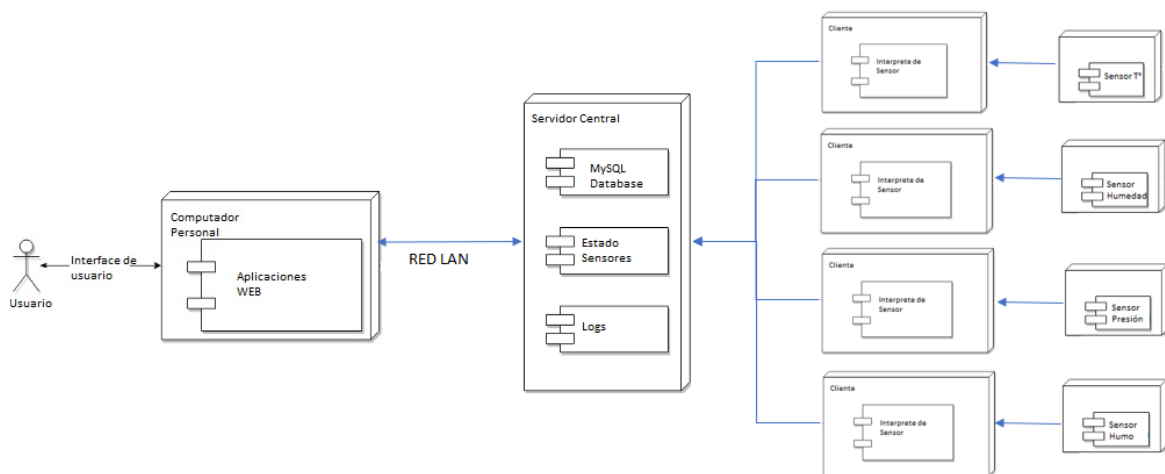


Figura 19 –Diagrama de Componentes

6.2. Modelamiento de Datos

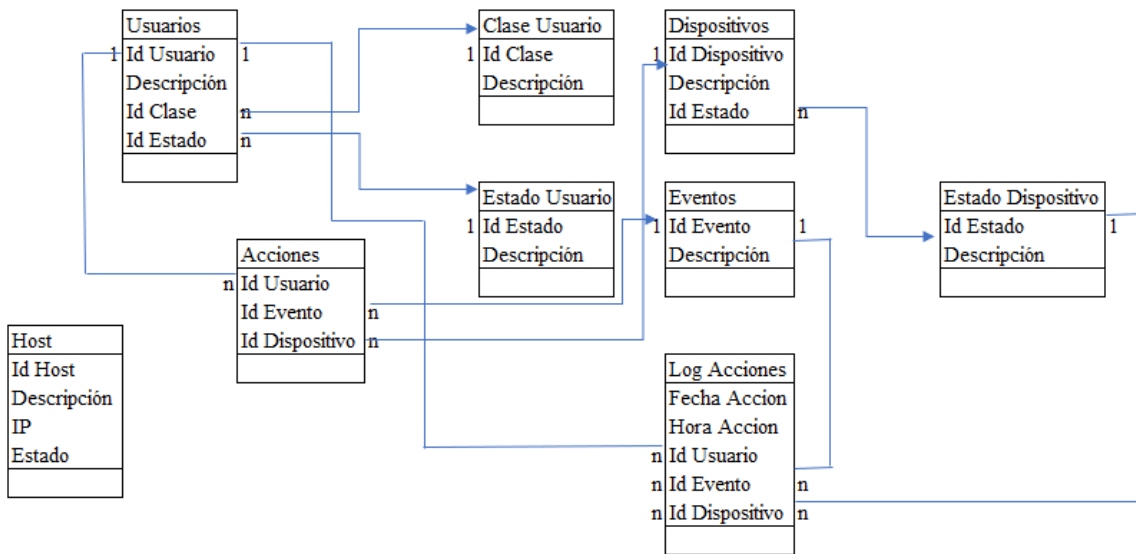



Figura 20 – Modelo de Datos

7. CONSTRUCCIÓN PROTOTIPO FUNCIONAL

El prototipo de acceso al sistema muestra una ventana con el título "Sistema de Monitoreo - Taller Ramon Osorio". En la parte superior derecha hay un icono de un coche. El texto "Acceso al Sistema" está acompañado de un icono de una llave. Hay dos campos de entrada para "Usuario" y "Contraseña". En la parte inferior hay dos botones: "Aceptar" y "Cancelar".

