

**DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE SISTEMA DE
INFORMACIÓN EN LA WEB PARA LAS EMPRESAS PRESTADORAS DEL
SERVICIO PUBLICO DE TAXIS EN PEREIRA.**

**PAOLA ANDREA AGUDELO RENDÓN
DENIER ADOLFO ALCALDE GUEVARA**

**UNIVERSIDAD TECNOLÓGICA DE PEREIRA
FACULTAD DE INGENIERIAS ELÉCTRICA, ELECTRÓNICA, FÍSICA Y
CIENCIAS DE LA COMPUTACIÓN
PROGRAMA INGENIERIA DE SISTEMAS Y COMPUTACIÓN
PEREIRA
2015**

**DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE SISTEMA DE
INFORMACIÓN EN LA WEB PARA LAS EMPRESAS PRESTADORAS DEL
SERVICIO PUBLICO DE TAXIS EN PEREIRA.**

**PAOLA ANDREA AGUDELO RENDÓN
DENIER ADOLFO ALCALDE GUEVARA**

**PROYECTO DE GRADO PARA OPTAR AL TITULO DE
INGENIERO DE SISTEMAS Y COMPUTACIÓN**

**CARLOS ALBERTO OCAMPO SEPÚLVEDA
INGENIERO DE SISTEMAS Y COMPUTACIÓN
ASESOR DE PROYECTO DE GRADO**

**UNIVERSIDAD TECNOLÓGICA DE PEREIRA
FACULTAD DE INGENIERIAS ELÉCTRICA, ELECTRÓNICA, FÍSICA Y
CIENCIAS DE LA COMPUTACION
PROGRAMA INGENIERIA DE SISTEMAS Y COMPUTACIÓN
PEREIRA**

2015

Nota de aceptación:

Firma del presidente del jurado

Firma del jurado

Firma del jurado

AGRADECIMIENTOS

Quiero agradecer a Dios por iluminarme y cuidarme en cada paso que doy, a mis padres y hermanita por confiar en mí y mis capacidades, dándome siempre una palabra positiva que me anima a continuar y ser mejor, a mi novio quien con su conocimiento me aportó muchas y valiosas ideas para culminar con éxito este proyecto, a todos mis compañeros por sus diferentes contribuciones en cada etapa de este proceso siendo más que compañeros amigos, a Denier Adolfo Alcalde por ser una persona responsable, dedicada y con quien establecí una gran equipo de trabajo. Doy especiales agradecimientos a todos los profesores que a lo largo de mi carrera me han dado herramientas con las que pude desarrollar este trabajo, especialmente a nuestro asesor Carlos Alberto Ocampo.

Paola Andrea Agudelo Rendón

Agradezco a Dios por darme la vida, a mis padres por su apoyo incondicional durante estos años por su sacrificio, esfuerzo y dedicación en mi formación académica y personal, a mis hermanos con quienes crecí y atravesé diferentes etapas de la vida, doy gracias a la universidad por dar el espacio y las herramientas para la formación integral de las próximas generaciones, a mis profesores, compañeros y demás personas que de diferentes formas han aportado conocimiento en mi vida, a Paola Andrea Agudelo por estos 5 años de amistad.

Denier Adolfo Alcalde Guevara

TABLA DE CONTENIDO

1. INTRODUCCIÓN	14
2. CAPITULO I – GENERALIDADES	15
2.1. DESCRIPCIÓN DEL PROBLEMA.....	15
3. JUSTIFICACIÓN	16
4. OBJETIVOS	17
4.1. OBJETIVO GENERAL.....	17
4.2. OBJETIVOS ESPECÍFICOS	17
5. ALCANCE DEL PROTOTIPO A DESARROLLAR.....	18
6. CAPÍTULO II - MARCO REFERENCIAL.....	19
6.1 MARCO TEÓRICO	19
6.1.1 INGENIERÍA DEL SOFTWARE.....	19
6.1.2 ¿QUÉ ES UN PROCESO DE SOFTWARE?.....	20
6.1.3 MODELOS DE PROCESOS DE DESARROLLO DE SOFTWARE.....	21
6.1.3.1 EL MODELO EN CASCADA	21
6.1.3.2 DESARROLLO EVOLUTIVO	23
6.1.3.3 DESARROLLO BASADO EN COMPONENTES.....	27
6.1.4 METODOLOGIAS AGILES.....	28
6.1.4.1 DYNAMIC SYSTEMS DEVELOPMENT METHOD (DSDM).....	30
6.1.4.2 CRYSTAL METHODOLOGIES.....	31
6.1.4.3 ADAPTATIVE SOFTWARE DEVELOPMENT (ASD).....	32
6.1.4.4 EXTREME PROGRAMMING (XP)	33
6.1.4.5 SCRUM.....	34
6.1.5 LEVANTAMIENTO Y ANALISIS DE REQUERIMIENTOS.....	37
6.1.5.1 HISTORIA DE USUARIO.....	38
6.1.6 DISEÑO DE SOFTWARE	39
6.1.6.1 LENGUAJE UNIFICADO DE MODELADO (UML).....	39
6.1.6.1.1 VISTA DE ESCENARIOS (CASOS DE USO).....	41
6.1.6.1.2 VISTA LOGICA (DIAGRAMA DE CLASES)	42

6.1.6.1.3	VISTA DE DESARROLLO (DIAGRAMA DE COMPONENTES)	42
6.1.6.1.4	VISTA DE PROCESOS (DIAGRAMA DE SECUENCIA).....	43
6.1.6.1.5	VISTA FISICA (DIAGRAMA DE DESPLIEGUE).....	43
6.1.6.2	MODELO RELACIONAL.....	43
6.1.6.3	ARQUITECTURA DE SOFTWARE.....	44
6.1.6.3.1	MODELO VISTA CONTROLADOR.....	44
6.1.7	CODIFICACIÓN	45
6.1.8	PRUEBAS	46
6.1.9	ENTREGA	48
6.1.10	ESTANDARES WEB	48
6.1.10.1	HTML (HYPER TEXT MARKUP LENGUAJE).....	48
6.1.10.2	XML (EXTENSIBLE MARKUP LENGUAJE).....	49
6.1.10.3	JAVASCRIPT	49
6.1.10.4	CSS	49
6.1.10.5	PHP (PHP: Hypertext Preprocessor)	50
6.1.10.6	FRAMEWORK	50
6.1.10.7	CODEIGNITER.....	51
6.1.10.8	MY SQL	51
7	CAPITULO III-OTRAS CONSIDERACIONES	52
7.1	MARCO LEGAL.....	55
8	CAPÍTULO IV-DEFINICIÓN DE LOS REQUERIMIENTOS DEL SISTEMA.....	56
8.1	SPRINT 1.....	56
8.1.1	HISTORIAS DE USUARIO	56
8.1.2	DISEÑO Y MODELADO DEL SISTEMA	64
8.1.2.1	VISTA DE ESCENARIOS (CASOS DE USO).....	64
8.1.2.2	VISTA DE PROCESOS (DIAGRAMAS DE SECUENCIA).....	78
8.2	SPRINT 2.....	87
8.2.1	HISTORIAS DE USUARIO	87
8.2.2	DISEÑO Y MODELADO DEL SISTEMA	90
8.2.2.1	VISTA DE ESCENARIOS (CASOS DE USO).....	90
8.2.2.2	VISTA DE PROCESOS (DIAGRAMAS DE SECUENCIA)	97
8.3	SPRINT 3.....	102

8.3.1	HISTORIAS DE USUARIO	102
8.3.2	DISEÑO Y MODELADO DEL SISTEMA	105
8.3.2.1	VISTA DE ESCENARIOS (CASOS DE USO)	106
8.3.2.2	VISTA DE PROCESOS (DIAGRAMAS DE SECUENCIA)	121
8.4	VISTA LOGICA (DIAGRAMA DE CLASES)	131
8.5	VISTA DE DESARROLLO (DIAGRAMA DE COMPONENTES)	132
8.6	VISTA FISICA (DIAGRAMA DE DESPLIEGUE)	133
8.7	MODELO RELACIONAL BASE DE DATOS	134
9.	CONCLUSIONES	135
10.	RECOMENDACIONES	136
11.	BIBLIOGRAFÍA	137

LISTA DE DIAGRAMAS

DIAGRAMA CASO DE USO: USUARIO REGISTRADO	64
DIAGRAMA CASO DE USO: USUARIO ADMINISTRADOR.....	70
DIAGRAMA DE SECUENCIA # 1: AUTENTICARSE	78
DIAGRAMA DE SECUENCIA #2: GESTIONAR DATOS BÁSICOS	79
DIAGRAMA DE SECUENCIA # 3: CAMBIO DE CONTRASEÑA	80
DIAGRAMA DE SECUENCIA #4: RECUPERACIÓN DE CONTRASEÑA.....	80
DIAGRAMA DE SECUENCIA #5: CREAR USUARIO ADMINISTRADOR.....	81
DIAGRAMA DE SECUENCIA #6: ELIMINACIÓN CUENTA	82
DIAGRAMA DE SECUENCIA #7: CREACIÓN DE USUARIO OPERARIO	82
DIAGRAMA DE SECUENCIA #8: ELIMINAR USUARIO OPERARIO	83
DIAGRAMA DE SECUENCIA #9: CONSULTA INFORMACIÓN OPERARIO.....	83
DIAGRAMA DE SECUENCIA #10: ASIGNACIÓN DE TURNOS.....	84
DIAGRAMA DE SECUENCIA # 11 A: INFORMACIÓN DE LA EMPRESA.....	85
DIAGRAMA DE SECUENCIA # 11B: GENERACIÓN DE REPORTES	86
DIAGRAMA CASO DE USO: OPERARIO.....	91
DIAGRAMA DE SECUENCIA #12: CONSULTAR TURNOS	97
DIAGRAMA DE SECUENCIA #13: REVISIÓN DE ALARMAS	98
DIAGRAMA DE SECUENCIA #14: AGREGAR CONDUCTOR	99
DIAGRAMA DE SECUENCIA #15: ELIMINAR CONDUCTOR	99
DIAGRAMA DE SECUENCIA #16: CONSULTA INFORMACIÓN CONDUCTOR	100
DIAGRAMA DE SECUENCIA #17: GESTIÓN DE TURNOS	101
DIAGRAMA DE CASO DE USO: OPERARIO	106
DIAGRAMA DE CASO DE USO: INVITADO	118
DIAGRAMA DE SECUENCIA #18: ACTUALIZAR INFORMACIÓN CONDUCTOR....	121
DIAGRAMA DE SECUENCIA #19: AGREGAR DOCUMENTACIÓN DEL CONDUCTOR	122
DIAGRAMA DE SECUENCIA #20: ASIGNACIÓN DE VEHÍCULOS	123
DIAGRAMA DE SECUENCIA #21: QUEJAS, COMPARENDOS Y SINIESTROS.....	124
DIAGRAMA DE SECUENCIA #22: AGREGAR VEHÍCULO	125
DIAGRAMA DE SECUENCIA #23: ELIMINAR VEHÍCULO	125

DIAGRAMA DE SECUENCIA #24: CONSULTA INFORMACIÓN DE VEHÍCULO	126
DIAGRAMA DE SECUENCIA #25: ACTUALIZAR INFORMACIÓN VEHÍCULO.....	127
DIAGRAMA DE SECUENCIA #26: MANTENIMIENTO	128
DIAGRAMA DE SECUENCIA #27: SERVICIOS REALIZADOS	129
DIAGRAMA DE SECUENCIA #28: CONSULTA INFORMACIÓN (INVITADO)	129
DIAGRAMA DE SECUENCIA #29: REGISTRO QUEJAS	130

LISTA DE FIGURAS

Figura 1: modelo en cascada.....	22
Figura 2: desarrollo evolutivo	23
Figura 3: el paradigma de hacer prototipos.....	25
Figura 4: modelo en espiral	26
Figura 5: desarrollo basado en componentes	27
Figura 6: flujo del proceso scrum	36

LISTA DE TABLAS

Tabla 1: niveles de pruebas	47
Tabla 2: Información general taxis web gps, administración de vehículos	53
Tabla 3: información general admintaxi y gtaxi.....	53
Tabla 4: información general Logirastreos, SILOG	54
Tabla 5: Historia de usuario 1.....	56
Tabla 6: Historia de usuario 2.....	57
Tabla 7: Historia de usuario 3.....	57
Tabla 8: Historia de usuario 4.....	58
Tabla 9: Historia de usuario 5.....	58
Tabla 10: Historia de usuario 6.....	59
Tabla 11: Historia de usuario 7.....	60
Tabla 12: Historia de usuario 8.....	60
Tabla 13: Historia de usuario 9.....	61
Tabla 14: Historia de usuario 10.....	61
Tabla 15: Historia de usuario 11.....	62
Tabla 16: Historia de usuario 15.....	62
Tabla 17: Historia de usuario 16.....	63
Tabla 18: Historia de usuario 17.....	63
Tabla 19: Caso de uso # 1	65
Tabla 20: Caso de uso # 2	67
Tabla 21: Caso de uso # 3	68
Tabla 22: Caso de uso # 4	70
Tabla 23: Caso de uso # 5	71
Tabla 24: Caso de uso # 6	72
Tabla 25: Caso de uso # 7	73
Tabla 26: Caso de uso # 8	74
Tabla 27: Caso de uso # 9	75
Tabla 28: Caso de uso # 10	77

Tabla 29: Caso de uso # 11	78
Tabla 30: Historia de Usuario 12	87
Tabla 31: Historia de Usuario 13	88
Tabla 32: Historia de Usuario 18	88
Tabla 33: Historia de Usuario 19	89
Tabla 34: Historia de Usuario 20	89
Tabla 35: Historia de Usuario 21	90
Tabla 36: Caso de Uso # 12.....	92
Tabla 37: Caso de Uso # 13.....	93
Tabla 38: Caso de Uso # 14.....	94
Tabla 39: Caso de Uso # 15.....	95
Tabla 40: Caso de Uso # 16.....	96
Tabla 41: Caso de Uso # 17.....	97
Tabla 42: Historia de usuario 14.....	102
Tabla 43: Historia de usuario 22.....	103
Tabla 44: Historia de usuario 23.....	103
Tabla 45: Historia de usuario 24.....	104
Tabla 46: Historia de usuario 25.....	104
Tabla 47: Historia de usuario 26.....	105
Tabla 48: Historia de usuario 27.....	105
Tabla 49: Caso de uso #18	107
Tabla 50: Caso de uso # 19	108
Tabla 51: Caso de uso # 20	110
Tabla 52: Caso de uso # 21	111
Tabla 53: Caso de uso # 22	112
Tabla 54: Caso de uso # 23	113
Tabla 55: Caso de uso # 24	114
Tabla 56: Caso de uso # 25	115
Tabla 57: Caso de uso # 26	116
Tabla 58: Caso de uso # 27	118
Tabla 59: Caso de uso # 28	119
Tabla 60: Caso de uso # 29	120

LISTA DE ANEXOS

ANEXO 1.....	139
PRODUCT BACKLOG.....	139
ANEXO 2.....	141
PLAN DE PRUEBAS	141
ANEXO 3.....	144
casos de prueba sprint 1	144
ANEXO 4.....	158
REPORTE DE PRUEBAS SPRINT 1	158
ANEXO 5.....	161
casos de prueba sprint 2	161
ANEXO 6.....	166
REPORTE DE PRUEBAS SPRINT 2	166
Anexo 7	168
casos de prueba sprint 3	168
ANEXO 8.....	175
MATRIZ DE TRAZABILIDAD.....	175

1. INTRODUCCIÓN

El mundo organizacional encuentra cada día nuevos desafíos que afrontar, con competencia creciente, mejoras tecnológicas, lucha de precios y la necesidad por mejorar la atención al cliente, esta serie de cambios se realizan de manera rápida así que, se debe responder siempre en plazos cortos de tiempo.

Los sistemas de información (SI)¹ son el motor que mueve a las organizaciones en la actualidad, por esto la tarea de los SI es proveer soluciones y ayudar en las tareas de la empresa para alcanzar beneficios y objetivos estratégicos, no ser un lastre o problema en el desarrollo de las actividades.

Los SI juegan el papel estratégico deben ofrecerle ventajas competitivas y capacidad de adaptación a los cambios, esto marcara la diferencia entre una organización competitiva en un mercado feroz o una condenada a desaparecer, esta adaptación constante no solo debe buscar mejorar si no realizar todos estos procesos con el mínimo costo posible en recursos, y con una filosofía de trabajo ágil y eficaz.

El sector transporte no es ajeno a estos cambios, en un mundo de dispositivos electrónicos manejando las riendas de este tipo de negocios, deben estar las organizaciones preparadas para afrontar estos cambios, prueba de esto son aplicaciones móviles como easy taxi, uber, tappsi reconocidas a nivel mundial las cuales con millones de descargas cambiaron el rumbo del sector de transporte publico de taxis, adaptar estas tecnologías a sus necesidades y por su puesto seguir innovando, permite a las empresas mantenerse, ofreciéndoles grandes beneficios en aumento de clientes y por consecuente de rentabilidad.

¹ Pressman R. Ingeniería del software 7ma.Ed."un enfoque práctico", McGraw-Hill, 2010.

2. CAPITULO I – GENERALIDADES

2.1. DESCRIPCIÓN DEL PROBLEMA

Las empresas prestadoras del servicio público de taxis requieren particularmente tener un estricto control sobre el desempeño operativo de cada uno de sus conductores, teniendo en cuenta que este control les puede asegurar más éxito en su negocio que se traducirá en rentabilidad. Si bien se posee información básica como datos personales y laborales, también se requiere que los datos concernientes al desarrollo de su actividad laboral estén documentados para poder de esa manera hacer un seguimiento claro de cada uno, ejerciendo claro control sobre todas las partes articuladas en el negocio.

El mejoramiento continuo es clave en cualquier organización, la relación con los clientes y la calidad del servicio ofrecido determinan el crecimiento de esta, es por esto que la toma de decisiones basadas en información clara y pertinente se vuelve de vital importancia para mantener el buen nombre de la organización en el mercado, así como de mantenerla competitiva frente a otras organizaciones.

Los datos acerca de acontecimientos que repercuten directamente con el negocio, a menudo se encuentran solo en forma física y registrada en diferentes lugares, inclusive mucha de dicha información ni siquiera se encuentra reportada a las empresas, un sistema de información que contenga todos estos datos de manera conjunta sobre comparendos, accidentes, y demás estadísticas facilitaría el control sobre los conductores.

Un sistema de información centralizado con datos unificados y actualizados permite mejor toma de decisiones e incluso ayudaría en la prevención de futuros problemas que se puedan desencadenar a partir de algún incidente, esto con el fin de guardar siempre el buen nombre de la empresa para poder asegurar a los clientes que solo personal realmente capacitado para su tarea se desempeña bajo su nombre.

3. JUSTIFICACIÓN

Debido a la creciente necesidad de automatización en el mundo actual, las empresas deben garantizar cada día un mejor servicio, uno de más alta calidad, este logro no se tiene si falla en forma previa el control sobre los puntos críticos del negocio como piezas claves dentro del mismo, cuando hablamos de un servicio público se tienen muchas más variables en juego, sin duda alguna el trato al cliente es la más importante, tal es el caso de una empresa de taxis donde por un mal servicio se puede perder a un cliente potencial.

Los propietarios son directamente afectados con respecto a la calidad y la imagen que se tiene ante los clientes, es por esto que es de vital necesidad conocer a fondo como es el servicio que presta cada uno de los trabajadores que día a día laboran en la empresa, además tener conocimiento de cualquier otro variante en el desarrollo diario de sus actividades, como el caso de comparendos o algún tipo de infracción o accidente no reportado.

Controlar puede ser una tarea difícil teniendo en cuenta el ámbito del negocio del transporte, sin embargo al ofrecer herramientas tecnológicas que ayuden en la tarea de supervisar cada vehículo, conductor y operario y cada una de sus tareas asociadas, asegura a las empresas que su nombre se potencializara en el mercado con la mitigación de cualquier tipo de riesgos al tener de forma unificada y permanentemente monitoreada este tipo de información, para así tener bases más sólidas en torno a las decisiones que se tomen, sobre el futuro de la empresa y sus operarios.

El compromiso en las organizaciones no solo es con la innovación, sino también con el medio ambiente, sistemas de información automatizados permiten que toda la información se encuentre de manera organizada, sistematizada, redundante y adicional a esto se ahorran costos y perjuicios adicionales al planeta y los ecosistemas gracias a la reducción en la utilización de papel.

4. OBJETIVOS

4.1. OBJETIVO GENERAL

Desarrollar e implementar un prototipo de un sistema de información en la web para empresas prestadoras del servicio público de taxis en Pereira.

4.2. OBJETIVOS ESPECÍFICOS

- Diseñar un prototipo web de sistema de información para las empresas de taxis utilizando la metodología ágil Scrum.
- Implementar el prototipo de la aplicación web diseñada utilizando las técnicas que ofrece la ingeniería del software.
- Realizar pruebas funcionales del prototipo.
- Elaborar manuales de usuario y técnico para el prototipo.

5. ALCANCE DEL PROTOTIPO A DESARROLLAR

a través de este documento se muestra el análisis y diseño de un sistema de información para una empresa de taxis, con un nivel de detalle medio en las funcionalidades y procedimientos que deben realizarse para la operación de una empresa de este tipo, todo esto desarrollado siguiendo técnicas de ingeniería del software como son establecimiento de requerimientos, vistas 4 + 1, implementación mediante una metodología ágil de desarrollo, creación de planes de prueba y diseño de manuales técnicos y de usuario.

La implementación del prototipo abarcará las siguientes funcionalidades las cuales se describen por módulos:

- ✓ Módulo ingreso de usuarios
 - Creación de cuentas
 - Autenticación de usuarios
 - Recuperar contraseñas

- ✓ Módulo Administrador

- ✓ Módulo gestión de operarios

- ✓ Módulo operario

- ✓ Módulo gestión de vehículos

- ✓ Módulo gestión conductores

- ✓ Módulo ingresos

- ✓ Módulo servicios realizados

- ✓ Vista como usuario invitado

6. CAPÍTULO II - MARCO REFERENCIAL

6.1 MARCO TEÓRICO

Esta sección presenta cada uno de los conceptos, procedimientos y herramientas que se utilizaron en la construcción del prototipo del sistema, las cuales incluyen: lenguajes de programación, técnicas de desarrollo, metodologías desarrolladas y otros.

6.1.1 INGENIERÍA DEL SOFTWARE

Actualmente el mundo se encuentra informatizado, las industrias implementan cada día software de control en cada una de sus procesos, esto con el fin de agilizar tareas y sobre todo para el seguimiento y análisis de resultados, bajo este marco mundial es necesario que aparezca la ingeniería del software como método para producir software costeable, accequible por los millones de usuarios a nivel global de las nuevas tecnologías.

¿Qué es la ingeniería del software?

El IEEE computer society define la ingeniería del software como:

“Aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento del software, es decir, la aplicación de la ingeniería del software”.²

“Es una disciplina de la ingeniería que comprende todos los aspectos de la producción de software, desde las etapas iniciales de la especificación del sistema, hasta el mantenimiento de este después de que se utiliza.”³

La ingeniería del software comprende además de los procesos técnicos del desarrollo del software, actividades tales como gestión de proyectos de software y el desarrollo de herramientas, métodos y teorías de apoyo a la producción de software.

² IEEE Standard glossary of software engineering terminology, IEEE std 610.12-1990.1990

³ Sommerville Ian, Ingeniería del software 7ma. Ed.-, Pearson education S.A, séptima edición, p. 6

La ingeniería del software se encuentra organizada en 15 áreas del conocimiento, según un proyecto liderado por IEEE Computer Society, proyecto denominado swebok, cada una de estas áreas esta descrita por una visión general de su ámbito y como se relaciona con las demás áreas del conocimiento, constan de sub-áreas y sub-temas que ofrecen un marco más amplio del conocimiento de los tópicos enmarcados dentro de cada una para describir a gran nivel de detalle cómo funciona cada parte de la ingeniería del software. ⁴

Las 15 áreas del conocimiento son:

- Requisitos de Software
- Diseño de Software
- Construcción de Software
- Pruebas de Software
- Mantenimiento de Software
- Gestión de la configuración
- Gestión de la Ingeniería de Software
- Proceso de Ingeniería de Software
- Herramientas y métodos de la Ingeniería de Software
- Calidad del Software
- Práctica Profesional de la Ingeniería de Software
- Economía de la Ingeniería de Software
- Fundamentos de Computación
- Fundamentos Matemáticos
- Fundamentos de la Ingeniería

6.1.2 ¿QUÉ ES UN PROCESO DE SOFTWARE? ⁵

Es el conjunto de actividades y resultados asociados que producen el producto de software, principalmente se encuentran 4 actividades para describir este proceso:

1: especificaciones del software, el cliente y el ingeniero de software definen que software se va a producir, las condiciones y restricciones de su operación.

2: Desarrollo del software, se realiza el diseño y codificación del programa.

⁴ <http://www.computer.org/web/swebok>

⁵ Sommerville Ian, Ingeniería del software 7ma. Ed.-, Pearson education S.A, séptima edición pag 7

3: Validación del software, Se prueba el software para asegurarse de que el producto es lo que el cliente requiere.

4: Evolución del software, modificaciones al software para adaptarlo a los cambios requeridos por parte del mercado y los clientes.

Existen diferentes tipos de sistemas de información, dependiendo de la naturaleza y objetivo de estos se requieren igualmente distintos procesos para su desarrollo. Por ejemplo algunos sistemas deben ser totalmente especificados antes de empezar su desarrollo como lo son los de “tiempo real”, sin embargo se encuentran otros que pueden ser desarrollados de manera modular, permitiendo probar lo realizado mediante la implantación progresiva del sistema, dando pie a mejoras continuas, por lo tanto es necesario elegir un proceso de desarrollo de software adecuado a las características del producto que se va a desarrollar, puesto que elegir un proceso que no cumple con los lineamientos del producto puede retrasar su consecución y comprometer la calidad del trabajo final.

6.1.3 MODELOS DE PROCESOS DE DESARROLLO DE SOFTWARE.

6.1.3.1 EL MODELO EN CASCADA

Etapas del modelo o actividades fundamentales del desarrollo de software:

1. Análisis y definición de requerimientos: servicios, restricciones y metas del sistema, estos se definen a partir de consultas con usuarios, se definen en detalle y son las especificaciones del sistema.
2. Diseño del sistema de software: el proceso de diseño de software toma los requerimientos y los divide en funcionales y no funcionales, esta etapa establece una arquitectura completa del sistema, identifica y describe las abstracciones fundamentales del sistema y sus relaciones.
3. Implementación y prueba de unidades: el diseño del software se lleva a cabo en esta etapa como un conjunto de unidades de programas, probar estas unidades implica verificar que cada una cumpla con sus especificaciones.
4. Integración y prueba del sistema: los programas o unidades individuales se integran y prueban como un sistema completo para asegurar que se cumplan

los requerimientos de software. Después de las pruebas el sistema es entregado al cliente.

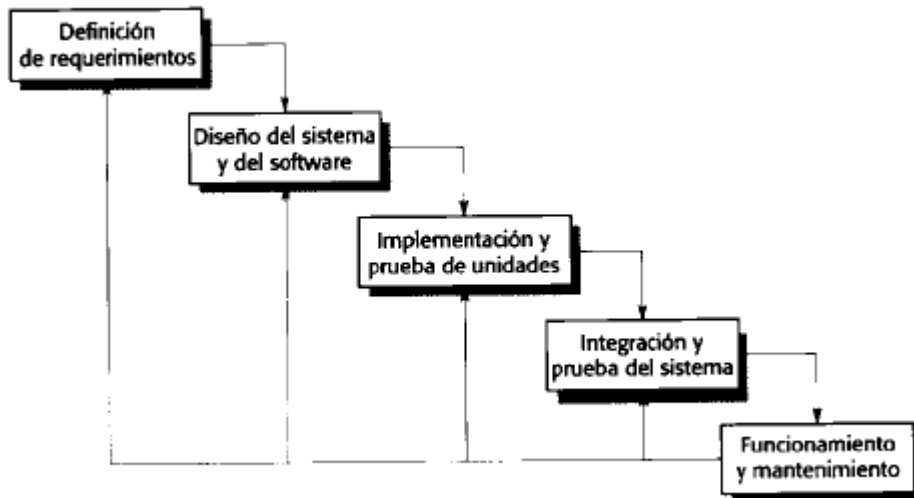


Figura 1: modelo en cascada ⁶

El modelo de la cascada es el paradigma más antiguo de la ingeniería de software propuesto por Winston Royce(1970). Sin embargo, en las últimas tres décadas, las críticas hechas al modelo han ocasionado que incluso sus defensores más obstinados cuestionen su eficacia.

Entre los problemas que en ocasiones surgen al aplicar el modelo de la cascada se encuentran los siguientes:

1. los proyectos reales casi nunca siguen el flujo secuencial propuesto por este modelo. Aunque el modelo lineal acepta repeticiones, lo hace en forma indirecta. Como resultado, los cambios generan confusión conforme el equipo del proyecto avanza.
2. A menudo, es difícil para el cliente enunciar en forma explícita todos los requerimientos. El modelo de la cascada necesita que se haga y tiene dificultades para aceptar la incertidumbre natural que existe al principio de muchos proyectos.

⁶ Fuente: Sommerville Ian, Ingeniería del software 7ma. Ed.-, Pearson education S.A, séptima edición, figura 4.1 el ciclo de vida del software, p. 62

3. El cliente debe tener paciencia. No se dispondrá de una versión funcional del sistema hasta que el proyecto esté muy avanzado. Un error grande sería desastroso si se detectara hasta revisar el programa en funcionamiento.⁷

El desarrollo de software en la actualidad es acelerado y cambiante en cuanto a funcionalidades, contenidos y nuevas tecnologías incorporadas, este modelo puede presentarse inapropiado para este contexto, no obstante sigue siendo un modelo útil para procesos que siguen un desarrollo lineal y que tienen unos requerimientos fijos.

6.1.3.2 DESARROLLO EVOLUTIVO

Se basa en la idea de desarrollar una implementación inicial, exponiéndola a la validación del usuario y refinándola a través de diferentes versiones hasta que se desarrolle el sistema completo,⁸ las actividades de especificación desarrollo y validación se unen en vez de separarse, con rápida retroalimentación entre estas.

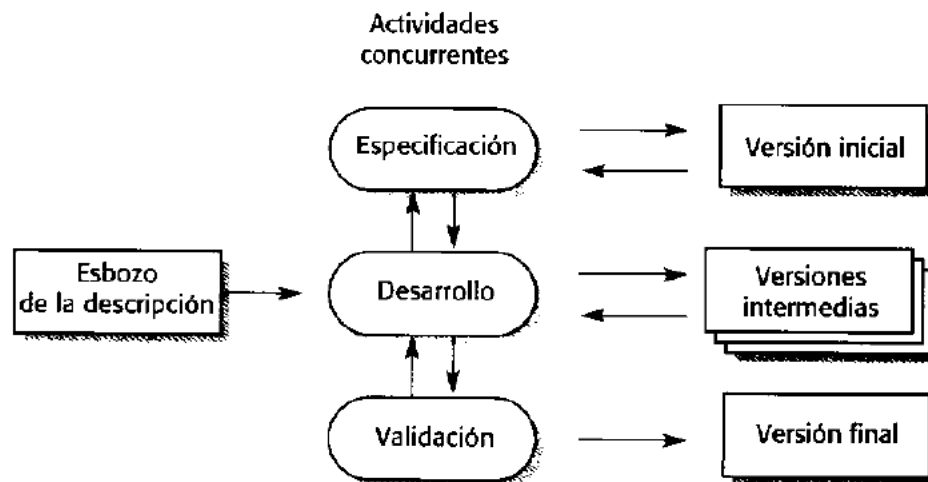


Figura 2: desarrollo evolutivo⁹

⁷ Pressman R. 2010, Ingeniería del software 7ma. Ed. "un enfoque práctico", McGraw-Hill, p. 34.

⁸ Sommerville Ian, Ingeniería del software 7ma. Ed., Pearson education S.A, séptima edición, p. 63.

⁹ Fuente: Sommerville Ian, Ingeniería del software 7ma. Ed., Pearson education S.A, séptima edición, figura 4.2 desarrollo evolutivo, pag 63

Existen dos modelos comunes para realizar procesos evolutivos:

Prototipos

Frecuentemente se definen el conjunto de objetivos que un software a desarrollar debe cumplir, sin embargo puede no encontrarse a alto nivel de detalle sus características y funcionamiento debido a esto es necesario realizar una versión parcial que pasara a ser evaluada y mejorada en otra iteración, también es prudente la realización de prototipos en casos donde no se tiene plenamente definido sobre que arquitectura operara el sistema, al igual que el tipo de sistema operativo o cómo será la interacción hombre máquina, en todos estos casos realizar la implementación mediante versiones, o prototipos asegura que el producto va desarrollándose conforme a las necesidades reales, contrario de lo que pasaría con un proceso en cascada en el cual podría realizarse todo el producto y al final descubrir que este no cumple con los objetivos planteados o requeridos por el cliente.

La realización del prototipo tiene como objetivo clarificar los requerimientos del software, es ampliamente utilizado pues a los desarrolladores les permite implementar algo de manera rápida mientras a los usuarios se les ofrece una interacción con el sistema de manera real, no obstante puede llegar a ser problemático porque:

1. Cuando los usuarios ven una versión funcional del software, ignoran que por la rápida realización de esta aún faltan muchas características propias de los requerimientos no funcionales, como la calidad general del software y facilidad de mantenimiento.
2. En el afán de realizar una versión funcional de manera rápida se puede escoger un sistema operativo inapropiado, implementar algoritmos poco eficientes y utilizar cualquier tipo de herramienta que a la larga pueda ir en detrimento del producto final.

Al realizarse un prototipo se debe recordar siempre que es solo el camino para una eficaz toma de requerimientos, cuando se acepta el prototipo se emprende la construcción del producto real enfocados en la calidad.

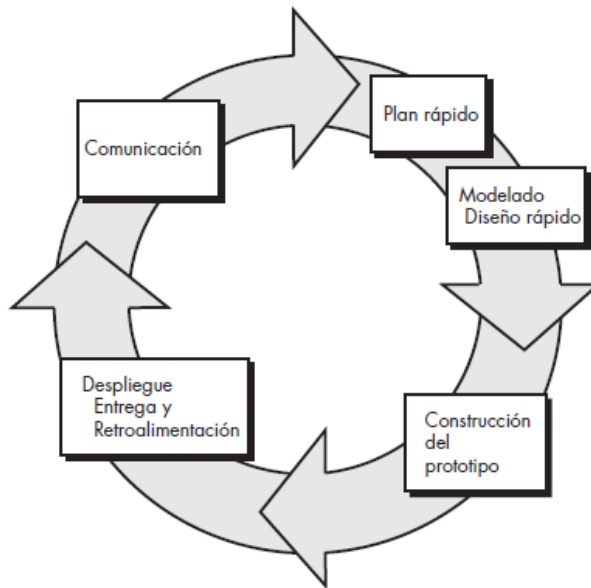


Figura 3: el paradigma de hacer prototipos¹⁰

El modelo espiral

Fue propuesto en 1986 por, Barry Boehm, quien lo describe así:

*“El modelo de desarrollo espiral es un generador de modelo de proceso impulsado por el riesgo, que se usa para guiar la ingeniería concurrente con participantes múltiples de sistemas intensivos en software. Tiene dos características distintivas principales. La primera es el enfoque cíclico para el crecimiento incremental del grado de definición de un sistema y su implementación, mientras que disminuye su grado de riesgo. La otra es un conjunto de puntos de referencia de anclaje puntual para asegurar el compromiso del participante con soluciones factibles y mutuamente satisfactorias.”*¹¹

Emplear este método significa desarrollar un proceso evolutivo de entregas, por ejemplo en las primeras entregas se puede presentar un modelo o prototipo, en iteraciones posteriores se realizan versiones más completas del software hasta tener el producto final.

¹⁰ Pressman R.2010, Ingeniería del software 7ma.Ed.”un enfoque práctico”, McGraw-Hill, figura 2,6 el paradigma de hacer prototipos p. 37

¹¹ Sommerville Ian, Ingeniería del software 7ma. Ed.-, Pearson education S.A, séptima edición, p. 39

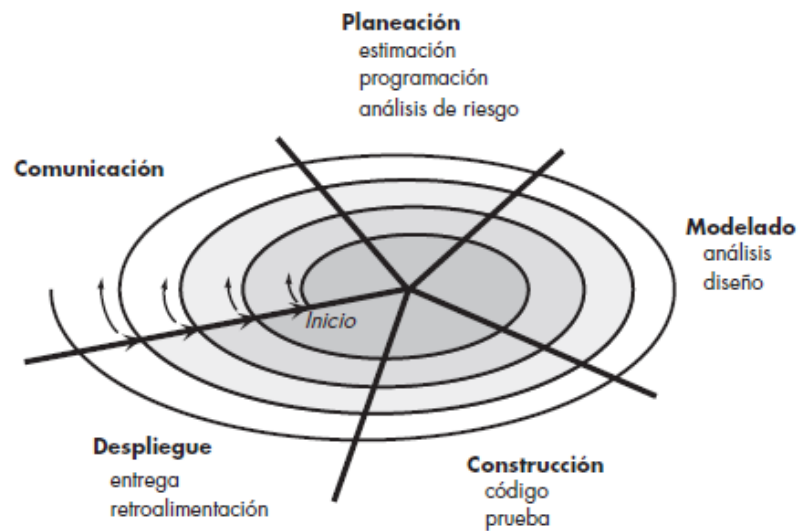


Figura 4: modelo en espiral¹²

El equipo de desarrollo del software define un conjunto de actividades de manera estructurada, dentro de las cuales también existen tareas implícitas, se empieza con el proceso, cuando se han atravesado todas las etapas se hace una entrega las cuales son incrementales en cuanto a funcionalidades y calidad, al llegar de nuevo a la planeación se hacen ajustes al plan del proyecto con base en la retroalimentación del equipo de trabajo y del nivel de satisfacción del cliente.

Este modelo tiene como gran característica la capacidad de adaptarse para ser usado en todo el ciclo de vida del software, es decir mientras algunos modelos acaban con la entrega del producto, este por su naturaleza en espiral puede desembocar en nuevos productos alejados totalmente de la idea inicial con el pasar de las iteraciones.

El modelo espiral es un enfoque realista para el desarrollo de sistemas y de software a gran escala. Como el software evoluciona a medida que el proceso avanza, el desarrollador y cliente comprenden y reaccionan mejor ante los riesgos en cada nivel de evolución. El modelo espiral usa los prototipos como mecanismo de reducción de riesgos, pero, más importante, permite aplicar el enfoque de hacer prototipos en cualquier etapa de la evolución del producto. Mantiene el enfoque de escalón sistemático sugerido por el ciclo de vida clásico, pero lo incorpora en una estructura iterativa que refleja al mundo real en una forma más

¹² Pressman R.2010, Ingeniería del software 7ma.Ed."un enfoque práctico", McGraw-Hill, figura 2,7 modelo de espiral común p. 39

realista. El modelo espiral demanda una consideración directa de los riesgos técnicos en todas las etapas del proyecto y, si se aplica de manera apropiada, debe reducir los riesgos antes de que se vuelvan un problema.¹³

En los procesos o metodologías ágiles se usan estas filosofías incrementales.

6.1.3.3 DESARROLLO BASADO EN COMPONENTES

Este modelo consiste en construir aplicaciones de software basados en “artefactos prefabricados”, es decir tomar un fragmento de código y reutilizarlo.

Las actividades de modelado y construcción comienzan con la identificación de candidatos de componentes. Éstos pueden diseñarse como módulos de software convencional o clases orientadas a objetos o paquetes de clases. Sin importar la tecnología usada para crear los componentes, el modelo de desarrollo basado en componentes incorpora las etapas siguientes (se implementan con el uso de un enfoque evolutivo):¹⁴

1. Se investigan y evalúan, para el tipo de aplicación de que se trate, productos disponibles basados en componentes.
2. Se consideran los aspectos de integración de los componentes.
3. Se diseña una arquitectura del software para que reciba los componentes.
4. Se integran los componentes en la arquitectura.
5. Se efectúan pruebas exhaustivas para asegurar la funcionalidad apropiada.

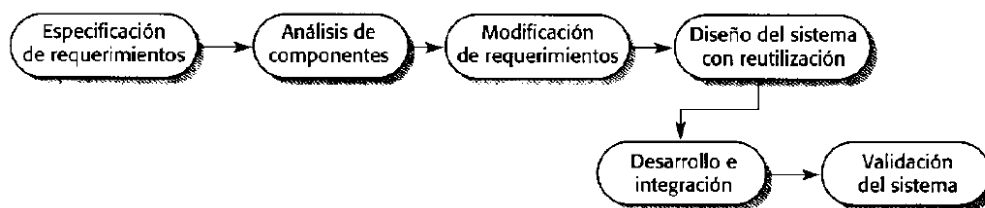


Figura 5: desarrollo basado en componentes¹⁵

¹³ Pressman R.2010, Ingeniería del software 7ma.Ed.”un enfoque práctico”, McGraw-Hill, p. 40

¹⁴ Pressman R.2010, Ingeniería del software 7ma.Ed.”un enfoque práctico”, McGraw-Hill, p. 43

¹⁵ Fuente: Sommerville Ian, Ingeniería del software 7ma. Ed.-, Pearson education S.A, séptima edición, figura 4.3 ingeniería del software basada en componentes, p. 65

Las etapas de especificación de requerimientos y validación son iguales que en otros modelos, son esas actividades intermedias las diferenciadoras en este tipo de desarrollo, durante el análisis de componentes es difícil encontrar un nivel alto de concordancia entre las funcionalidades que este componente pueda ofrecer y lo que se requiere sin embargo en este punto entra a jugar la experiencia y habilidades del equipo de trabajo en la unión de estas partes.

Cuando se han encontrado los componentes a utilizar pueden surgir modificaciones a los requerimientos para que estos reflejen lo que puede ofrecer los elementos que se utilizaran, en la etapa de reutilización simplemente se toman un marco de trabajo y con base en los componentes escogidos se construye el nuevo producto.

Este modelo tiene como principal característica y ventaja la disminución en el código a escribir, lo cual reduce costos y riesgos al trabajar con componentes probados y que funcionan, una desventaja puede ser que en el afán de adaptar los requerimientos a los componentes se pueden perder o modificar características puntuales y necesarias en el software a producir.

6.1.4 METODOLOGIAS AGILES

Las metodologías ágiles o desarrollo ágil de software, son un concepto que se refiere al conjunto de métodos y estándares utilizados para un desarrollo incremental o iterativo de un producto de software, donde los requerimientos van evolucionando a través de un conjunto de prácticas organizadas y estructuras en grupos de colaboración multidisciplinarios que realizan el producto mediante iteraciones, las cuales tienen un ciclo de vida que incluye etapas conocidas como planificación, análisis, diseño, codificación, pruebas y documentación.

El manifiesto ágil

En febrero de 2001 un grupo de 17 críticos expertos en la industria del software se reunieron para discutir sobre como adecuar las metodologías a las necesidades de la industria; tras esta reunión surgió el manifiesto ágil y la organización “the agile Alliance”. El manifiesto recoge la filosofía de las metodologías ágiles,

mientras que la segunda está dedicada a promover conceptos relacionados con el desarrollo ágil y a ayudar a las organizaciones para adoptar estas metodologías.

El manifiesto ágil define cuatro valores por los que se deberían guiar las metodologías ágiles:

Estamos buscando mejores maneras para desarrollar software y ayudar a otros a desarrollarlo.

En este trabajo valoramos:

- *Al individuo y sus interacciones más que al proceso y las herramientas.*
- *Desarrollar software que funciona, más que obtener una buena documentación.*
- *La colaboración con el cliente más que la negociación de un contrato.*
- *Responder a los cambios más que seguir una planificación.*

De esta manera, mientras mayor valor tengamos en la parte derecha más apreciaremos los de la parte izquierda."

"Manifiesto ágil"¹⁶

“Doce principios que caracterizan un proceso ágil:

- 1) La prioridad es satisfacer al cliente mediante tempranas y continuas entregas de software que le aporte un valor.
- 2) Dar la bienvenida a los cambios incluso al final del desarrollo. Los cambios le darán una ventaja competitiva a nuestro cliente.
- 3) Hacer entregas frecuentes de software que funcione, desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible entre entregas.
- 4) Las personas del negocio y los desarrolladores deben trabajar juntos diariamente a lo largo de todo el proyecto.
- 5) Construir el proyecto en torno a individuos motivados. Darles el entorno y el apoyo que necesitan y confiar en ellos.
- 6) El diálogo cara a cara es el método más eficiente y efectivo para comunicar información dentro de un equipo de desarrollo.

¹⁶ El manifiesto ágil disponible en: <http://www.agilemanifesto.org/>

- 7) El software que funciona es la principal medida del progreso.
- 8) Los procesos ágiles promueven un desarrollo sostenido. Los promotores, usuarios y desarrolladores deben poder mantener un ritmo de trabajo constante de forma indefinida.
- 9) La atención continua a la calidad técnica y al buen diseño mejoran la agilidad.
- 10) La simplicidad es esencial. Se ha de saber maximizar el trabajo que no se debe realizar.
- 11) Las mejores arquitecturas, requisitos y diseños surgen de los equipos que se han organizado ellos mismos.
- 12) En intervalos regulares, el equipo debe reflexionar con respecto a cómo llegar a ser más efectivo, y ajustar su comportamiento para conseguirlo".¹⁷

Existen numerosas metodologías de desarrollo de software que se describen como ágiles, estas utilizan diferentes métodos y principios en su camino por satisfacer las necesidades del sistema que buscan implementar.

Algunas de las más importantes y populares son:

6.1.4.1 DYNAMIC SYSTEMS DEVELOPMENT METHOD (DSDM)

Nace en 1994 y con el objetivo de crear una herramienta RAD (rapid applications development) unificada, mediante la definición de un framework de desarrollo para procesos de producción de software, toma como base el tiempo y los recursos que se tienen, para evaluar que funcionalidades se pueden construir con estos.

Esta metodología plantea como ideas fundamentales:

- Nada es construido a la perfección a la primera
- 80 por ciento de una aplicación puede entregarse en 20 por ciento del tiempo que tomaría entregarla completa (100 por ciento). *(el 80% de las funcionalidades del proyecto se realizan con el 20% del tiempo, y el 20% restante, los detalles, consumen el 80% del tiempo restante).*
- Es improbable que alguien conozca todos los requisitos del sistema desde el primer día.¹⁸

¹⁷ Gonzales F.J."Introducción a las metodologías ágiles", p. 12.