Figura 21 – Prototipo de acceso al Sistema



Sistema de Monitoreo - Taller Ramon Osorio

Administración  

Usuarios
 Dispositivos
 Alertas
 Sistema

Figura 22 – Prototipo formulario Administración de Usuarios



Sistema de Monitoreo - Taller Ramon Osorio

Usuarios  

Creación
 Modificación
 Eliminación
 Bitácora de accesos

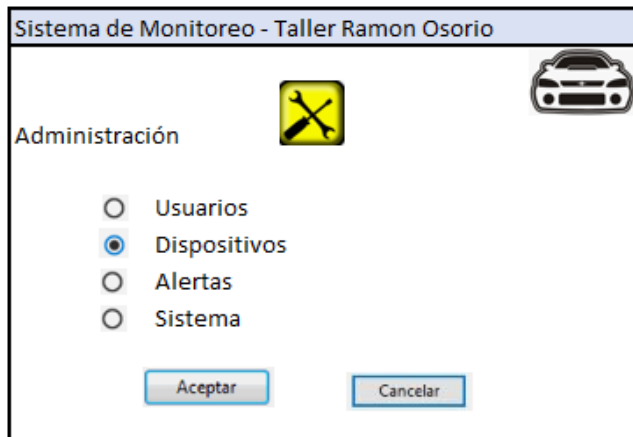
Figura 23 – Prototipo formulario Alta de un usuario

Sistema de Monitoreo - Taller Ramon Osorio



Creación  

Nombre Descripción
 Clase
 Contraseña

Figura 24 – Prototipo formulario ingreso datos de usuario



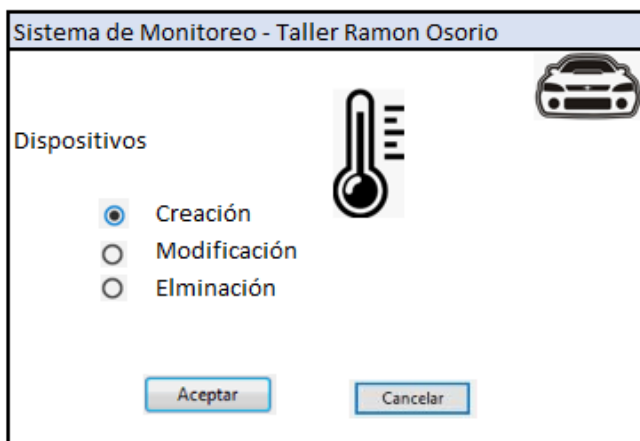
Sistema de Monitoreo - Taller Ramon Osorio

Administración  



Usuarios
 Dispositivos
 Alertas
 Sistema

Aceptar Cancelar

Figura 25 – Prototipo formulario Administración de Dispositivos



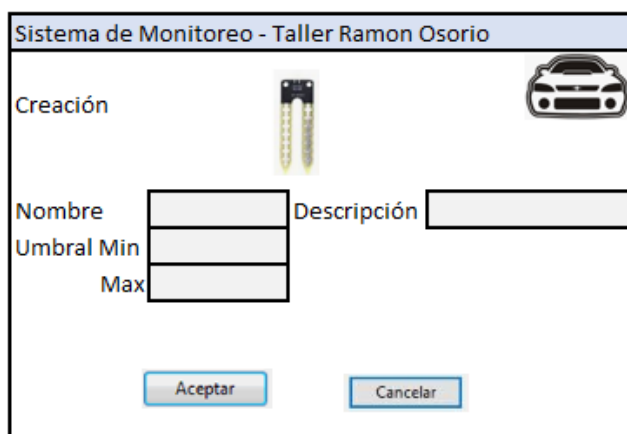
Sistema de Monitoreo - Taller Ramon Osorio

Dispositivos  



Creación
 Modificación
 Eliminación

Aceptar Cancelar

Figura 26 – Prototipo formulario Alta de un dispositivo



Sistema de Monitoreo - Taller Ramon Osorio

Creación  

Nombre Descripción
 Umbral Min
 Max

Aceptar Cancelar

Figura 27 – Prototipo formulario ingreso datos de dispositivo

Sistema de Monitoreo - Taller Ramon Osorio


Monitoreo

En línea
 Historial

Figura 28 – Prototipo formulario Monitoreo de sensores

Sistema de Monitoreo - Taller Ramon Osorio

Estado sensores 








Ubicación	Estado	Valor	
T° Compresor		75°	Ver Más Detalles
Humedad Lubricantes		30%	
Gases CO2		0%	
Humedad Pozo		90%	Ver Más Detalles
Nivel colector		60%	Ver Más Detalles

Figura 29 – Prototipo formulario Estado de sensores

Sistema de Monitoreo - Taller Ramon Osorio

Detalles  T° Compresor 

FECHA dd/mm/aa
HORA 12:30

08:00-09:00-10:00-11:00-12:00-13:00

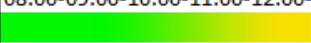
 75°

Figura 30 – Prototipo formulario detallado por sensor seleccionado.