La metodología DSDM propone un modelo de proceso ágil, conocido como “*ciclo de vida DSDM*”, este presenta tres ciclos iterativos y estos a su vez son precedidos por dos actividades adicionales al ciclo de vida así:

Estudio de factibilidad: se establecen requerimientos y restricciones básicas del producto, posteriormente se evalúa si DSDM es apropiada para el proyecto.

Estudio del negocio: se estudia el negocio, sus características funcionales, se define la arquitectura básica de la aplicación y se identifican los requerimientos para dar mantenimiento.

Iteración del modelo funcional: se crean prototipos incrementales que muestran al cliente las funcionalidades, el objetivo de este ciclo iterativo es recoger requerimientos adicionales como producto de la retroalimentación que producen los usuarios al interactuar con el prototipo.

Diseño e iteración de la construcción: se revisan los prototipos construidos, con el fin de garantizar que en cada iteración se ha hecho ingeniería en forma que permita dar valor operativo del negocio a los usuarios finales, (ocurre de manera concurrente a la iteración del modelo funcional).

Implementación: se pasa de un prototipo a un sistema en operación, el trabajo de desarrollo DSDM continua y vuelve a la actividad de iteración del modelo funcional.

6.1.4.2 CRYSTAL METHODOLOGIES

Alistar Cockburn creó la familia cristal de métodos ágiles, donde el equipo de personas que se tiene para desarrollar el software es la base, cristal establece políticas de trabajo orientadas a explotar y mejorar las habilidades individuales del grupo.

*“un juego cooperativo con recursos limitados, de invención y comunicación, con el objetivo primario de entregar software útil que funcione y con la meta secundaria de plantear el siguiente juego”.*¹⁹

Alistar Cockburn

¹⁸ Gonzales F.J. “Introducción a las metodologías ágiles”, P. 14

¹⁹ Pressman R. 2010, Ingeniería del software 7ma.Ed. “un enfoque práctico”, McGraw-Hill, p. 72.

De acuerdo al número de personas se establece la metodología:

3-8 personas crystal clear

10-20 crystal yellow

25-50 crystal orange

6.1.4.3 ADAPTATIVE SOFTWARE DEVELOPMENT (ASD)

Fue propuesto por Jim Highsmith como técnica para elaborar sistemas de software complejos, se fundamenta en la colaboración humana y la organización del equipo.

²⁰Los objetivos de esta metodología son cuatro:

- 1) Concienciar a la organización de que debe esperar cambio e incertidumbre y no orden y estabilidad.
- 2) Desarrollar procesos iterativos de gestión del cambio.
- 3) Facilitar la colaboración y la interacción de las personas a nivel interpersonal, cultural y estructural.
- 4) Marcar una estrategia de desarrollo rápido de aplicaciones pero con rigor y disciplina.

El ciclo de vida definido para esta metodología consta de tres fases:

Especulación: se inicia el proyecto, se realiza la planeación del ciclo, al tener la información del proyecto como: objetivo del producto y restricciones para el desarrollo, (fechas de entrega y requerimientos básicos), se definen los ciclos de entrega y los incrementos de software que se requerirán.

Colaboración: incluye comunicación, trabajo en equipo, multiplicación del talento, el equipo de trabajo empieza a desarrollar los componentes que hacen parte de un ciclo adaptativo, recabando requerimientos, con énfasis siempre en el aprendizaje.

Aprendizaje:

- Componentes implementados o aprobados: se realizan retroalimentaciones enfocadas en revisiones técnicas formales.

²⁰ Gonzales F.J. Introducción a las metodologías ágiles, p. 16

- Análisis post mortem del proyecto

6.1.4.4 EXTREME PROGRAMMING (XP)

Creada por Kent Beck, Ward Cunningham y Ron Jeffries a finales de los noventa, es un enfoque orientado a objetos como paradigma de programación, con cuatro actividades estructurales:

Planeación: se inicia recabando requerimientos, con esto los miembros técnicos del equipo XP entienden el objetivo del producto de software, y adquieren las características principales y funcionalidades requeridas.

Se crean historias de usuarios que describen las salidas necesarias, características y funcionalidades del software, el cliente asigna un valor o prioridad a cada historia, los miembros del equipo le asignan un costo (medido en tiempo de desarrollo), los clientes y desarrolladores trabajan de manera conjunta para decidir que historias de usuario van en cada entrega.

A medida que el trabajo avanza, el cliente puede agregar historias, modificar las existentes, descomponerlas o eliminarlas.

Diseño: el diseño guía la implementación de una historia, tal y como está escrita sin quitar ni agregar nada, utiliza el concepto de tarjetas de responsabilidad (CRC), como mecanismo para pensar el software en un contexto orientado a objetos, las tarjetas identifican y organizan clases orientadas a objetos que son relevantes para el incremento actual de software.

Estas tarjetas son el único producto del trabajo de diseño que se genera en el proceso XP.

Codificación: inicialmente el equipo realiza pruebas unitarias, a cada una de las historias que se incluirán en la entrega que este en operación, se crea la prueba unitaria, empiezan las tareas de integración y compatibilidad de interfaces.

Pruebas: las pruebas unitarias creadas en la codificación, deben implementarse con una estructura que las automatice para que puedan ejecutarse en repetidas veces y con facilidad, esto estimula pruebas de regresión siempre que se modifique el código.

Las pruebas de aceptación son especificadas por el cliente, se refieren y concentran en las características y funcionalidades a nivel general del software visibles para el cliente, las pruebas de aceptación se derivan de las historias que se han implementado como parte de la liberación del software.

6.1.4.5 SCRUM

Scrum es un marco de trabajo de procesos que ha sido usado para gestionar el desarrollo de productos complejos desde principios de los años 90. Scrum no es un proceso o una técnica para construir productos; en lugar de eso, es un marco de trabajo dentro del cual se pueden emplear varias técnicas y procesos. Scrum muestra la eficacia relativa de las prácticas de gestión de producto y las prácticas de desarrollo, de modo que podamos mejorar.²¹

Scrum es una metodología congruente con el manifiesto ágil, sus principios se utilizan para guiar actividades de desarrollo dentro de un proceso de análisis que incorpora en las actividades estructurales de: requerimientos, análisis, diseño, evolución y entrega, dentro de cada actividad estructural, las tareas del trabajo ocurren con un patrón de proceso llamado sprint.

¿Cuándo se utiliza?

En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales.

Scrum también se utiliza para resolver situaciones en que no se está entregando al cliente lo que necesita, cuando las entregas se alargan demasiado, los costes se disparan o la calidad no es aceptable, cuando se necesita capacidad de reacción ante la competencia, cuando la moral de los equipos es baja y la rotación alta, cuando es necesario identificar y solucionar ineficiencias sistemáticamente o cuando se quiere trabajar utilizando un proceso especializado en el desarrollo de producto.²²

El núcleo central del desarrollo de la metodología scrum es el sprint, lo cual es una iteración las cuales definen su duración al inicio del proyecto, idealmente los sprint tienen una duración de 2 a 4 semanas.

Scrum prescribe cuatro eventos formales, contenidos dentro del Sprint:

²¹ Schwaber K. Sutherland J, "La guía de scrum" disponible en: <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-ES.pdf> p. 4

²² Que es scrum disponible en: <http://www.proyectosagiles.org/que-es-scrum>

Eventos de scrum²³:

- ✓ Reunión de Planificación del Sprint (Sprint Planning Meeting)
- ✓ Scrum Diario (Daily Scrum)
- ✓ Revisión del Sprint (Sprint Review)
- ✓ Retrospectiva del Sprint (Sprint Retrospective)

El equipo scrum:

Product owner (dueño del producto)

Scrum master

- Equipo de desarrollo.

Durante el sprint no se deben realizar cambios que afecten el objetivo del sprint, el alcance puede ser negociado con el product owner, un sprint se puede considerar como un proyecto, los proyectos tienen como fin lograr algo, cada sprint tiene la definición de lo que requiere construir lo cual le brinda el horizonte como proyecto.

- Artefactos de scrum:

Lista del producto (product backlog): La Lista de Producto es una lista ordenada de todo lo que podría ser necesario en el producto, y es la única fuente de requisitos para cualquier cambio a realizarse en el producto.

La Lista de Producto enumera todas las características, funcionalidades, requisitos, mejoras y correcciones que constituyen cambios a ser hechos sobre el producto para entregas futuras. Los elementos de la Lista de Producto tienen como atributos la descripción, la ordenación, la estimación y el valor.²⁴

Seguimiento del Progreso Hacia un Objetivo

En cualquier momento, es posible sumar el trabajo total restante para alcanzar el objetivo. El Dueño de Producto hace seguimiento de este trabajo restante total al menos en cada Revisión de Sprint. El Dueño de Producto compara esta cantidad con el trabajo restante en Revisiones de Sprint previas, para evaluar el progreso

²³ Schwaber K. Sutherland J, "La guía de scrum" disponible en:
<http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-ES.pdf> p. 5

²⁴ Schwaber K. Sutherland J, "La guía de scrum" disponible en:
<http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-ES.pdf> p. 15

hacia la finalización del trabajo proyectado en el tiempo deseado para el objetivo. Esta información se muestra de forma transparente a todos los interesados.

Varias prácticas de proyección sobre tendencias se han utilizado para predecir el progreso, como trabajo consumido (burndown), avanzado (burnup) y flujo acumulado (cumulative flow).²⁵

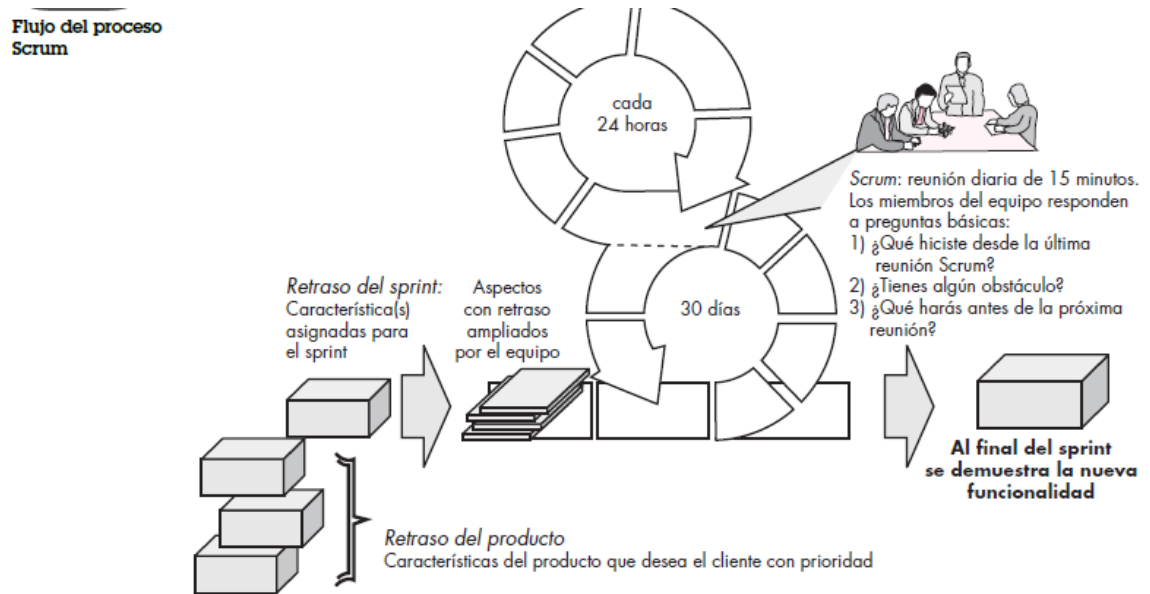


Figura 6: flujo del proceso scrum²⁶

²⁵ Schwaber K. Sutherland J, "La guía de scrum" disponible en: <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-ES.pdf> p. 16

²⁶ Pressman R, 2010. Ingeniería del software 7ma.Ed."un enfoque práctico", McGraw-Hill, figura 2,7 flujo del proceso scrum, p. 70

6.1.5 LEVANTAMIENTO Y ANALISIS DE REQUERIMIENTOS

Los requerimientos para un sistema son la descripción de los servicios proporcionados por el sistema y sus restricciones operativas, estos requerimientos reflejan las necesidades de los clientes de un sistema que ayude a resolver algún problema, el proceso de descubrir, analizar, documentar y verificar estos servicios y restricciones se denomina ingeniería de requerimientos RE.²⁷

Requerimientos del usuario: declaraciones en lenguaje normal y diagramas, de servicios que se espera que el sistema brinde, al igual que las restricciones que se deben tener.

Requerimientos del sistema: se detallan a un alto nivel las funciones y restricciones operativas. (A menudo conocidos como requerimientos funcionales).

Requerimientos funcionales: son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que este debe reaccionar a entradas particulares. En algunos casos, los requerimientos funcionales de los sistemas también pueden declarar explícitamente lo que el sistema debe hacer.

Requerimientos no funcionales: son restricciones de los servicios o funciones ofrecidos por el sistema, incluyen restricciones de tiempo, sobre el proceso de desarrollo y estándares.²⁸

El proceso de obtención y análisis de requerimientos puede ser clasificado en las siguientes fases:

Descubrimiento: proceso de interactuar con interesados del sistema y recopilar sus requerimientos.

Clasificación: organización en grupos coherentes, por prioridades, funcionales y no funcionales y otros.

Documentación: se debe realizar una documentación clara de los requerimientos recabados, ya que esto es la base para empezar el proceso de diseño del sistema que se quiera realizar.

²⁷ Sommerville Ian, Ingeniería del software 7ma. Ed.-, Pearson education S.A, séptima edición, p. 108.

²⁸ Sommerville Ian, Ingeniería del software 7ma. Ed.-, Pearson education S.A, séptima edición, p. 110

6.1.5.1 HISTORIA DE USUARIO

Es una técnica utilizada en el desarrollo de software, consiste en la redacción de un requerimiento en unas pocas frases, adicional a esto se utiliza el lenguaje común del usuario.

Las historias de usuario son ideales para desarrollos con metodologías ágiles, definen las especificaciones del proyecto, posteriormente y debido a la prioridad asignada a cada una, se acomodan dentro de las diferentes iteraciones y se realizan los estimados en tiempos para la realización de cada historia, este proceso lo realizan los desarrolladores.

Características de una historia de usuario²⁹

Independiente: pueden existir historias con tengan dependencias con otras, es necesario separar de manera que resulten lo más independiente posible, ya que esta independencia es clave a la hora de la etapa del diseño.

Negociable: las pruebas de validación son la manera en que los clientes pueden verificar el alcance de la historia de usuario.

Valoradas por clientes o usuarios: las historias de usuarios son importantes para clientes y usuarios, deben reflejar lo que ellos esperan, no lo que el desarrollador quiera hacer.

Estimables: cuando se estiman los tiempos para cada historia de usuario es más fácil definir cuanto tiempo se tardara un proyecto.

Pequeñas: deben ser pequeñas por la naturaleza iterativa de los proyectos donde se utilizan.

Verificables: debe ser posible la verificación de los logros obtenidos en las historias de usuario en cada entregable de los proyectos.

Las historias de usuario responden a tres preguntas:

¿Quién se beneficia? ¿Qué se quiere? ¿Cuál es el beneficio?

Como (rol) quiero (algo) para poder (beneficio).

²⁹ Mike Cohn, 2004 "User Stories Applied", , Addison Wesley

6.1.6 DISEÑO DE SOFTWARE

Una vez se tienen los requisitos del sistema que se construirá los ingenieros de software pasan a la fase de diseño, en esta y gracias a técnicas como el lenguaje unificado de modelado UML, se realiza el proceso de crear diagramas que especificaran desde como interactuara el sistema con los usuarios, hasta los componentes físicos que tendrá el mismo.

Esta actividad juega un rol vital en el desarrollo de software, debido a que es realmente en la etapa de diseño que un ingeniero de software descubre si es posible realizar un requerimiento tal y como lo especifico el cliente.

6.1.6.1 LENGUAJE UNIFICADO DE MODELADO (UML)

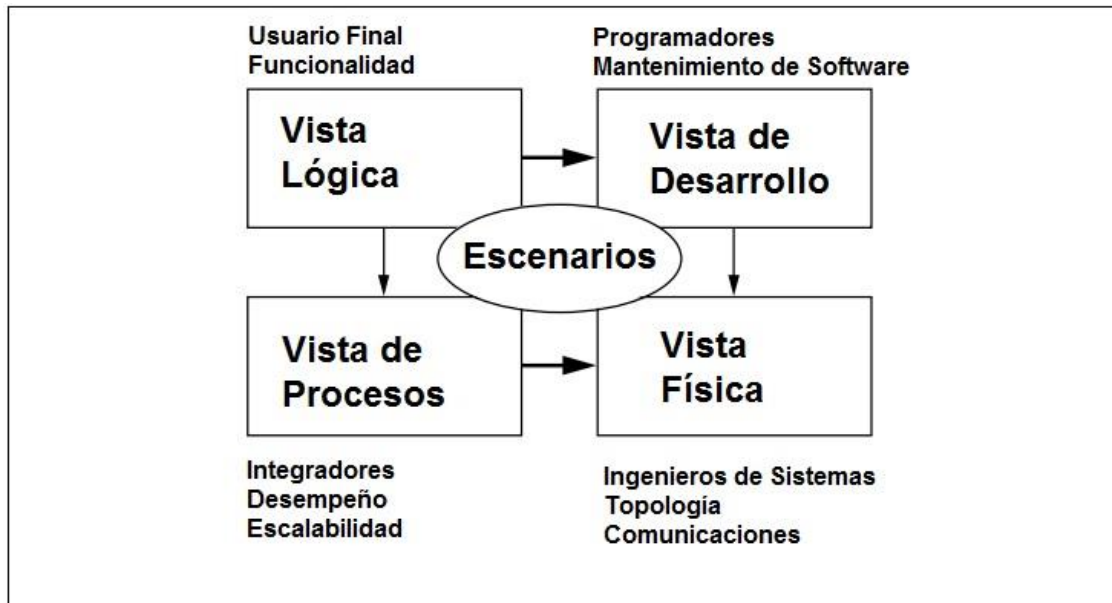
UML es un conjunto de notaciones graficas que ayudan en la descripción de sistemas de software³⁰. El UML nació de la unificación de varios gráficos de modelado orientados a objetos

UML ofrece 9 tipos de diagramas:

- diagramas de casos de uso: modelar procesos.
- diagramas de secuencia: modelar el paso de mensajes entre objetos.
- diagramas de colaboración: modelar interacciones entre objetos.
- diagrama de estados: modelar comportamiento de los objetos en el sistema.
- diagramas de actividades: modelar el comportamiento de los casos de uso, objetos u operaciones.
- diagrama de clases: modelar estructura estática de las clases del sistema.
- diagrama de objetos: modelar la estructura estática de los objetos del sistema.
- diagrama de componentes: modelar componentes.
- diagrama de implementación: modelar la distribución del sistema.

³⁰ Fowler, M. (2004) *UML distilled: a brief guide to the standard object modeling language*. Addison-Wesley Professional.

Vistas 4 + 1



La arquitectura de software trata con la abstracción, con descomposición y composición, con estilo y estética, para describir una arquitectura de software se utiliza un modelo compuesto por múltiples vistas o perspectivas. En orden de abordar eventualmente arquitecturas grandes y desafiantes, el modelo propuesto se compone de 5 vistas principales, vistas en la figura anterior³¹

La vista Lógica: es el modelo del objeto del diseño.

La vista de procesos: captura los aspectos de concurrencia y sincronización del diseño.

La vista física: describe el mapeo del software en el hardware, reflejando su aspecto distribuido.

La vista de desarrollo: describe la organización estática del software en su entorno de desarrollo.

Escenarios: este muestra los elementos de las 4 vistas trabajando juntos sin problemas con el uso de un pequeño conjunto de importantes escenarios - instancias o generalmente casos de uso.

³¹ Krutchen Phillipe, Rational Software Corp (1995). Architectural Blueprints- The "4+1" View Model of Software Architecture

Las vistas 4 + 1 ofrecen una gran ayuda en el desarrollo de un producto de software pues nos llevan paso a paso en la construcción de todos los componentes que tendrá el sistema que se desea crear, empezando con los casos de uso que nos muestran como interactuaran los usuarios del sistema con el mismo, en diferentes situaciones, todo esto dependiente del tipo de usuario y privilegios dentro del sistema con que este cuente.

Con la descripción de la secuencia de actividades que se realiza en cada caso de uso se ve la relación entre el usuario y las respuestas que dará el sistema de manera más detallada, es decir este tipo de diagramas nos permiten detallar como debe responder nuestro sistema a cada petición de un usuario.

Después de tener la interacción de los usuario con el sistema este modelo de vistas 4 + 1 nos da la posibilidad de detallar los componentes que tendrá el sistema físicos y lógicos: como estará este sistema descrito a nivel de hardware y software con componentes como framework, lenguajes de programación, y por otro lado interfaces de usuario, bases de datos, servidores y otros.

6.1.6.1.1 VISTA DE ESCENARIOS (CASOS DE USO)

Una vez recopilados los requisitos, se pueden crear un conjunto de escenarios que identifiquen una línea de utilización para el sistema que va a ser construido. Los escenarios, algunas veces llamados casos de uso, facilitan una descripción de cómo el sistema se usará.

Para crear un caso de uso, el analista debe primero identificar los diferentes tipos de personas (o dispositivos) que utiliza el sistema o producto. Estos actores actualmente representan papeles que los usuarios juegan como impulsores del sistema.

Una vez que se han identificado los actores, se pueden desarrollar los casos de uso. El caso de uso describe la manera en que los actores interactúan con el sistema. Jacobson³² sugiere un número de preguntas que deberán responderse por el caso de uso:

¿Cuáles son las principales tareas o funciones que serán realizadas por el actor?

¿Cuál es el sistema de información que el actor adquiere, produce o cambia?

¿Qué actor informará al sistema de los cambios en el entorno externo?

³² Jacobson, I. 1992, Object-Oriented Software Engineering, Addison-Wesley.

- ¿Qué información necesita el actor sobre el sistema?

Cuando se obtiene solución a cada una de las preguntas se podrán realizar unos casos de uso más acertados.

6.1.6.1.2 VISTA LOGICA (DIAGRAMA DE CLASES)

Los diagramas de clases muestran las diferentes clases que componen un sistema y cómo se relacionan unas con otras. Se dice que los diagramas de clases son diagramas «estáticos» porque muestran las clases, junto con sus métodos y atributos, así como las relaciones estáticas entre ellas: qué clases «conocen» a qué otras clases o qué clases «son parte» de otras clases, pero no muestran los métodos mediante los que se invocan entre ellas.

Clase: Una clase define los atributos y los métodos de una serie de objetos. Todos los objetos de esta clase (instancias de esa clase) tienen el mismo comportamiento y el mismo conjunto de atributos (cada objetos tiene el suyo propio).

Atributos: En UML, los atributos se muestran al menos con su nombre, y también pueden mostrar su tipo, valor inicial y otras propiedades.³³

6.1.6.1.3 VISTA DE DESARROLLO (DIAGRAMA DE COMPONENTES)

El diseño de componentes para el software describe por completo los detalles internos de cada componente.

Para lograrlo, este diseño define estructuras de datos para todos los objetos de datos locales y detalles algorítmicos para todo el procesamiento que tiene lugar dentro de un componente, así como la interfaz que permite el acceso a todas las operaciones de los componentes.³⁴

³³ Elementos de UML disponible en: <https://docs.kde.org/stable4/es/kdesdk/umbrello/uml-elements.html>

³⁴ Pressman R, 2010. Ingeniería del software 7ma.Ed."un enfoque práctico",McGrw-Hill, P. 201

6.1.6.1.4 VISTA DE PROCESOS (DIAGRAMA DE SECUENCIA)

Indica la forma en la que los eventos provocan transiciones de un objeto a otro. Una vez identificados los objetos por medio del análisis del caso de uso, el modelador crea un diagrama de secuencia: representación del modo en el que los eventos causan el flujo de uno a otro como función del tiempo. En esencia, el diagrama de secuencia es una versión taquigráfica del caso de uso.³⁵

Un diagrama de secuencia se diferencia de un diagrama de estados por que el primero representa el comportamiento del sistema describiendo la forma en la que las clases van pasando de un estado a otro, mientras que el segundo no se fija en las clases que están involucradas.

6.1.6.1.5 VISTA FISICA (DIAGRAMA DE DESPLIEGUE)

Los diagramas de implementación muestran las instancias existentes al ejecutarse así como sus relaciones. También se representan los nodos que identifican recursos físicos, típicamente un ordenador así como interfaces y objetos (instancias de las clases).³⁶

6.1.6.2 MODELO RELACIONAL

El modelo relacional es el fundamento teórico de las bases de datos denominadas SQL porque en su mayoría utilizan este lenguaje para la descripción de las bases de datos, este modelo utiliza relaciones pensadas como tablas conformadas por tuplas de datos o filas de esta tabla mientras los atributos son cada una de las columnas que representan la información que estas contienen.

Este modelo se ocupa de tres aspectos principales de la información a nivel lógico que teóricos como Date³⁷ denominan: estructura de datos, manipulación de datos e integridad de los datos.

ASPECTO ESTRUCTURAL: El usuario percibe la información de la base de datos como tablas y nada más que tablas.

³⁵ Pressman R, 2010. Ingeniería del software 7ma.Ed."un enfoque práctico",McGrw-Hill, P. 168

³⁶ Elementos de UML disponible en: <https://docs.kde.org/stable4/es/kdesdk/umbrello/uml-elements.html>

³⁷ Date C. J, 2001, Introducción a los sistemas de bases de datos 7ma. Ed., Pearson education P. 58

ASPECTO DE INTEGRIDAD: Estas tablas satisfacen ciertas restricciones de integridad.

ASPECTO DE MANIPULACIÓN: Los operadores disponibles para que el usuario manipule estas tablas, ejemplos: restringir, proyectas y combinar.

DEFINICION FORMAL DE MODELO DE DATOS

“Un conjunto de conceptos, reglas y convenciones que nos permiten describir y, en ocasiones, manipular los datos de un cierto mundo real que deseamos almacenar en la base de datos.”

Componente estático: Contempla aquellas propiedades del universo del discurso que son invariantes en el tiempo, es decir, su estructura.

Componente dinámico: Relativa a propiedades del universo del discurso que varían con el tiempo. Incluye las operaciones que se aplican a los datos o valores almacenados en la estructura.³⁸

6.1.6.3 ARQUITECTURA DE SOFTWARE

La Arquitectura de Software es, a grandes rasgos, una vista del sistema que incluye los componentes principales del mismo, la conducta de esos componentes según se la percibe desde el resto del sistema y las formas en que los componentes interactúan y se coordinan para alcanzar la misión del sistema. La vista arquitectónica es una vista abstracta, aportando el más alto nivel de comprensión y la supresión o diferimiento del detalle inherente a la mayor parte de las abstracciones.³⁹

6.1.6.3.1 MODELO VISTA CONTROLADOR

El patrón Modelo-Vista-Controlador (MVC) surge con el objetivo de reducir el esfuerzo de programación, necesario en la implementación de sistemas múltiples

³⁸ Ocampo Sepulveda C.A. Modelo relacional “Definición estática”, Universidad Tecnológica de Pereira

³⁹ Paul Clements. (1996) “A Survey of Architecture Description Languages”. Proceedings of the International Workshop on Software Specification and Design, Alemania

y sincronizados de los mismos datos, a partir de estandarizar el diseño de las aplicaciones. El patrón MVC es un paradigma que divide las partes que conforman una aplicación en el Modelo, las Vistas y los Controladores, permitiendo la implementación por separado de cada elemento, garantizando así la actualización y mantenimiento del software de forma sencilla y en un reducido espacio de tiempo. A partir del uso de frameworks basados en el patrón MVC se puede lograr una mejor organización del trabajo y mayor especialización de los desarrolladores y diseñadores.⁴⁰

Utilizar el patrón modelo vista-controlador para los desarrollos de productos de software asegura una integridad de los datos, debido a que con este modelo se separan las vistas de los usuarios de los controladores de las mismas, es decir primero se asegura que cada usuario solo tendrá acceso dentro del sistema a las partes a las que efectivamente pueda acceder con este modelo se puede controlar esto, además cada vista tiene su controlador y sus propias acciones sobre el sistema.

Los patrones a la hora de desarrollar software son una tendencia que se debe seguir ya que ayudan en la integridad del software, además contribuyen a que la funcionalidad requerida del sistema se cumple a cabalidad.

Un factor muy importante en cualquier desarrollo de un sistema de software, es la seguridad de los datos y quien puede acceder a ellos el modelo vista controlador ayuda en esta tarea de salvaguardar la información pues a través de su modelo de vista y controlador los usuarios no tienen acceso a toda la información que pueda tener dicha base de datos y esto hace el sistema menos vulnerable a ataques.

6.1.7 CODIFICACIÓN

La construcción del software en el paso a seguir, una vez se tiene modelado el sistema a construir y definido el tipo de arquitectura, se continúa con la actividad de plasmar todo el análisis y diseño en código, tarea también conocida como implementación.

PRINCIPIOS DE CODIFICACIÓN:

“Principios de preparación:

⁴⁰ González, Y. D., & Romero, Y. F(2012). Patrón Modelo-Vista-Controlador.Revista Telem@tica, 11(1), 47-57.

- *entender el problema*
- *comprender los principios y conceptos básicos de diseño*
- *elegir un lenguaje de programación que satisfaga las necesidades del software que se va a elaborar y el ambiente en que opera*
- *seleccionar un ambiente de programación que disponga de herramientas que hagan más fácil su trabajo*

Principios de codificación:

- *seleccionar estructuras de datos que satisfagan las necesidades del diseño*
- *entender la arquitectura del software y crear interfaces congruentes con ella*
- *mantener la lógica condicional tan sencilla como sea posible*
- *crear nombres significativos para las variables y seguir otros estándares de codificación*
- *escribir código que se documente a sí mismo*

Principios de validación

- *realizar el recorrido del código cuando sea apropiado*
- *llevar a cabo pruebas unitarias y corregir los errores que se detecten*
- *rediseñar el código*⁴¹

6.1.8 PRUEBAS

La tarea de testing es crucial debido a que mide la efectividad de los desarrollos realizados, ayudando a detectar defectos y fallas.

Defectos: errores internos, los encuentran los desarrolladores

Fallas: vista externa y es detectado por usuarios

Probar un software consiste en realizar un conjunto de test sobre los requerimientos funcionales y no funcionales que debe cumplir de acuerdo con lo descrito en el nivel de análisis, de esta manera verificar que fallos y defectos se encuentran en este para realizar posibles mejoras antes de la entrega final del producto.

Características de testing

⁴¹ Pressman R, 2010. Ingeniería del software 7ma.Ed."un enfoque práctico",McGrw-Hill, P. 94-95

- Se debe tener una definición clara de los resultados esperados para la prueba
- Los códigos deben ser idealmente probados por personas ajenas a su desarrollo
- Se deben probar entradas válidas y no válidas para el software

NIVELES DE PRUEBAS

TEST	OBJETIVO	PARTICIPANTES	AMBIENTE	MÉTODO
UNITARIO	Detectar errores en los datos, lógica algoritmos	Programadores	Desarrollo	Caja blanca
INTEGRACIÓN	Detectar errores de interfaces y relaciones entre componentes	Programadores	Desarrollo	Caja blanca, top down, bottom up
FUNCIONAL	Detectar errores en la implementación de requerimientos	Tester, Analistas	Desarrollo	Funcional
SISTEMA	Detectar fallas en el cubrimiento de los requerimientos	Tester, Analistas	Desarrollo	Funcional
ACEPTACIÓN	Detectar fallas en la implementación del sistema	Tester, Analistas, Clientes	productivo	Funcional

Tabla 1: niveles de pruebas⁴²

⁴² It.mentor-Capacitación y guía para el desarrollo de software, “pruebas de software”, disponible en: <http://materias.fi.uba.ar/7548/PruebasSoftware.pdf>

6.1.9 ENTREGA

La construcción de un producto de software pasa por las fases de análisis, diseño, codificación y pruebas, incluso para los desarrollos evolutivos se realizan todos estos pasos antes de la entrega.

La entrega sea para procesos de desarrollo completos o iterativos incluye el código o ejecutables (dependiendo de lo acordado con los clientes), además de la documentación generada para el proyecto y producto de software.

6.1.10 ESTANDARES WEB

Un estándar es un conjunto de reglas normalizadas que describen los requisitos que deben ser cumplidos por un producto, proceso o servicio, con el objetivo de establecer un mecanismo base para permitir que distintos elementos hardware o software que lo utilicen, sean compatibles entre sí. Por lo tanto, si se hace una analogía, los estándares web son un conjunto de reglas normalizadas con el propósito de estandarizar el desarrollo web y posterior ofrecimiento de productos, procesos o servicios web con el fin de lograr que este sea fácil de usar y que todo el mundo pueda confiar. Con estas tecnologías abiertas y de uso libre se pretende evitar la fragmentación de la Web y mejorar las infraestructuras para que se pueda evolucionar hacia una Web con la información mejor organizada⁴³

6.1.10.1 HTML (HYPER TEXT MARKUP LENGUAJE)

Lenguaje compuesto de una serie de etiquetas o marcas que permiten definir el contenido y apariencia de las páginas web. Aunque se basa en el estándar SGML, no se puede considerar que sea un subconjunto de él, existen cientos de etiquetas con diferentes atributos. W3C⁴⁴ se encarga de su estandarización.⁴⁵

⁴³ Guía breve sobre estándares web- disponible en
<<http://www.w3c.es/Divulgacion/GuiasBreves/Estandares>>

⁴⁴ W3C HTML disponible en : <http://www.w3.org/html/>

⁴⁵ MORA, Sergio Lujan,2002, Programación de aplicaciones web: historia, principios básicos y clientes web, p. 23.

6.1.10.2 XML (EXTENSIBLE MARKUP LANGUAGE)

Metalinguaje de etiquetado basado en SGML .diseñado específicamente para la WWW por W3C⁴⁶. Permite que un usuario diseñe sus propias etiquetas, con sus atributos y las reglas de constricción de documentos (sintaxis).⁴⁷

6.1.10.3 JAVASCRIPT

Es el lenguaje interpretado orientado a objetos desarrollado por Netscape que se utiliza en millones de páginas web y aplicaciones de servidor en todo el mundo.⁴⁸

Este lenguaje permite crear aplicaciones que se ejecutan a través de internet. Basada en el paradigma cliente servidor. La parte de cliente se ejecuta en el navegador, mientras la parte servidor se ejecuta en un servidor de las compañías.⁴⁹

6.1.10.4 CSS

Hojas de Estilo en Cascada (Cascading Style Sheets), es un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos.

CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Los Estilos definen la forma de mostrar los elementos HTML y XML. CSS permite a los desarrolladores Web controlar el estilo y el formato de múltiples páginas Web al mismo tiempo. Cualquier cambio en el estilo marcado

⁴⁶ W3C XML disponible en: <http://www.w3.org/XML/>

⁴⁷ MORA, Sergio Lujan,2002, Programación de aplicaciones web: historia, principios básicos y clientes web, p. 28.

⁴⁸ Que es JavaScript disponible en:

https://developer.mozilla.org/es/docs/Web/JavaScript/Acerca_de_JavaScript

⁴⁹ Qué es Javascript? Disponible en: <http://www.desarrolloweb.com/javascript/#quees>

para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento.⁵⁰

6.1.10.5 PHP (PHP: Hypertext Preprocessor)

Es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.

Lo que distingue a PHP de algo del lado del cliente como Javascript es que el código es ejecutado en el servidor, generando HTML y enviándolo al cliente. El cliente recibirá el resultado de ejecutar el script, aunque no se sabrá el código subyacente que era. El servidor web puede ser configurado incluso para que procese todos los ficheros HTML con PHP, por lo que no hay manera de que los usuarios puedan saber qué se tiene debajo.⁵¹

6.1.10.6 FRAMEWORK

Una estructura de software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta.

Características de un framework:

- Abstracción de URLs y sesiones: No se necesita manipular las URL y sesiones directamente.
- Acceso a datos: Incluyen las herramientas e interfaces necesarias para integrarse con herramientas de acceso a datos, en BBDD, XML, etc.
- Controladores: Utilizar una serie de controladores para gestionar eventos como una introducción de datos mediante un formulario o el acceso a una página. Estos controladores suelen ser fácilmente adaptables a las necesidades de un proyecto concreto.
- Autenticación y control de acceso: Incluyen mecanismos para la identificación de usuarios mediante login y password y permiten

⁵⁰ Guías breves de CSS disponible en: <http://www.w3c.es/Divulgacion/GuiasBreves/HojasEstilo>

⁵¹ Que es php disponible en: <http://php.net/manual/es/intro-what-is.php>

restringir el acceso a determinadas páginas a determinados usuarios.

- Internacionalización: Utilizar estándares internacionales para sus procesos y funciones.
- Separación entre diseño y contenido⁵².

6.1.10.7 CODEIGNITER

Codeigniter es un Framework para el Desarrollo de Aplicaciones, una herramienta para la gente que crea webs usando PHP. Su meta es permitir desarrollar proyectos mucho más rápido que si se hiciera escribiendo el código desde cero, proporcionando una gran variedad de librerías para las tareas más corrientes, así como una interfaz simple y una estructura lógica para acceder. Codeigniter permite minimizar la cantidad de código necesaria para una tarea determinada.⁵³

Una de las grandes ventajas de este framework es su facilidad de configuración.

6.1.10.8 MY SQL

MySQL es un sistema gestor de bases de datos (SGBD, DBMS por sus siglas en inglés) muy conocido y ampliamente usado por su simplicidad y notable rendimiento.

Aunque carece de algunas características avanzadas disponibles en otros SGBD del mercado, es una opción atractiva tanto para aplicaciones comerciales, como de entretenimiento precisamente por su facilidad de uso y tiempo reducido de puesta en marcha. Esto y su libre distribución en Internet bajo licencia GPL le otorgan como beneficios adicionales (no menos importantes) contar con un alto grado de estabilidad y un rápido desarrollo.⁵⁴

⁵² Qué es un framework web? Disponible en:

http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf

⁵³ Guía del usuario de Codeigniter versión 2.1.4 disponible en: http://escodeigniter.com/guia_usuario/

⁵⁴ Casillas Santillán L, Gibert Ginestà M, Pérez Mora O, bases de datos en mysql, disponible en:

http://ocw.uoc.edu/computer-science-technology-and-multimedia/bases-de-datos/bases-de-datos/P06_M2109_02151.pdf

7 CAPITULO III-OTRAS CONSIDERACIONES

El mercado de las tecnologías presenta numerosas herramientas a los usuarios para gestionar sistemas de información, todas ellas ofrecen el uso de diferentes tecnologías variando en costos y funcionalidades del sistema además de diferentes plataformas para su soporte.

Algunas de las plataformas que se encuentran disponibles en el mercado para el sector de transporte público son:

TAXIWEB GPS (desarrollado por ELEINCO S.A.S) ⁵⁵	Administración de vehículos (producido por TWS systems) ⁵⁶
Objetivo: Solución para gestión satelital	Objetivo: servicio de software de software comercial de gestión
Ubicación: Envigado, Antioquia	
Módulos que contiene: <ol style="list-style-type: none"> 1. Software para la Base: en la empresa de taxis. Es el encargado de la recepción de llamadas y de encontrar un taxi para satisfacer el requerimiento del usuario. 2. Hardware y Software para el taxi (MQ MOBILE): Es el encargado de la logística de asignación de servicios. Funciona para Tablets Android y para las unidades MQ2 vía radio y vía Celular. 3. Software para los Empresarios: Es el encargado de mostrarle al 	Módulos que contiene: <ol style="list-style-type: none"> 1.Vehiculos 2.Propietarios 3.Conductores 4. Recordar

⁵⁵ Taxiweb gps disponible en: <http://www.eleinco.com.co/pages/empresa-taxi.aspx>

⁵⁶ Administracióntaxis disponible en: <http://www.administraciotaxis.com/index.htm>

<p>dueño del taxi, todas las actividades de su vehículo. Se accede vía Internet.</p> <p>4. Software para Usuarios Finales del sistema: Se coloca en teléfonos celulares con plataforma Android. Permite solicitar servicios desde cualquier lugar.</p>	
--	--

Tabla 2: Información general taxis web gps, administración de vehículos

ADMINTAXI ⁵⁷	G-TAXI ⁵⁸
Objetivo: Software online administrativo para taxis	Objetivo: sistema inteligente de despacho y gestión de flotas de taxi
Ubicación: Cartagena Colombia	Ubicación: Barranquilla Colombia
Módulos que contiene: <ol style="list-style-type: none"> 1. Propietarios de vehículos 2. Administradores de taxis 3. Cooperativa de taxis 	Módulos que contiene: <ol style="list-style-type: none"> 1. Aplicación taxista 2. Aplicación pasajero 3. Central de llamadas y administrador 4. Monitoreo de propietarios

Tabla 3: información general admintaxi y gtaxi

⁵⁷ Admintaxi disponible en: http://admintaxi.com/?page_id=362

⁵⁸ g- taxi disponible en: <http://www.gtaxi.co/>

LOGIRASTREOS S.A.S ⁵⁹	SI LOG (SISTEMA INFORMACIÓN LOGISTICO) ⁶⁰
Objetivo: especializada en brindar Logística a las flotas de vehículos a través del rastreo satelital	Objetivo: empresa especializada en el desarrollo de software e integración de productos de alta tecnología para el sector del transporte.
Ubicación: Colombia	Ubicación: Bogota Colombia
Módulos: logi buses Logicarga Logitaxis logiespeciales	Módulos: Si taxis Si gps Si carga Si móvil Si tiquete Si giros

Tabla 4: información general Logirastreos, SILOG

⁵⁹ Sistema de control de flotas disponible en: <http://www.logirastreo.com/control.html#>

⁶⁰ Sistema información logístico disponible en: <http://www.sitrans.com.co/productos.html>

7.1 MARCO LEGAL

LEY 1581 DE 2012

OBJETO “La presente ley tiene por objeto desarrollar el derecho constitucional que tienen todas las personas a conocer, actualizar y rectificar las informaciones que se hayan recogido sobre ellas en bases de datos o archivos, y los demás derechos, libertades y garantías constitucionales a que se refiere el artículo 15 de la Constitución Política; así como el derecho a la información consagrado en el artículo 20 de la misma”⁶¹.

Las disposiciones y principios que contiene esta ley, son aplicables a los datos personales (aquellos de carácter privado o íntimo y que pueden representar algún tipo de vulnerabilidad, como raza, sexo, religión), registrados en bases de datos, que por tanto sean susceptibles al tratamiento por entidades públicas o privadas y que sean usados para otros fines sin expresa autorización del titular.

DECRETO 1377 de 2013

OBJETO “El presente Decreto tiene como objeto reglamentar parcialmente la Ley 1581 de 2012, por la cual se dictan disposiciones generales para la protección de datos personales”⁶².

El decreto 1377 reglamenta parcialmente la ley 1582 de 2012, ahondando en la diferencia entre datos públicos y privados, la diferencia de una base de datos personal y publica y como para las primeras no era aplicable la ley, de igual manera se aclara que los datos privados pueden ser transmitidos solo bajo autorización expresa del titular de estos.

⁶¹ Congreso de Colombia, 2012, ley estatutaria No 1581 de 2012 “ por el cual se dictan disipaciones generales para la protección de datos personales”

⁶² Gobierno de la república de Colombia,2013, decreto 1377 de 2013” Por el cual se reglamenta parcialmente la Ley 1581 de 2012”

8 CAPÍTULO IV-DEFINICIÓN DE LOS REQUERIMIENTOS DEL SISTEMA

Los requerimientos del sistema se construyeron a partir de la técnica de historias de usuario (HU), como lo indica el marco de trabajo SCRUM este se dividen en iteraciones llamadas SPRINT, cada una de las HU pertenecen a uno de los sprint.

Dentro de la metodología de ingeniería utilizada para el desarrollo del proyecto se dividieron las HU en tres sprint, el documento generado para esto se conoce como product backlog uno de los artefactos de SCRUM. (ver ANEXO 1 product backlog).

8.1 SPRINT 1

8.1.1 HISTORIAS DE USUARIO

Historia de Usuario	
Número: 1	Nombre: Portada
Prioridad en Negocio:	Iteración Asignada:
Descripción: El sistema de información tendrá una interfaz de inicio que cuente con los campos de ingreso al sistema para usuarios registrados y la opción de ingreso como invitado.	
Observaciones:	

Tabla 5: Historia de usuario 1

Historia de Usuario	
Número: 2	Nombre: Tipos de perfiles
Prioridad en Negocio:	Iteración Asignada:

<p>Descripción:</p> <p>El sistema de información tendrá diferentes tipos de usuario:</p> <ul style="list-style-type: none"> - Administrador - Operario - Invitado
<p>Observaciones:</p> <p>Cada tipo de usuario posee diferentes permisos en el sistema, los cuales son descritos más adelante en el documento.</p>

Tabla 6: Historia de usuario 2

Historia de Usuario	
Número: 3	Nombre: Recuperar contraseña
Prioridad en Negocio:	Iteración Asignada:
<p>Descripción:</p> <p>el sistema de información tendrá la opción de recuperar contraseña para un usuario del tipo administrador y operario.</p>	
<p>Observaciones:</p> <p>La recuperación se realiza a través de un correo proporcionado por el usuario al momento de ser creada su cuenta.</p>	

Tabla 7: Historia de usuario 3

Historia de Usuario

Número: 4	Nombre: Cambio de contraseña
Prioridad en Negocio:	Iteración Asignada:
Descripción:	
El sistema de información tendrá la opción de cambio de contraseña para los usuarios registrados.	
Observaciones:	
Para el cambio de contraseña se deberá ingresar la contraseña actual, así como la nueva.	

Tabla 8: Historia de usuario 4

Historia de Usuario	
Número: 5	Nombre: Creación del usuario administrador
Prioridad en Negocio:	Iteración Asignada:
Descripción:	
El sistema de información debe permitir la creación de usuarios del tipo administrador, estos solo podrán ser creados a través de la cuenta de otro administrador.	
Observaciones:	
El sistema debe tener un administrador inicial.	