8. DISEÑO DE PRUEBAS DEL SOFTWARE

Nombre de la prueba:	Ingreso al Sistema
Objetivo de la prueba:	Verificar el correcto funcionamiento del ingreso al sistema
Prueba:	<ul style="list-style-type: none"> ● Una vez desplegado el formulario inicial se ingresarán los datos que corresponde al Usuario, contraseña asociada y pulsar <Aceptar> ● Se ingresarán usuarios inexistentes para validar la correcta búsqueda ● Se ingresarán contraseñas incorrectas para validar la correcta asociación ● Se ingresarán usuarios existentes para validar la búsqueda ● Se ingresarán usuarios y su contraseña asociada para validar la entrada al sistema.
Herramientas necesarias:	Ninguna
Criterio del éxito:	Se revisará la tabla de usuarios en la base de datos y se verificará que los registros usados contengan los valores de acuerdo al resultado de las pruebas.

Figura 31 – Plan de Pruebas – Ingreso al Sistema

Nombre de la prueba:	Creación de Usuarios
Objetivo de la prueba:	Verificar el correcto funcionamiento en el alta de Usuarios
Prueba:	<ul style="list-style-type: none"> ● Una vez desplegado el formulario de creación de usuarios, ingresar nombre, descripción, seleccionar clase y finalmente la contraseña ● No se ingresarán todos los datos para que se exijan los faltantes ● Se ingresarán usuarios existentes para validar la búsqueda
Herramientas necesarias:	Ninguna
Criterio del éxito:	Se revisará la tabla de usuarios en la base de datos y se verificará la correcta existencia de los datos probados.

Figura 32 – Plan de Pruebas – Creación de Usuarios

Nombre de la prueba:	Ingreso de dispositivos
Objetivo de la prueba:	Verificar la correcta creación de dispositivos
Prueba:	<ul style="list-style-type: none"> ● Una vez desplegado el formulario de alta de dispositivos marcar

	<p>opción creación y pulsar <Aceptar></p> <ul style="list-style-type: none"> ● Se ingresarán dispositivos inexistentes para validar la correcta búsqueda ● No se ingresará la descripción para que sea exigida ● Se ingresarán umbrales erróneos
Herramientas necesarias:	Ninguna
Criterio del éxito:	Se revisará la tabla de dispositivos en la base de datos y se verificará que los registros usados contengan los valores de acuerdo al resultado de las pruebas.

Figura 33 – Plan de Pruebas – Creación de Dispositivos

Nombre de la prueba:	Lectura de sensores
Objetivo de la prueba:	Verificar la correcta operación de los sensores
Prueba:	<ul style="list-style-type: none"> ● Una vez desplegado el formulario de monitoreo, seleccionar "En línea" y pulsar <Aceptar> ● Se forzaré una situación en el cambio de estado en un sensor ● Se desconectará un sensor
Herramientas necesarias:	Ninguna
Criterio del éxito:	Se revisará el correcto comportamiento de los sensores en concordancia con los umbrales y estado del sensor.

Figura 34 – Plan de Pruebas – Lectura de Sensores

9. CONCLUSIONES DEL PROYECTO

9.1. Conclusiones.

Finalmente se presentan las conclusiones resultantes luego de diseñar una solución o Sistema de control .

Los objetivos especificados al inicio son logrados satisfactoriamente al implementar la solución que permite al usuario monitorear una red de sensores situados al interior de un Taller Automotriz..

Producto de esta solución se puede conseguir el incremento de la seguridad y monitoreo de los eventos que se presenten en dicho Taller.

En conclusión por lo anteriormente señalado se considera que la implementación de este proyecto implica adquirir un valor agregado a la seguridad apoyado en tecnología y equipamiento de fácil acceso.

9.2. Perspectivas Futuras

Producto de esta implementación se espera sea usada a futuro como plataforma inicial para el desarrollo de Soluciones del tipo Domóticas, con ella se podrá aplicar escalamiento para así conseguir ampliar la granja de dispositivos e ir incorporando nuevas funcionalidades.

Dependiendo de la tecnología que vaya emergiendo se podría ir considerando aplicar este modelo a escenarios más complejos o industriales.

BIBLIOGRAFIA

Fairley R. Ingeniería de Software

Pressman, R.S. Ingeniería del Software, enfoque practico

DOMÓTICA PARA VIVIENDAS Y EDIFICIOS. Autor: Werner Harke (traducido al español). Año 2010

Título: Learning web design Autor: Jennifer Niederst Robbins Editorial: O'Reilly

Automatización y control

2004 - McGraw Hill Interamericana - Prácticas de laboratorio

6ª Edición / 276 págs. / Rústica / Castellano / Libro

EL ABC DE LA AUTOMATIZACION

SISTEMAS DE CONTROL AUTOMÁTICO; por Patricio Abarca

<http://www.aie.cl/files/file/comites/ca/abc/sistemas-de-control-automatico.pdf>

Teoría del Control Automático

http://www.profesormolina.com.ar/tecnologia/teoria_con_aut.htm