Tabla 9: Historia de usuario 5

Historia de Usuario	
Número: 6	Nombre: Creación de usuario operario
Prioridad en Negocio:	Iteración Asignada:
Descripción:	
<p>El sistema de información debe permitir la creación de usuarios del tipo operario solo a través de la cuenta de un administrador en el módulo de gestión de operarios.</p>	
Observaciones:	

Tabla 10: Historia de usuario 6

Historia de Usuario	
Número: 7	Nombre: Usuario administrador
Prioridad en Negocio:	Iteración Asignada:
Descripción:	
<p>El sistema de información a través del usuario administrador podrá realizar las siguientes actividades:</p> <ul style="list-style-type: none"> • Datos básicos. • Creación de un nuevo administrador. • Consulta de vehículo, conductores y operarios registrados. • Acceso al módulo gestión de operarios, en donde podrá: <ul style="list-style-type: none"> - Crear y eliminar un operario. - Realizar consultas sobre los datos del operario. - Asignar turnos. • Acceso al sub módulo de información general de la empresa. • Eliminación de usuarios. 	

<ul style="list-style-type: none"> • Eliminar cuenta.
<p>Observaciones:</p> <p>El usuario administrador no podrá eliminar la cuenta de otro administrador.</p>

Tabla 11: Historia de usuario 7

Historia de Usuario	
Número: 8	Nombre: Datos básicos
Prioridad en Negocio:	Iteración Asignada:
<p>Descripción:</p> <p>Los diferentes usuarios del sistema podrán gestionar su información básica tal como:</p> <ul style="list-style-type: none"> • Fecha de nacimiento. • correo electrónico 	
Observaciones:	

Tabla 12: Historia de usuario 8

Historia de Usuario	
Número: 9	Nombre: Sub módulo información general
Prioridad en Negocio:	Iteración Asignada:
<p>Descripción:</p> <p>El administrador debe tener acceso (gestión) a la información de los trabajadores de la empresa, además de permitir generar los siguientes tipos</p>	

de informes: <ul style="list-style-type: none"> • Información individual y listada del personal administrativo. • Información individual y listada de los propietarios y conductores de los vehículos. • Información individual y listada de los vehículos. • Información individual y listada de los accidentes de los vehículos, de los despachos, ingresos, horarios.
Observaciones:

Tabla 13: Historia de usuario 9

Historia de Usuario	
Número: 10	Nombre: Eliminación de usuarios
Prioridad en Negocio:	Iteración Asignada:
Descripción: El usuario administrador podrá eliminar las cuentas de usuarios.	
Observaciones: El usuario administrador no podrá eliminar la cuenta de otro administrador.	

Tabla 14: Historia de usuario 10

Historia de Usuario	
Número: 11	Nombre: Eliminación de cuenta.
Prioridad en Negocio:	Iteración Asignada:
Descripción:	

Un usuario administrador podrá eliminar su propia cuenta.
<p>Observaciones:</p> <p>El administrador no podrá borrar su cuenta en caso de ser el único administrador.</p>

Tabla 15: Historia de usuario 11

Historia de Usuario	
Número: 15	Nombre: Módulo de gestión de operarios
Prioridad en Negocio:	Iteración Asignada:
<p>Descripción:</p> <p>El sistema debe permitir la adición de un nuevo operario además de los sub módulos</p> <ul style="list-style-type: none"> • datos del operario • asignación de turnos. 	
<p>Observaciones:</p> <p>La adición se hace desde la cuenta de un administrador.</p>	

Tabla 16: Historia de usuario 15

Historia de Usuario	
Número: 16	Nombre: Sub módulo datos del

	operario
Prioridad en Negocio:	Iteración Asignada:
<p>Descripción:</p> <p>El sub módulo tendrá la opción de gestión de hoja de vida y de documentación.</p>	
<p>Observaciones:</p> <p>Cada usuario tiene diferentes privilegios los cuales son especificados en las historias de usuario #7 y #12</p>	

Tabla 17: Historia de usuario 16

Historia de Usuario	
Número: 17	Nombre: Sub módulo turnos
Prioridad en Negocio:	Iteración Asignada:
<p>Descripción:</p> <p>El sub módulo tendrá la opción de gestión de los turnos de trabajo asignados al operario, así como la consulta de horas trabajadas y pagos realizados.</p>	
<p>Observaciones:</p> <p>Cada usuario tiene diferentes privilegios los cuales son especificados en las historias de usuario #7 y #12</p>	

Tabla 18: Historia de usuario 17

8.1.2 DISEÑO Y MODELADO DEL SISTEMA

La siguiente fase de la metodología de ingeniería del software es el diseño, a continuación se describen las vistas 4+1 del SPRINT.

8.1.2.1 VISTA DE ESCENARIOS (CASOS DE USO)

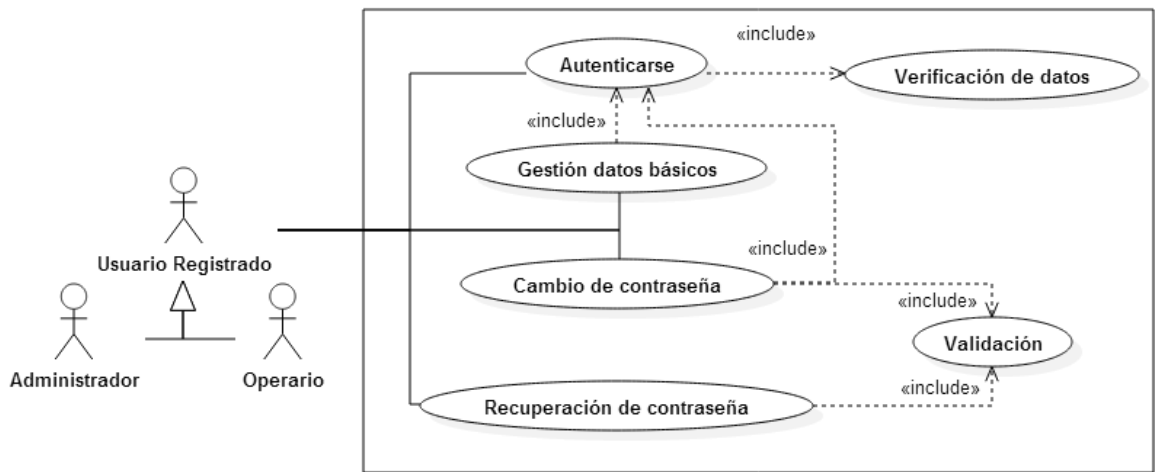


DIAGRAMA CASO DE USO: USUARIO REGISTRADO

Caso de uso # 1	
Caso de uso	Autenticarse
Actores	Usuario registrado
Propósito	Este caso de uso permite a los usuarios registrados ingresar al sistema.
Resumen	Este caso de uso empieza cuando el usuario ingresa a una cuenta dentro del sistema.

Tipo	Esencial
Referencias	
Prerrequisito	Ingresar al sistema. (cuenta creada)
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El usuario ingresa su ID y contraseña en la página de inicio.	2. El sistema verifica la información suministrada. (Sección verificación de los datos)
	3. El sistema permite el ingreso al usuario.
Curso alternativo	
Acción 2: Si el usuario ingresa datos erróneos, el sistema debe mostrar un mensaje de "ID o contraseña incorrectas" y regresar al paso 1.	
Sección Verificación de los datos	
	1. El sistema verifica que el nombre de usuario ingresado se encuentre en la base de datos y que la contraseña sea la correcta.

Tabla 19: Caso de uso # 1

Caso de uso #2	
Caso de uso	Gestionar datos básicos
Actores	Usuario registrado
Propósito	Permitir a los usuarios la gestión de sus datos básicos.
Resumen	El usuario puede ingresar y editar la información requerida en los campos.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El usuario selecciona la opción de gestionar datos básicos.	2. El sistema despliega la interfaz de gestión de datos básicos.
3. El usuario realiza los cambios deseados y selecciona la opción guardar.	4. El sistema verifica los cambios realizados.
	5. El sistema guarda los cambios.
	6. El sistema despliega un mensaje de "cambios realizados con éxito".
Curso alternativo	

Acción 4: Si el sistema detecta algún cambio erróneo despliega un mensaje de error y vuelve a la acción 3

Tabla 20: Caso de uso # 2

Caso de uso #3	
Caso de uso	Cambiar contraseña
Actores	Usuario registrado
Propósito	Permitir al usuario cambiar su contraseña en caso de que este lo requiera.
Resumen	Si un usuario desea el cambio de su contraseña después de ingresar a su cuenta selecciona dicha opción para modificarla ingresando la contraseña actual y la nueva.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El usuario selecciona la opción de cambiar contraseña.	2. El sistema despliega un mensaje para que el usuario ingrese su contraseña actual y la nueva.
3. El usuario digita su contraseña actual y la nueva.	4. El sistema valida la contraseña actual en la base de datos y modifica por la

	contraseña nueva.
	5. el sistema envía un mensaje de cambio realizado con éxito.
Curso alterno	
Acción 3: Si el usuario ingresa un una contraseña invalida como actual, se despliega un mensaje de error y se regresa al paso 2.	
Sección validación de nueva contraseña	
	1. El sistema verifica que la contraseña nueva ingresada cumpla con los requerimientos de seguridad para una contraseña.
Curso alterno	
Acción 1: si el usuario ingresa una contraseña que no cumple los parámetros de seguridad regresa al paso 2.	

Tabla 21: Caso de uso # 3

Caso de uso #4	
Caso de uso	Recuperar contraseña
Actores	Usuario registrado
Propósito	Permitir al usuario recuperar o cambiar su contraseña en caso de que la olvide.
Resumen	Si el usuario llega a olvidar la contraseña de ingreso, tendrá la opción de recuperar la contraseña por medio de un link enviado al correo registrado en la base

	de datos.
Tipo	Esencial
Referencias	
Prerrequisito	Ingresar al sistema (cuenta creada)
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El usuario selecciona la opción de recuperar contraseña.	2.El sistema despliega un mensaje Solicitando el ID del usuario.
3. El usuario digita su ID.	4. El sistema valida la existencia del ID en la base de datos.
	5. El sistema envía un link de cambio de contraseña al correo registrado en la base de datos del usuario. (Sección envío de datos de recuperación al correo)
	6. El sistema despliega un mensaje indicando al usuario que el link ha sido enviado a su correo.
Curso alterno	
Acción 4: Si el usuario ingresa un ID inválido, el sistema debe mostrar un mensaje de "ID inválido" y regresar al paso 3.	
Sección envío de datos de recuperación al correo	
	1. El sistema verifica que el ID del usuario exista.
	2. El sistema envía un correo al email del

	usuario con el link para el cambio de contraseña.
--	---

Tabla 22: Caso de uso # 4

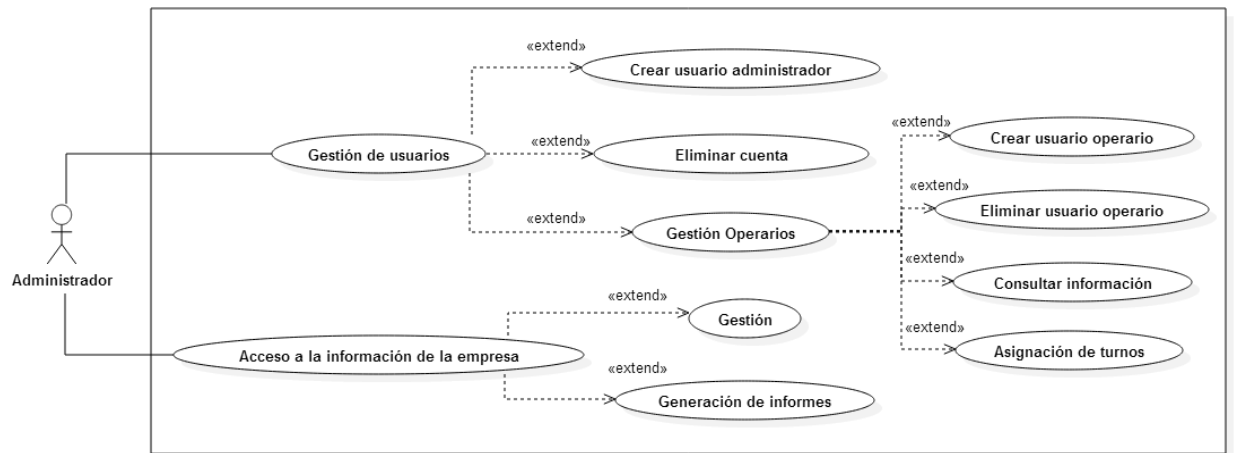


DIAGRAMA CASO DE USO: USUARIO ADMINISTRADOR

Caso de uso #5	
Caso de uso	Crear usuario administrador
Actores	Administrador
Propósito	Permitir al usuario administrador la creación de otros administradores en el sistema
Resumen	El administrador tiene la posibilidad de crear otros administradores
Tipo	Esencial

Referencias	
Prerrequisito	Caso de uso #1
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El administrador selecciona la opción de crear nuevo administrador.	2. El sistema despliega un formulario con los datos del nuevo administrador.
3. El administrador ingresa los datos correspondientes.	4. El sistema valida los datos y los guarda en la base de datos.
	5. El sistema despliega un mensaje de éxito.
Curso alternativo	
Acción 4: Si el usuario ingresa un dato inválido o deja campos vacíos, el sistema debe mostrar un mensaje de error y regresar al paso 3.	

Tabla 23: Caso de uso # 5

Caso de uso #6	
Caso de uso	Eliminar cuenta
Actores	Administrador
Propósito	Permitir al usuario administrador la eliminación de su propia cuenta
Resumen	El administrador tiene la posibilidad de

	eliminar su cuenta
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El administrador selecciona la opción eliminar cuenta.	2. El sistema despliega un mensaje de confirmación.
3. El administrador confirma la eliminación de la cuenta.	4. El sistema verifica la base de datos.
	5. El sistema elimina la cuenta.
Curso alternativo	
Acción 4: Si el usuario es el único administrador en el sistema, su cuenta no se elimina.	

Tabla 24: Caso de uso # 6

Caso de uso #7	
Caso de uso	Crear usuario operario
Actores	Administrador
Propósito	Permitir al usuario administrador la

	creación de usuarios operarios.
Resumen	El administrador puede crear cuentas para operarios.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El administrador ingresa al módulo gestión de operarios.	2. El sistema muestra el módulo (teniendo en cuenta los permisos que tiene el usuario).
3. El administrador selecciona la opción crear operario.	4. El sistema despliega la interfaz para la creación del operario.
5. El administrador ingresa los datos del operario.	6. El sistema verifica los datos.
	7. El sistema crea la cuenta.
Curso alterno	
Acción 6: Si el usuario ingresa un dato inválido o deja campos vacíos, el sistema debe mostrar un mensaje de error y regresar al paso 5.	

Tabla 25: Caso de uso # 7

Caso de uso #8	
Caso de uso	Eliminar usuario operario
Actores	Administrador
Propósito	Permitir al usuario administrador la eliminación de operarios
Resumen	El administrador puede eliminar la cuenta de un operario.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1, Caso de uso #7
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El administrador ingresa al módulo gestión de operarios.	2. El sistema muestra el módulo (teniendo en cuenta los permisos que tiene el usuario), despliega la lista de usuarios operarios.
3. El administrador elige la opción eliminar usuario.	
	4. El sistema elimina la cuenta

Tabla 26: Caso de uso # 8

Caso de uso #9	
Caso de uso	Consultar información de operario
Actores	Administrador
Propósito	Permitir al usuario administrador consultar la información de un usuario operario.
Resumen	El administrador puede consultar información sobre un operario.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1, Caso de uso #7
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El administrador ingresa al módulo gestión de operarios.	2. El sistema muestra el módulo (teniendo en cuenta los permisos que tiene el usuario), despliega la lista de usuarios operarios.
3. El administrador elige la opción consultar información de operario.	
	4. El sistema despliega la información del usuario.

Tabla 27: Caso de uso # 9

Caso de uso #10	
Caso de uso	Asignación de turnos
Actores	Administrador
Propósito	Permitir al usuario administrador asignar turnos al usuario operario.
Resumen	El administrador puedo gestionar los turnos de los operarios.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1, Caso de uso #7, Relacionado con el Caso de uso #17
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El administrador ingresa al módulo gestión de operarios.	2. El sistema muestra el módulo (teniendo en cuenta los permisos que tiene el usuario), despliega la lista de usuarios operarios.
3. El administrador elige la opción asignación de turnos.	
	6. El sistema muestra la interfaz de turnos.
7. El administrador ingresa el turno.	8. El sistema valida la información
	9. El sistema almacena la información

Curso alterno
Acción 8: Si el usuario ingresa un dato inválido o deja campos vacíos, el sistema debe mostrar un mensaje de error y regresar al paso 7.

Tabla 28: Caso de uso # 10

Caso de uso #11	
Caso de uso	Acceso a la información de la empresa
Actores	Administrador
Propósito	Permitir al usuario administrador acceder a la información de la empresa.
Resumen	El administrador puede gestionar la información de la empresa.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El administrador ingresa al sub módulo de información general.	2. El sistema muestra el sub módulo.
4. El administrador elige la acción a realizar (Ver secciones).	3. El sistema realiza la acción (Ver secciones))
Sección gestión	

Acción 3: El sistema despliega la interfaz con la información de la empresa, así como la opción para editarla.

Acción 4: El administrador hace los cambios necesarios.

Acción 5: El sistema guarda los cambios.

Sección generación de informes

Acción 3: El sistema muestra los posibles informes a generar.

Acción 4: El administrador elige el informe necesitado.

Acción 5: El sistema genera el informe.

Tabla 29: Caso de uso # 11

8.1.2.2 VISTA DE PROCESOS (DIAGRAMAS DE SECUENCIA)

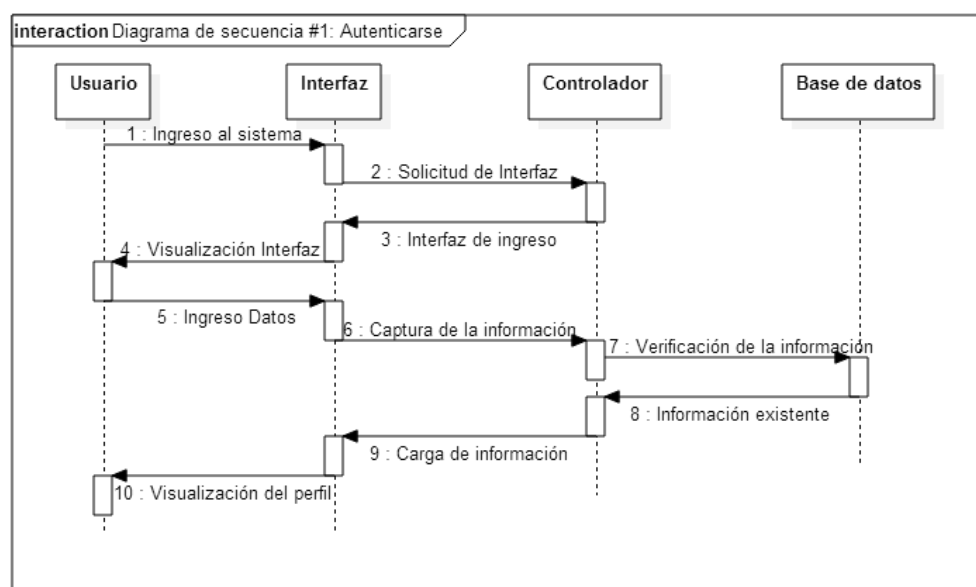


DIAGRAMA DE SECUENCIA # 1: AUTENTICARSE

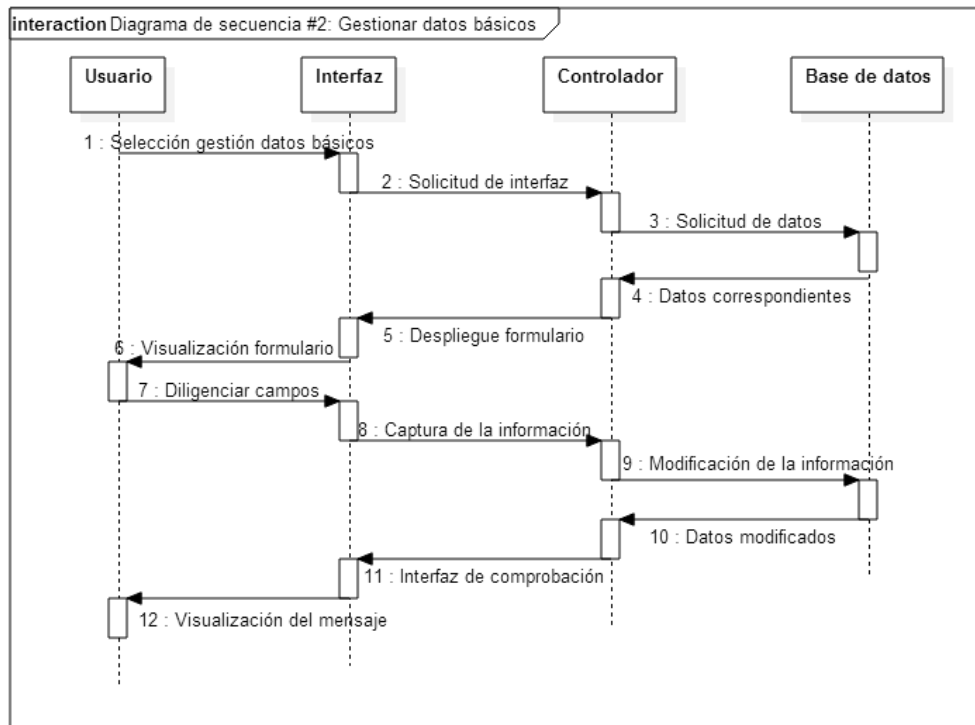


DIAGRAMA DE SECUENCIA #2: GESTIONAR DATOS BÁSICOS

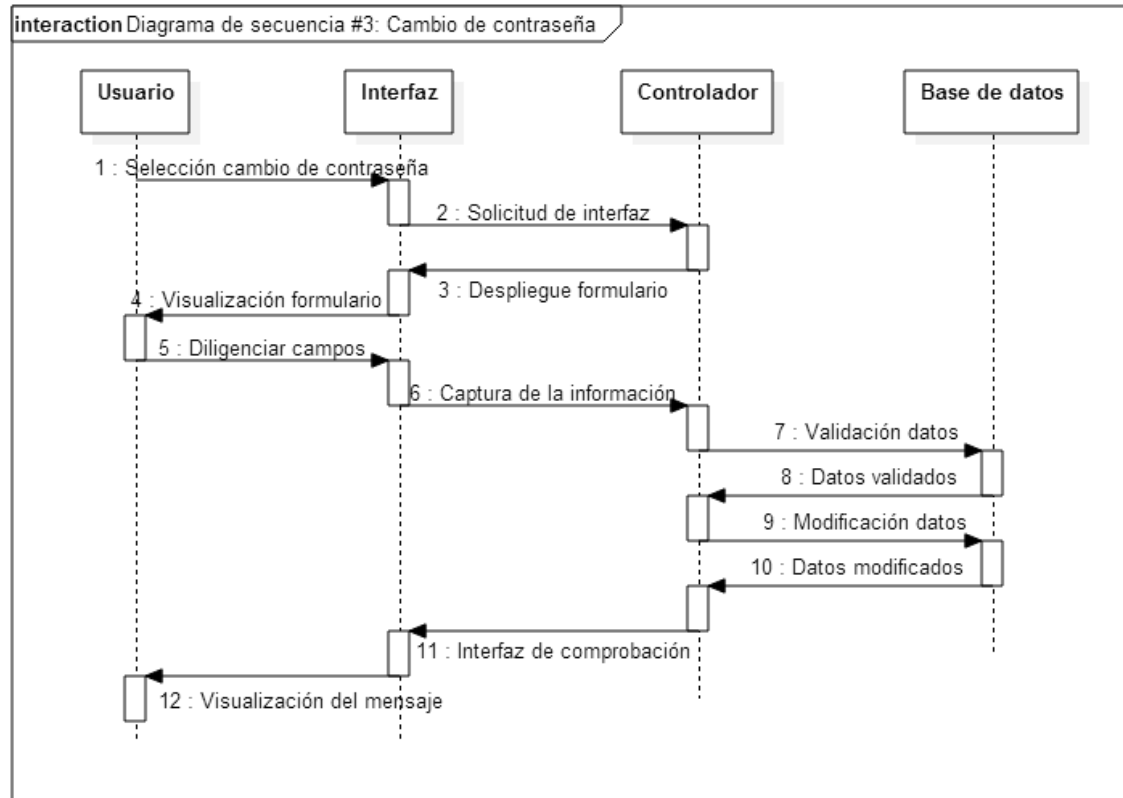


DIAGRAMA DE SECUENCIA # 3: CAMBIO DE CONTRASEÑA

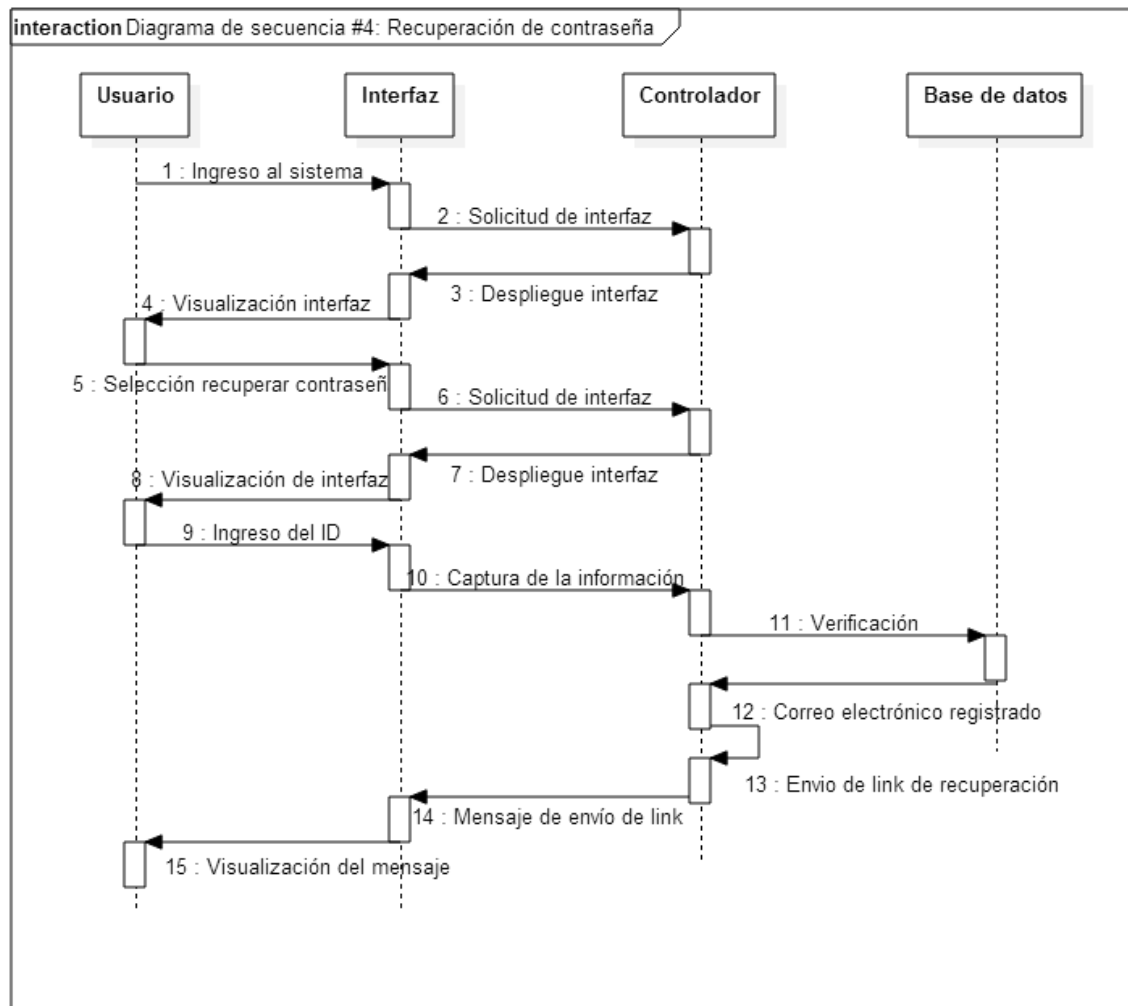


DIAGRAMA DE SECUENCIA #4: RECUPERACIÓN DE CONTRASEÑA

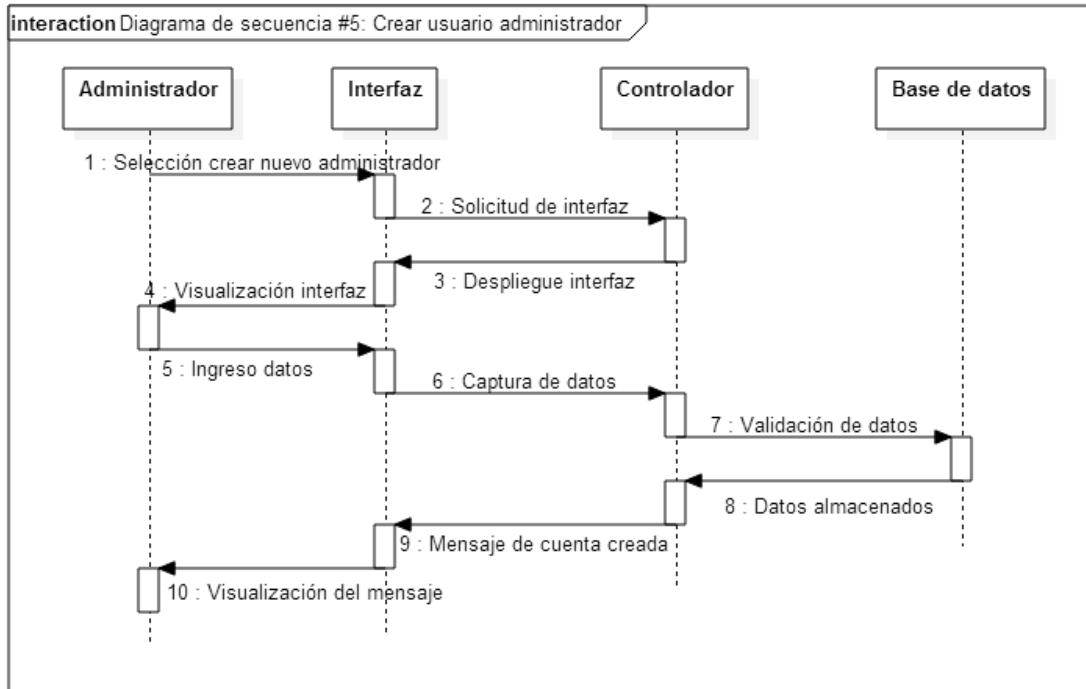


DIAGRAMA DE SECUENCIA #5: CREAR USUARIO ADMINISTRADOR

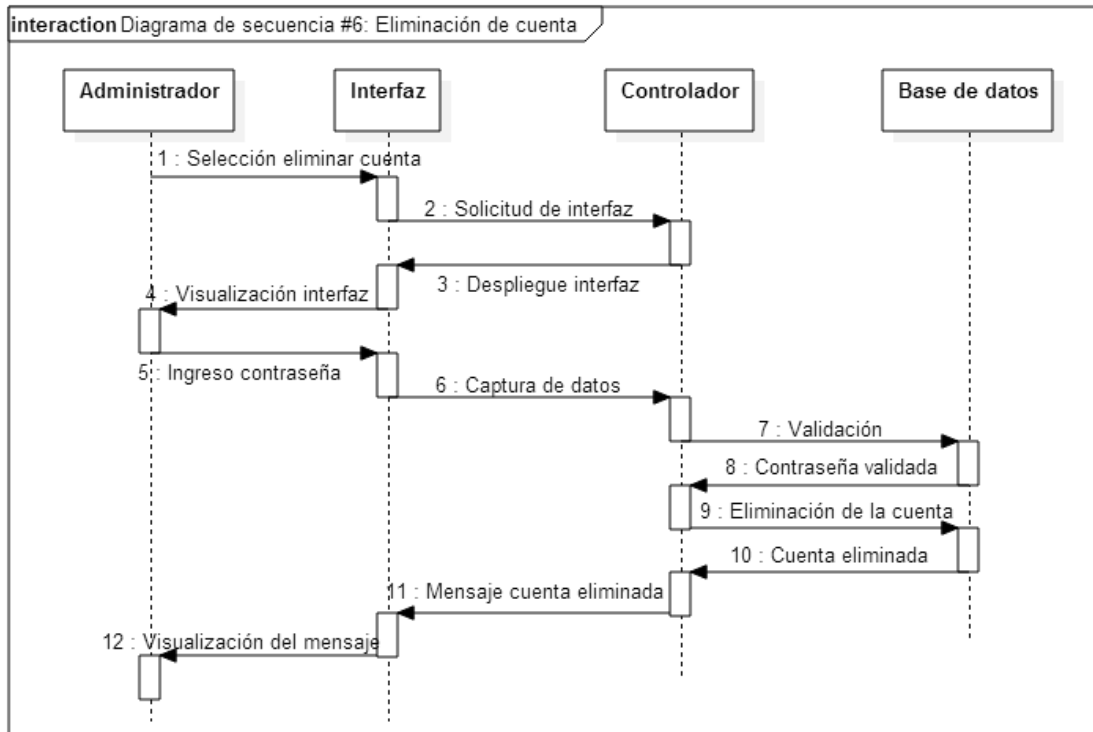


DIAGRAMA DE SECUENCIA #6: ELIMINACIÓN CUENTA

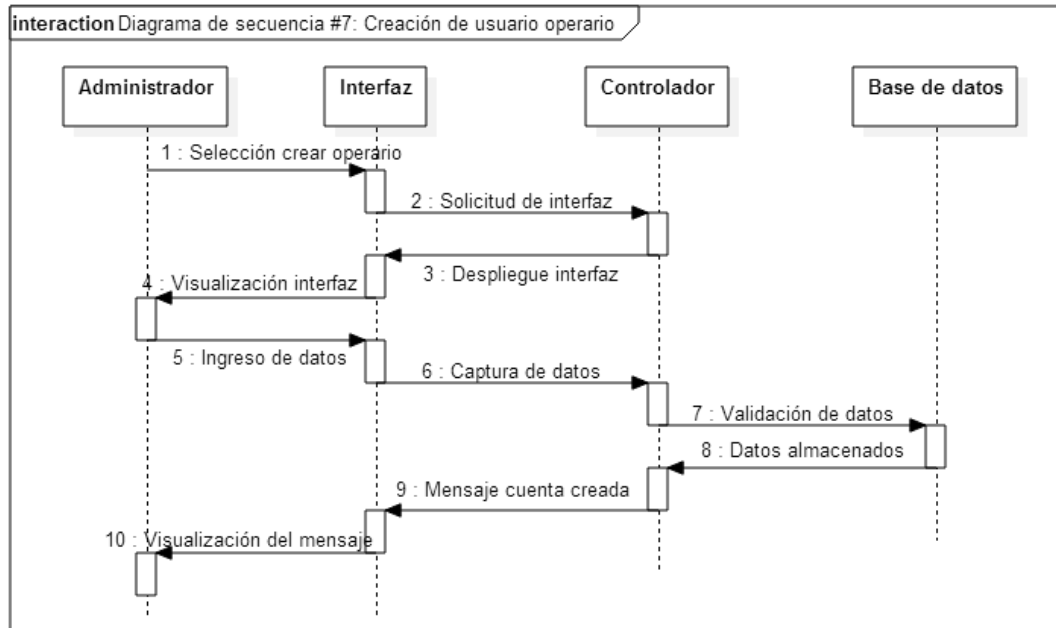


DIAGRAMA DE SECUENCIA #7: CREACIÓN DE USUARIO OPERARIO

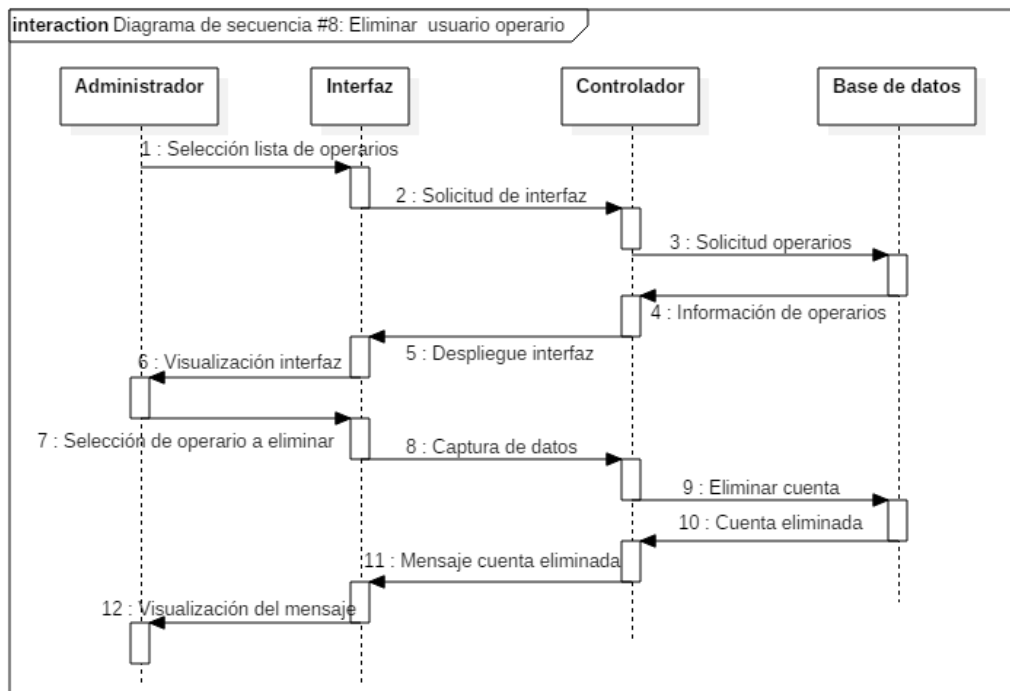


DIAGRAMA DE SECUENCIA #8: ELIMINAR USUARIO OPERARIO

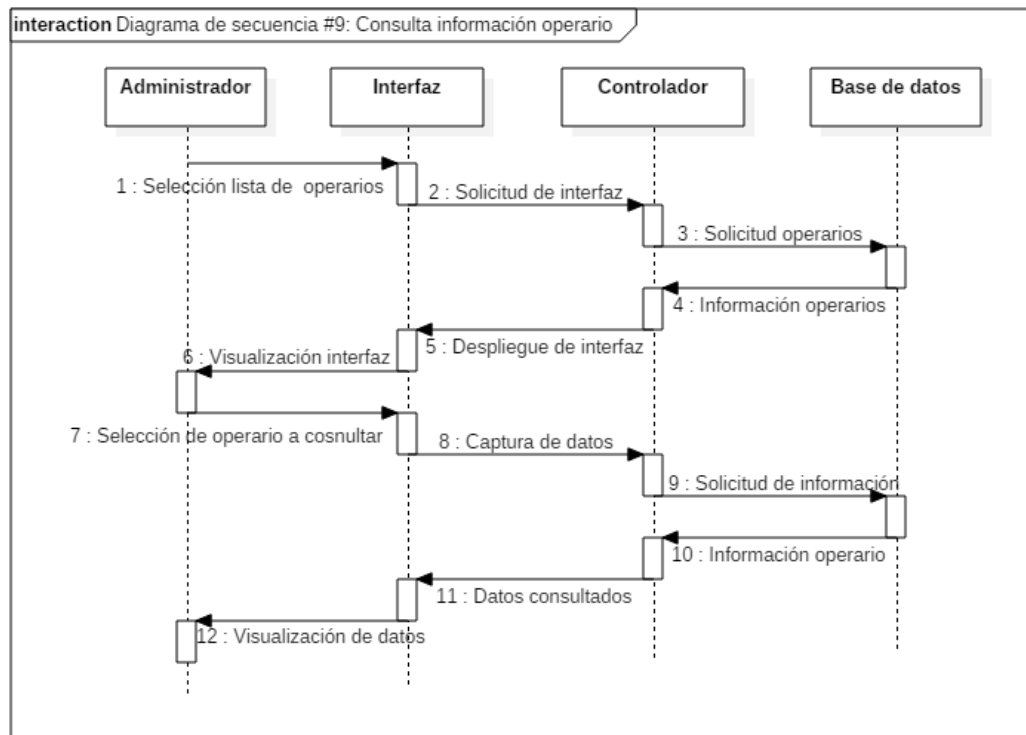


DIAGRAMA DE SECUENCIA #9: CONSULTA INFORMACIÓN OPERARIO

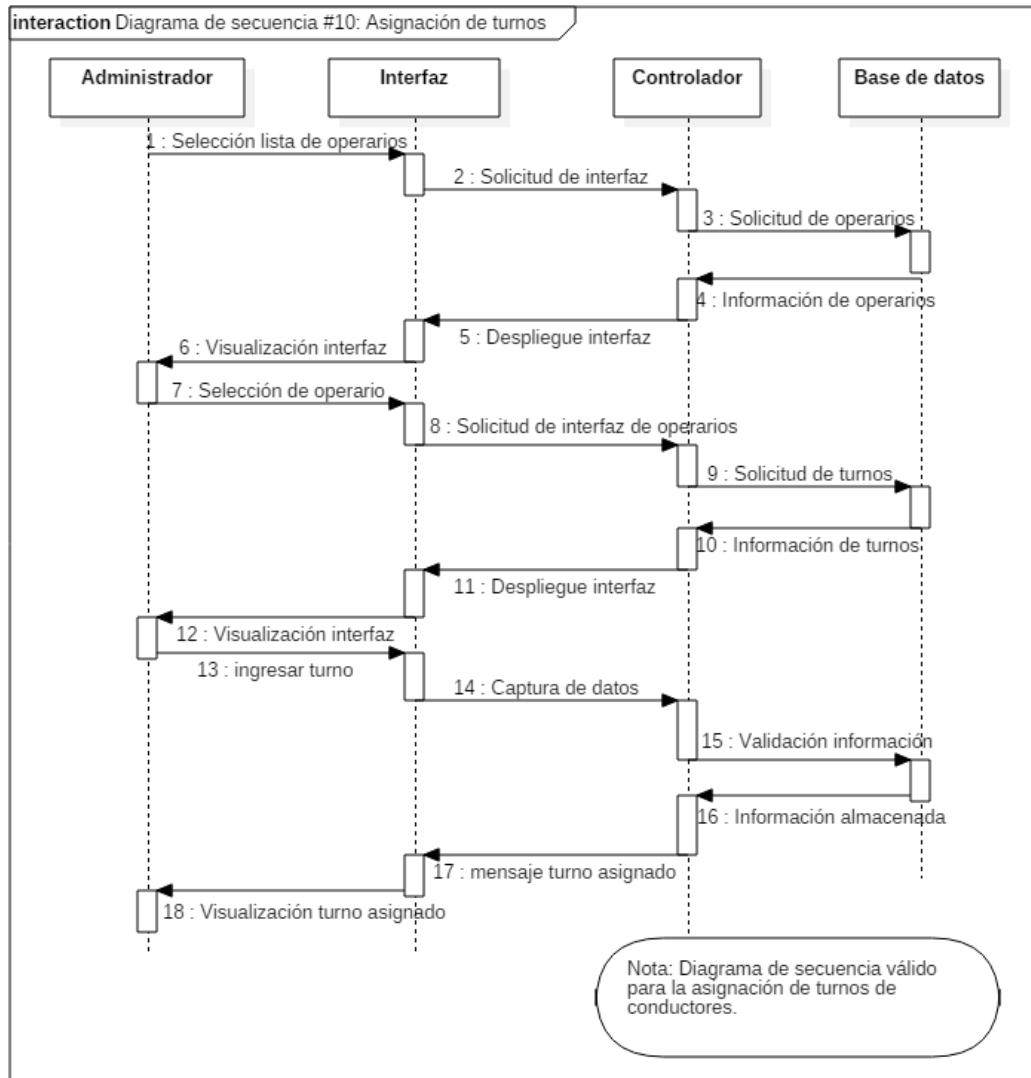


DIAGRAMA DE SECUENCIA #10: ASIGNACIÓN DE TURNOS

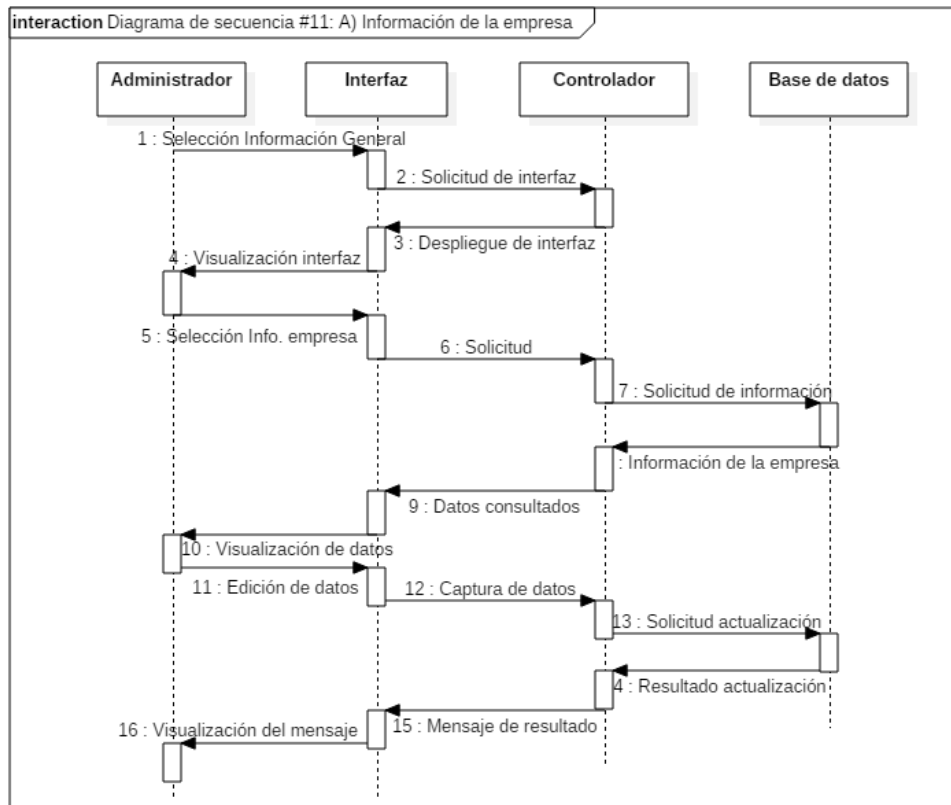


DIAGRAMA DE SECUENCIA # 11 A: INFORMACIÓN DE LA EMPRESA

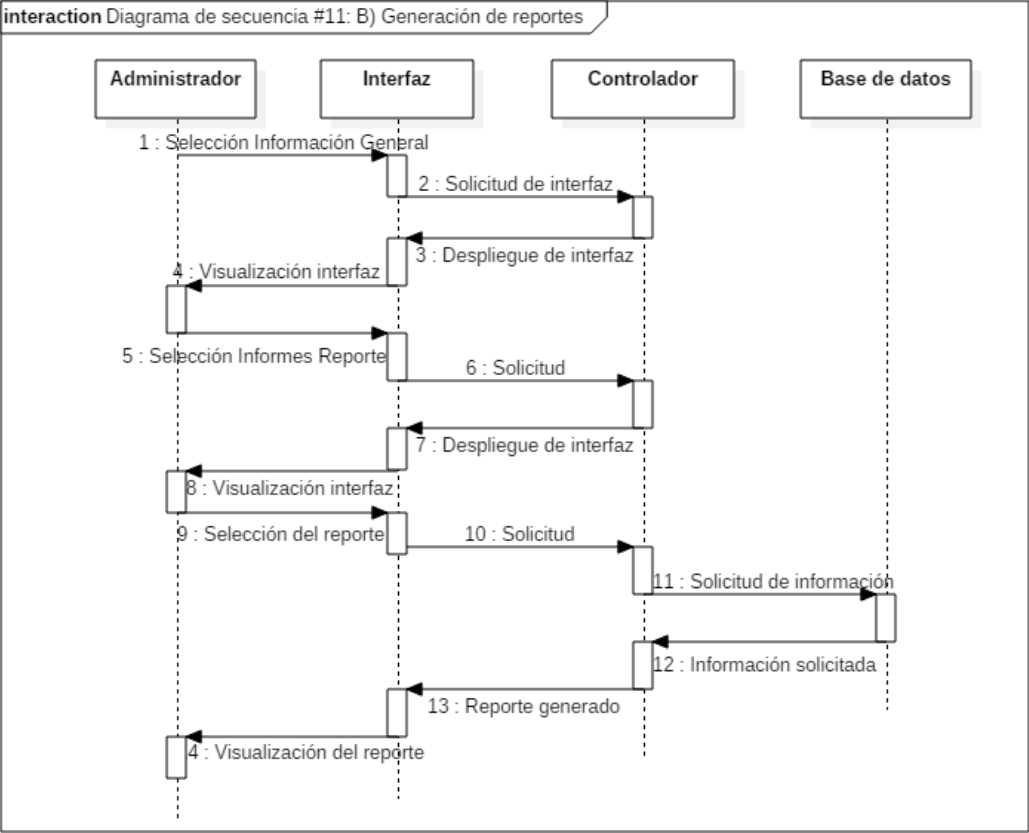


DIAGRAMA DE SECUENCIA # 11B: GENERACIÓN DE REPORTE

8.2 SPRINT 2

8.2.1 HISTORIAS DE USUARIO

Historia de Usuario	
Número: 12	Nombre: Usuario operario
Prioridad en Negocio:	Iteración Asignada:
Descripción: El sistema de información podrá a través del usuario operario realizar las siguientes actividades:	
<ul style="list-style-type: none">• Datos básicos.• Revisión de alertas.• Registro de servicios.• Consultas sobre conductores y/o vehículos.• Acceso al módulo gestión de operarios, en donde podrá:<ul style="list-style-type: none">- Gestionar su hoja de vida.- Consultar turnos.• Acceso al módulo gestión de conductores.• Acceso al módulo gestión de vehículos.• Acceso al módulo de ingresos.• Acceso al módulo de servicios realizados.	
Observaciones:	

Tabla 30: Historia de Usuario 12

Historia de Usuario	
Número: 13	Nombre: Revisión de alertas
Prioridad en Negocio:	Iteración Asignada:
Descripción:	

El sistema debe mostrar alertas en caso de expiración algún permiso de conducción o seguro.
Observaciones: Se mostrará la información básica de la alerta (conductor y/o vehículos) así como la opción para desactivar la alerta.

Tabla 31: Historia de Usuario 13

Historia de Usuario	
Número: 18	Nombre: Módulo gestión de conductores
Prioridad en Negocio:	Iteración Asignada:
Descripción: El software debe permitir la adición de un nuevo conductor, así como los submódulos datos del conductor, comparendos, siniestros y quejas, asignación de turnos y reportes.	
Observaciones: La adición se hace desde la cuenta de un operario.	

Tabla 32: Historia de Usuario 18

Historia de Usuario	
Número: 19	Nombre: Sub módulo datos del conductor
Prioridad en Negocio:	Iteración Asignada:
Descripción:	

<p>El sub módulo debe permitir la gestión de:</p> <ul style="list-style-type: none"> • hoja de vida del conductor. • Vehículo asignado. • Documentación del conductor. • Licencia del conductor.
<p>Observaciones:</p>

Tabla 33: Historia de Usuario 19

Historia de Usuario	
Número: 20	Nombre: Sub módulo comparendos, siniestros y quejas.
Prioridad en Negocio:	Iteración Asignada:
<p>Descripción:</p> <p>El sub módulo debe permitir la revisión y gestión sobre:</p> <ul style="list-style-type: none"> • Comparendos. • Siniestros. • Quejas. • Estadísticas generales del conductor. 	
<p>Observaciones:</p> <p>En caso de siniestros se debe habilitar la opción de subir imágenes.</p>	

Tabla 34: Historia de Usuario 20

Historia de Usuario	
Número: 21	Nombre: Sub módulo asignación de turnos.

Prioridad en Negocio:	Iteración Asignada:
Descripción:	
El sub módulo debe permitir el ingreso de turnos (hora inicio y finalización) así como de las comisiones de dicho turno.	
Observaciones:	
No confundir con el sub módulo turnos en gestión de operarios.	

Tabla 35: Historia de Usuario 21

8.2.2 DISEÑO Y MODELADO DEL SISTEMA

La siguiente fase de la metodología de ingeniería del software es el diseño, a continuación se describen las vistas 4+1 del SPRINT.

8.2.2.1 VISTA DE ESCENARIOS (CASOS DE USO)

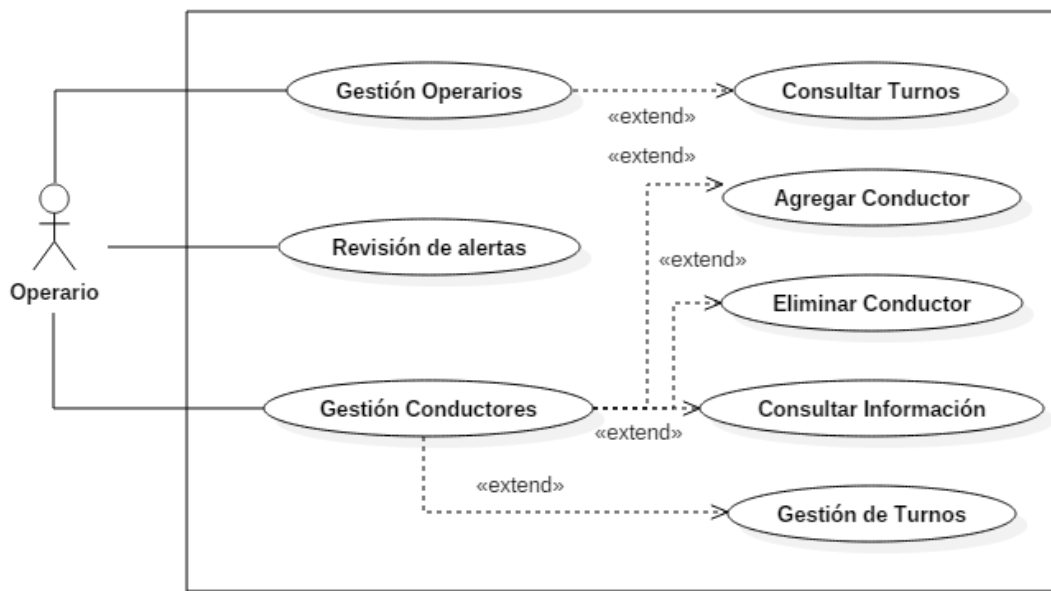


DIAGRAMA CASO DE USO: OPERARIO

Caso de uso # 12	
Caso de uso	Consultar Turnos
Actores	Operario
Propósito	Permitir al usuario Operarios la consulta y revisión de sus turnos asignados.
Resumen	El usuario tiene la posibilidad de consultar sus turnos asignados
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1, Caso de uso #10
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El operario ingresa al módulo gestión de operarios.	2. El sistema muestra el módulo (teniendo en cuenta los permisos que tiene el usuario).
3. El administrador elige la opción consultar turnos.	4. El sistema despliega la lista de turnos asignados al usuario.
7. El operario visualiza la información	
Curso alterno	
Acción 4: Si el usuario no posee turnos asignados visualizara un mensaje informando que no posee turnos.	

Tabla 36: Caso de Uso # 12

Caso de uso # 13	
Caso de uso	Revisión de alertas
Actores	Operario
Propósito	Informar al operario sobre situaciones y eventos de importancia en el funcionamiento de la empresa.
Resumen	El operario se podrá informar sobre vencimiento de licencias, revisiones técnico mecánicas y otras.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El operario ingresa al sistema.	2. El sistema mostrara un resumen de las alertas del día o semana.
3. El usuario selecciona una de las alertas.	4. El sistema despliega la información completa de la alerta
5. El usuario visualiza la información.	
Curso alterno	

Acción 2: En caso de no haber alertas en el momento el operario visualizara su página de usuario normalmente.

Tabla 37: Caso de Uso # 13

Caso de uso # 14	
Caso de uso	Agregar conductor
Actores	Operario
Propósito	Permitir al usuario operario la adición de conductores.
Resumen	El administrador puede crear registros para los conductores.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El operario ingresa al módulo gestión de conductores.	2. El sistema muestra el módulo (teniendo en cuenta los permisos que tiene el usuario).
3. El operario selecciona la opción agregar conductor.	4. El sistema despliega la interfaz para la adición del conductor.
5. El operario ingresa los datos del conductor.	6. El sistema verifica los datos.

	7. El sistema guarda la información.
Curso alterno	
Acción 6: Si el usuario ingresa un dato inválido o deja campos vacíos, el sistema debe mostrar un mensaje de error y regresar al paso 5.	

Tabla 38: Caso de Uso # 14

Caso de uso #15	
Caso de uso	Eliminar Conductor
Actores	Operario
Propósito	Permitir al usuario operario la eliminación de conductores
Resumen	El operario puede eliminar la cuenta de un conductor.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1, Caso de uso #14
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El administrador ingresa al módulo gestión de conductores.	2. El sistema muestra el módulo, despliega la lista de conductores.
3. El administrador elige la opción eliminar usuario.	

	4. El sistema elimina la cuenta
--	---------------------------------

Tabla 39: Caso de Uso # 15

Caso de uso # 16	
Caso de uso	Consultar información
Actores	Operario
Propósito	Permitir al usuario operario consultar la información de un conductor.
Resumen	El operario puede consultar información sobre un conductor.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1, Caso de uso #14
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El operario ingresa al módulo gestión de conductores.	2. El sistema muestra el módulo, despliega la lista de conductores.
3. El operario elige la opción consultar información de conductor.	
	4. El sistema despliega la información del conductor.

Tabla 40: Caso de Uso # 16

Caso de uso # 17	
Caso de uso	Gestión de Turnos
Actores	Operario
Propósito	Permitir al usuario operario asignar y consultar turnos a un conductor.
Resumen	El operario puedo gestionar los turnos de los conductores.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1, Caso de uso #14, Relacionado con el Caso de uso #10
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El operario ingresa al módulo gestión de conductores seguido ingresa al sub módulo gestión de turnos.	2. El sistema muestra el módulo, despliega la información relevante al caso.
3. El operario elige la opción de gestión (agregar, eliminar, modificar).	
	4. El sistema muestra la interfaz requerida.
7. El operario ingresa los datos necesarios.	8. El sistema valida la información

	9. El sistema almacena la información
Curso alterno	
Acción 8: Si el usuario ingresa un dato inválido o deja campos vacíos, el sistema debe mostrar un mensaje de error y regresar al paso 7.	

Tabla 41: Caso de Uso # 17

8.2.2.2 VISTA DE PROCESOS (DIAGRAMAS DE SECUENCIA)

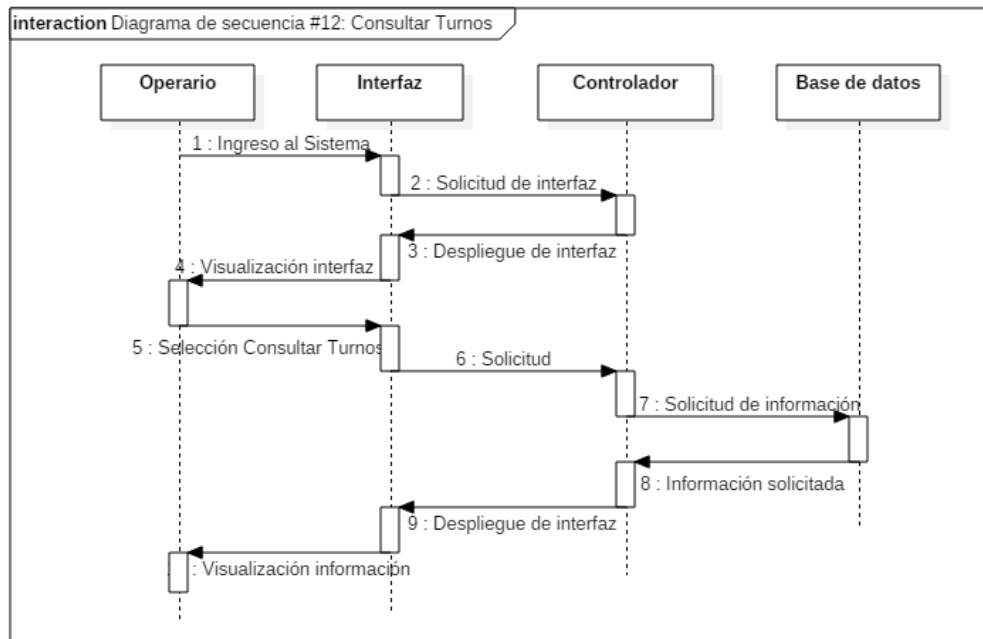


DIAGRAMA DE SECUENCIA #12: CONSULTAR TURNOS

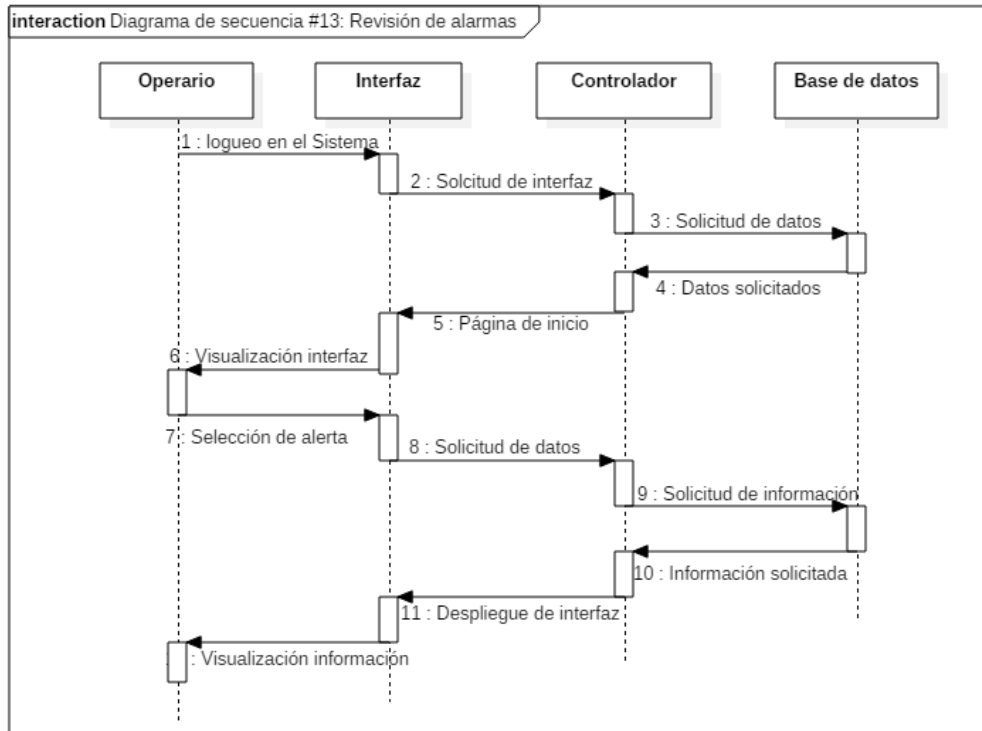


DIAGRAMA DE SECUENCIA #13: REVISIÓN DE ALARMAS

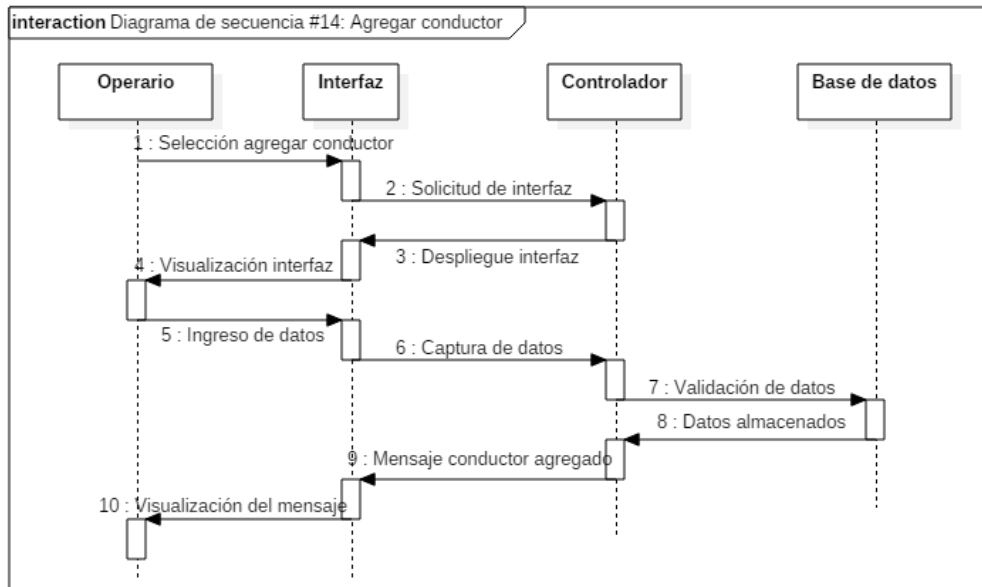


DIAGRAMA DE SECUENCIA #14: AGREGAR CONDUCTOR

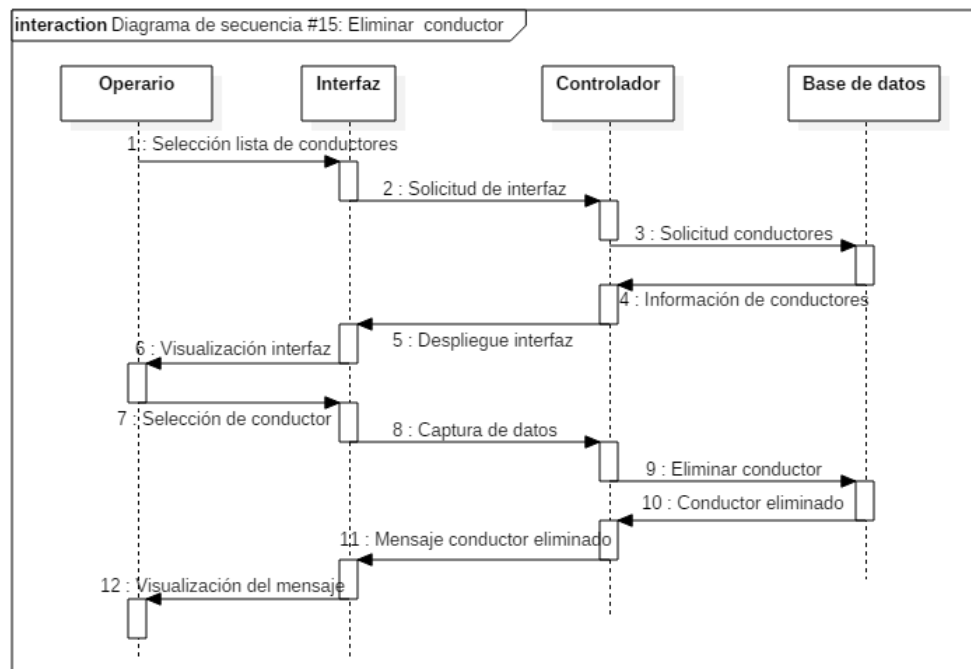


DIAGRAMA DE SECUENCIA #15: ELIMINAR CONDUCTOR

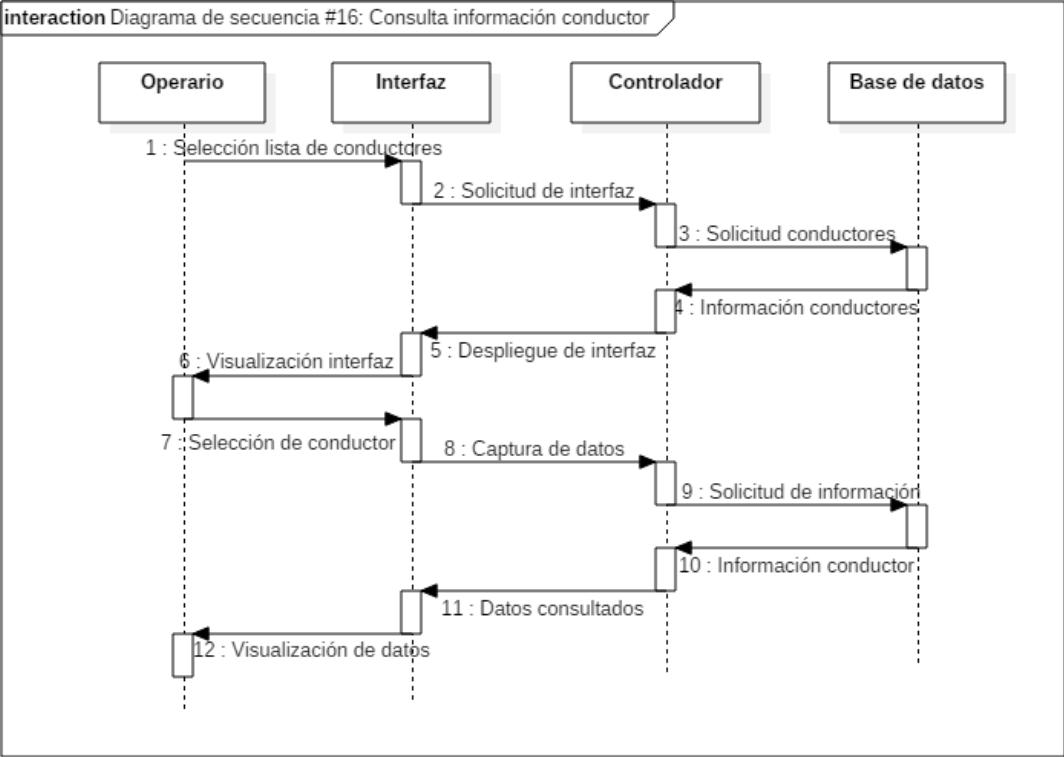


DIAGRAMA DE SECUENCIA #16: CONSULTA INFORMACIÓN CONDUCTOR

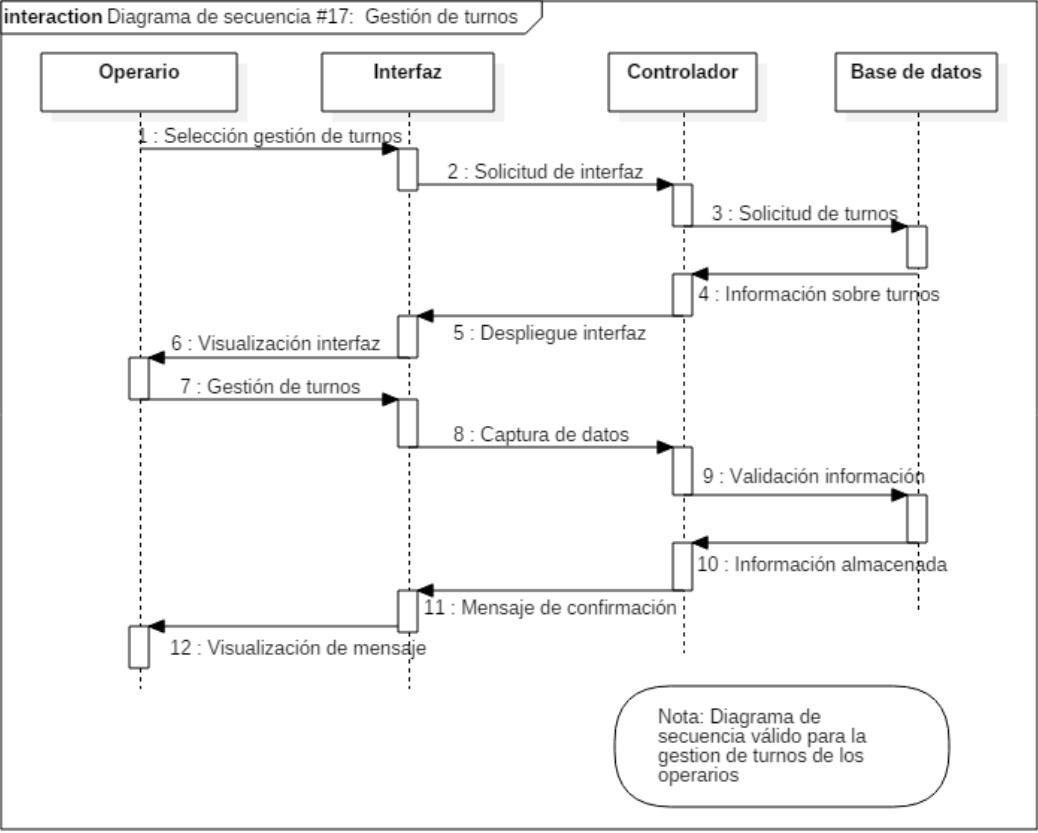


DIAGRAMA DE SECUENCIA #17: GESTIÓN DE TURNOS

8.3 SPRINT 3

8.3.1 HISTORIAS DE USUARIO

Historia de Usuario	
Número: 14	Nombre: Usuario invitado
Prioridad en Negocio:	Iteración Asignada:
Descripción: El sistema debe permitir el ingreso como invitado, donde solo se permitirá hacer ciertas consultas básicos tales como la fecha de expiración de seguros y permisos, también la posibilidad de ingresar quejas o reportes de algún tipo de incidente.	
Observaciones:	

Tabla 42: Historia de usuario 14

Historia de Usuario	
Número: 22	Nombre: Módulo de gestión de vehículos
Prioridad en Negocio:	Iteración Asignada:
Descripción: El software debe permitir la adición de un nuevo vehículo, así como de los sub módulos datos del vehículo, mantenimiento y reportes.	
Observaciones: La adición se hace desde la cuenta de un operario.	

Tabla 43: Historia de usuario 22

Historia de Usuario	
Número: 23	Nombre: Sub módulo datos del vehículo
Prioridad en Negocio:	Iteración Asignada:
Descripción:	
<p>El sub módulo debe permitir la gestión de:</p> <ul style="list-style-type: none"> • Datos básicos del vehículo (placa, marca, modelo, estado) • Conductor(es) asignado(s). • Documentos del vehículo. • Documentos por vencer. • Información básica sobre el dueño del vehículo. 	
Observaciones:	

Tabla 44: Historia de usuario 23

Historia de Usuario	
Número: 24	Nombre: sub modulo Mantenimiento
Prioridad en Negocio:	Iteración Asignada:
Descripción:	
<p>El sub módulo debe permitir la revisión y gestión de:</p> <ul style="list-style-type: none"> • Alertas de mantenimiento. • Registro de mantenimiento realizado. • Historial de mantenimiento. 	
Observaciones:	

Tabla 45: Historia de usuario 24

Historia de Usuario	
Número: 25	Nombre: Sub módulo reportes
Prioridad en Negocio:	Iteración Asignada:
Descripción:	
<p>El sub módulo de reportes debe permitir escoger los campos que se quieren dentro del informe a generar mediante un formulario.</p>	
Observaciones:	
<p>El sub módulo hace parte tanto del módulo de gestión de vehículos así como gestión de conductores.</p> <p>Dicho formulario se generara como un tipo de archivo portable (PDF).</p>	

Tabla 46: Historia de usuario 25

Historia de Usuario	
Número: 26	Nombre: Módulo ingresos
Prioridad en Negocio:	Iteración Asignada:
Descripción:	
<p>El software debe permitir la revisión e ingreso de:</p> <ul style="list-style-type: none"> • Ingresos por vehículo. • Ingresos por conductor. • Tabla de ingresos por fechas. • Ingresos por clientes y/o particulares. • Estadísticas de ingresos. 	
Observaciones:	

Tabla 47: Historia de usuario 26

Historia de Usuario	
Número: 27	Nombre: Módulo servicios realizados
Prioridad en Negocio:	Iteración Asignada:
Descripción:	
<p>El software debe permitir:</p> <ul style="list-style-type: none"> • Ingreso de servicio (tarifa, comisiones, tiempo) • Consultar servicios. • Filtrado por fechas, hora, conductor o vehículo. • Revisión de quejas. • Ingreso de incidentes presentados. 	
Observaciones:	

Tabla 48: Historia de usuario 27

8.3.2 DISEÑO Y MODELADO DEL SISTEMA

La siguiente fase de la metodología de ingeniería del software es el diseño, a continuación se describen las vistas 4+1 del SPRINT.

8.3.2.1 VISTA DE ESCENARIOS (CASOS DE USO)

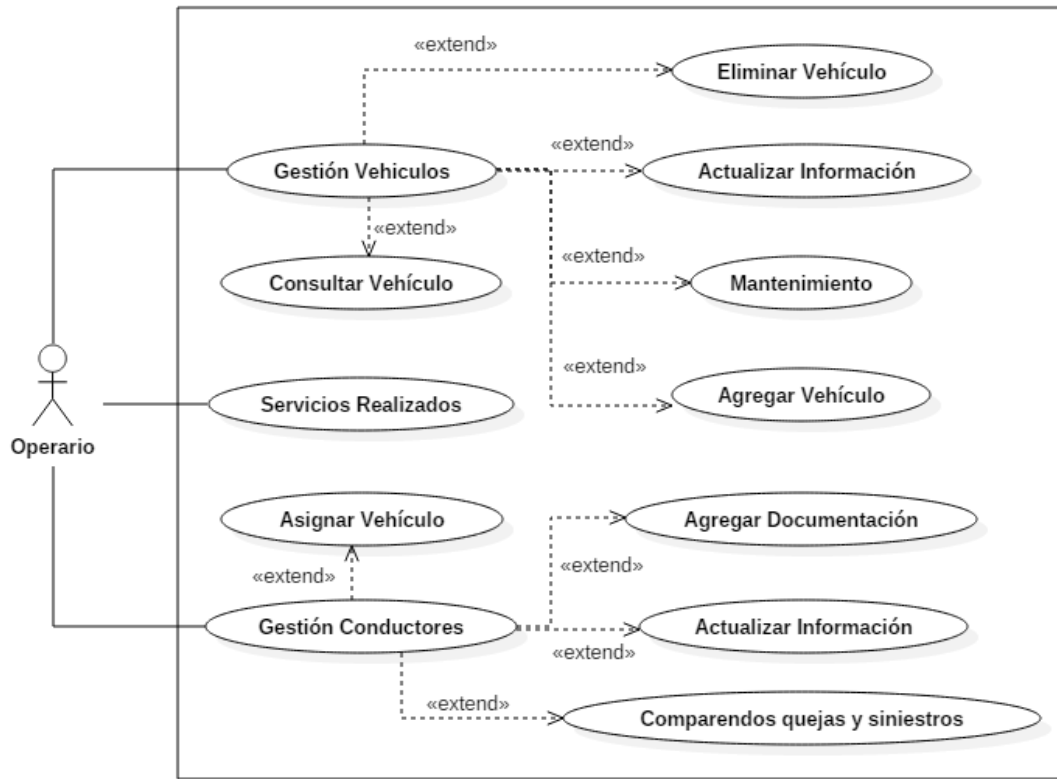


DIAGRAMA DE CASO DE USO: OPERARIO

Caso de uso # 18	
Caso de uso	Actualizar información
Actores	Operario
Propósito	Permitir al usuario operario actualizar la información de los conductores
Resumen	El operario puede modificar y actualizar información relevante.

Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1, Caso de uso #14
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El operario ingresa al módulo gestión de conductores.	2. El sistema muestra el módulo (teniendo en cuenta los permisos que tiene el usuario).
3. El operario elige la opción de actualizar información	4. El sistema muestra la interfaz requerida.
5. El operario ingresa los datos necesarios.	6. El sistema valida la información
	7. El sistema almacena la información
Curso alterno	
Acción 6: Si el usuario ingresa un dato inválido o deja campos vacíos, el sistema debe mostrar un mensaje de error y regresar al paso 5.	

Tabla 49: Caso de uso #18

Caso de uso # 19	
Caso de uso	Agregar documentación
Actores	Operario

Propósito	Permitir al usuario Operarios adicionar documentos relacionados a un conductor.
Resumen	El usuario tiene la posibilidad de adicionar documentación
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1, Caso de uso #14
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El operario ingresa al módulo gestión de conductores.	2. El sistema muestra el módulo (teniendo en cuenta los permisos que tiene el usuario).
3. El operario selecciona la opción agregar documentación.	4. El sistema despliega la interfaz para la adición de documentos.
5. El operario carga los documentos del conductor.	6. El sistema almacena los documentos.
	7. El sistema muestra mensaje de resultado
8. El operario visualiza el mensaje	

Tabla 50: Caso de uso # 19

Caso de uso # 20	
Caso de uso	Asignar vehículo

Actores	Operario
Propósito	Permitir al usuario operario asignar un vehículo al conductor.
Resumen	El operario puedo asignar vehículos a los conductores.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1, Caso de uso #14, Caso de uso #22
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El operario ingresa al módulo gestión de conductores.	2. El sistema muestra el módulo (teniendo en cuenta los permisos que tiene el usuario).
3. El operario selecciona la opción asignar vehículo.	4. El sistema muestra la interfaz requerida.
5. El operario selecciona el vehículo a asignar,	6. El sistema almacena la información
	7. El sistema almacena la información
	8. El sistema muestra mensaje de resultado
9. El usuario visualiza el mensaje	

Tabla 51: Caso de uso # 20

Caso de uso # 21	
Caso de uso	Comparendos, quejas y siniestros
Actores	Operario
Propósito	Permitir al usuario operario registrar y visualizar quejas, comparendos y siniestros relacionados al conductor.
Resumen	El operario puedo registrar y consultar estas situaciones.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1, Caso de uso #14
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El operario ingresa al módulo gestión de conductores.	2. El sistema muestra el módulo (teniendo en cuenta los permisos que tiene el usuario).
3. El operario selecciona la opción comparendos, quejas y siniestros.	4. El sistema despliega la interfaz solicitada.
5. El operario selecciona la opción (comparendo, queja o siniestro) y lo que desea hacer (registrar o consultar).	6. El sistema despliega la interfaz solicitada.
7. El usuario visualiza la información o	8. El sistema valida la información.

ingresa los datos de ser el caso.	
	9. El sistema almacena la información y muestra mensaje de resultado
10. El usuario visualiza el mensaje	
Curso alterno	
Acción 8: Si el usuario ingresa un dato inválido o deja campos vacíos, el sistema debe mostrar un mensaje de error y regresar al paso 7.	

Tabla 52: Caso de uso # 21

Caso de uso # 22	
Caso de uso	Agregar vehículo
Actores	Operario
Propósito	Permitir al usuario operario agregar vehículos al sistema.
Resumen	El operario puede agregar vehículos al sistema
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema

1. El operario ingresa al módulo gestión de vehículos.	2. El sistema muestra el módulo (teniendo en cuenta los permisos que tiene el usuario).
3. El operario selecciona la opción agregar vehículo.	4. El sistema despliega la interfaz para la adición del vehículo.
5. El operario ingresa los datos del vehículo.	6. El sistema verifica los datos.
	7. El sistema guarda la información.
	8. El sistema almacena la información
Curso alterno	
Acción 6: Si el usuario ingresa un dato inválido o deja campos vacíos, el sistema debe mostrar un mensaje de error y regresar al paso 5.	

Tabla 53: Caso de uso # 22

Caso de uso # 23	
Caso de uso	Eliminar vehículo
Actores	Operario
Propósito	Permitir al usuario operario eliminar un vehículo del sistema.
Resumen	El operario pueden eliminar vehículos del sistema.
Tipo	Esencial
Referencias	

Prerrequisito	Caso de uso #1, Caso de uso #22
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El administrador ingresa al módulo gestión de vehículos.	2. El sistema muestra el módulo, despliega la lista de vehículos.
3. El operario elige la opción eliminar vehículo.	
	4. El sistema elimina el vehículo

Tabla 54: Caso de uso # 23

Caso de uso # 24	
Caso de uso	Consultar información del vehículo
Actores	Operario
Propósito	Permitir al usuario operario consultar información sobre un vehículo.
Resumen	El operario puedo consultar información referida a un vehículo.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1, Caso de uso #22
Curso normal de los eventos	

Acción de los actores	Respuesta del sistema
1. El operario ingresa al módulo gestión de vehículos.	2. El sistema muestra el módulo, despliega la lista de vehículos.
3. El operario elige la opción consultar información del vehículo.	4. El sistema despliega la información del vehículo.

Tabla 55: Caso de uso # 24

Caso de uso # 25	
Caso de uso	Actualizar información del vehículo
Actores	Operario
Propósito	Permitir al usuario operario actualizar la información de los conductores
Resumen	El operario puede modificar y actualizar información relevante.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1, Caso de uso #22
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El operario ingresa al módulo gestión de vehículos.	2. El sistema muestra el módulo (teniendo en cuenta los permisos que tiene el usuario).

3. El operario elige la opción de actualizar información.	4. El sistema muestra la interfaz requerida.
5. El operario ingresa los datos necesarios.	6. El sistema valida la información
	7. El sistema almacena la información
	9. El sistema almacena la información
Curso alterno	
Acción 6: Si el usuario ingresa un dato inválido o deja campos vacíos, el sistema debe mostrar un mensaje de error y regresar al paso 5.	

Tabla 56: Caso de uso # 25

Caso de uso # 26	
Caso de uso	Mantenimiento
Actores	Operario
Propósito	Permitir al usuario operario registrar y visualizar el historial de mantenimiento de un vehículo.
Resumen	El operario pueden revisar y registrar actos de mantenimiento.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1, Caso de uso #22

Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El operario ingresa al módulo gestión de vehículos.	2. El sistema muestra el módulo (teniendo en cuenta los permisos que tiene el usuario).
3. El operario selecciona la opción mantenimiento.	4. El sistema despliega la interfaz solicitada.
5. El operario selecciona lo que desea hacer (registrar o consultar).	6. El sistema despliega la interfaz solicitada.
7. El usuario visualiza la información o ingresa los datos de ser el caso.	8. El sistema valida la información.
	9. El sistema almacena la información y muestra mensaje de resultado
10. El usuario visualiza el mensaje	
Curso alterno	
Acción 8: Si el usuario ingresa un dato inválido o deja campos vacíos, el sistema debe mostrar un mensaje de error y regresar al paso 7.	

Tabla 57: Caso de uso # 26

Caso de uso # 27	
Caso de uso	Servicios realizados
Actores	Operario
Propósito	Permitir al usuario operario registrar y

	visualizar el historial de servicios de un vehículo o un operario.
Resumen	El operario puedo revisar y registrar servicios realizados.
Tipo	Esencial
Referencias	
Prerrequisito	Caso de uso #1, Caso de uso #14, Caso de uso #22
Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. El operario ingresa al módulo servicios realizados.	2. El sistema muestra el módulo (teniendo en cuenta los permisos que tiene el usuario).
3. El operario selecciona lo que desea hacer (consultar o registrar).	4. El sistema despliega la interfaz solicitada.
5. El usuario visualiza la información o ingresa los datos de ser el caso.	6. El sistema valida la información.
	7. El sistema almacena la información y muestra mensaje de resultado
8. El usuario visualiza el mensaje	
Curso alterno	
Acción 6: Si el usuario ingresa un dato inválido o deja campos vacíos, el sistema debe mostrar un mensaje de error y regresar al paso 5.	

Tabla 58: Caso de uso # 27

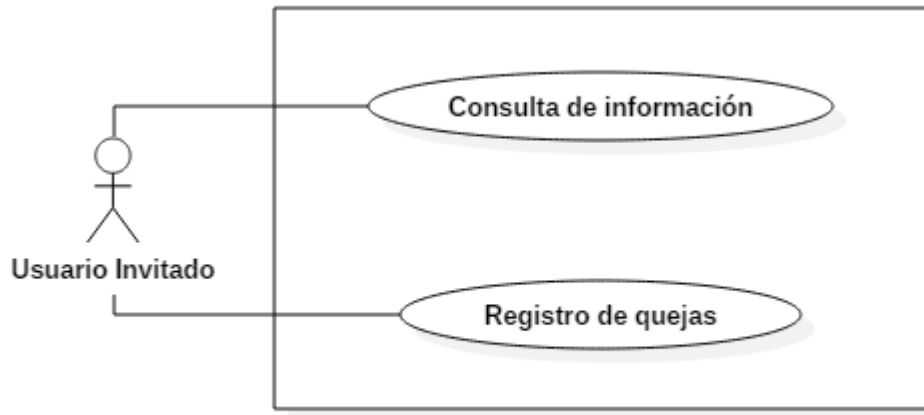


DIAGRAMA DE CASO DE USO: INVITADO

Caso de uso # 28	
Caso de uso	Consulta de información (Inv)
Actores	Usuario Invitado
Propósito	Permitir al usuario invitado consultar hacer consultas básicas.
Resumen	El usuario invitado puede consultar información.
Tipo	Esencial
Referencias	
Prerrequisito	Ingresar al sistema. (Invitado)
Curso normal de los eventos	

Acción de los actores	Respuesta del sistema
1. El usuario ingresa por la opción “Ingresar como invitado”, e ingresa su número de documento.	2. El sistema muestra las opciones válidas para invitado.
3. El usuario elige consultar, seguido de la opción de consulta deseada.	
4. El usuario ingresa el dato de búsqueda de ser necesario	5. El sistema muestra los resultados de la búsqueda.
7. El usuario visualiza la información	

Tabla 59: Caso de uso # 28

Caso de uso # 29	
Caso de uso	Registro de quejas
Actores	Operario
Propósito	Permitir al usuario invitado el registro de alguna queja por el servicio prestado.
Resumen	El usuario invitado puede registrar quejas en el servicio.
Tipo	Esencial
Referencias	
Prerrequisito	Ingresar al sistema. (Invitado)
Curso normal de los eventos	

Acción de los actores	Respuesta del sistema
1. El usuario ingresa por la opción "Ingresar como invitado", e ingresa su número de documento.	2. El sistema muestra las opciones válidas para invitado.
3. El operario elige la opción registrar queja	4. El sistema muestra la interfaz requerida.
5. El operario ingresa los datos necesarios.	6. El sistema valida la información
	7. El sistema almacena la información
Curso alterno	
Acción 6: Si el usuario ingresa un dato inválido o deja campos vacíos, el sistema debe mostrar un mensaje de error y regresar al paso 5.	

Tabla 60: Caso de uso # 29

8.3.2.2 VISTA DE PROCESOS (DIAGRAMAS DE SECUENCIA)

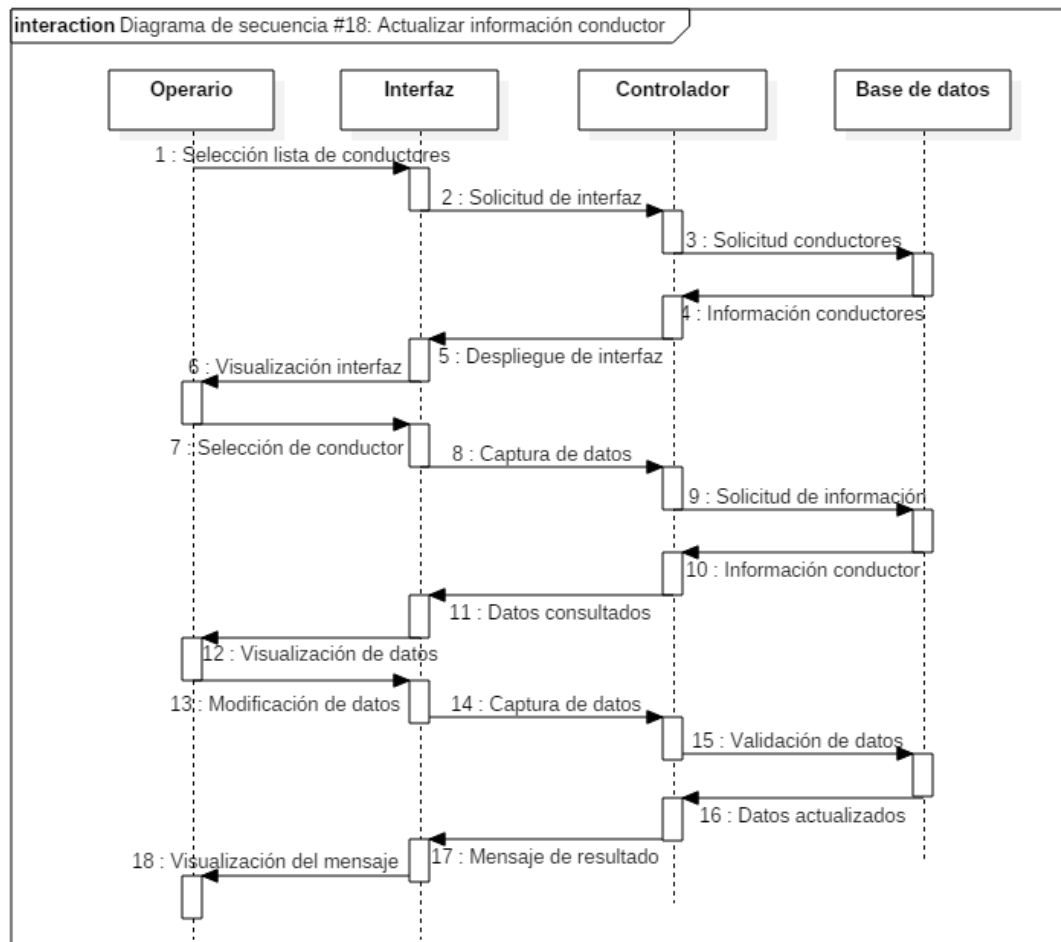


DIAGRAMA DE SECUENCIA #18: ACTUALIZAR INFORMACIÓN CONDUCTOR

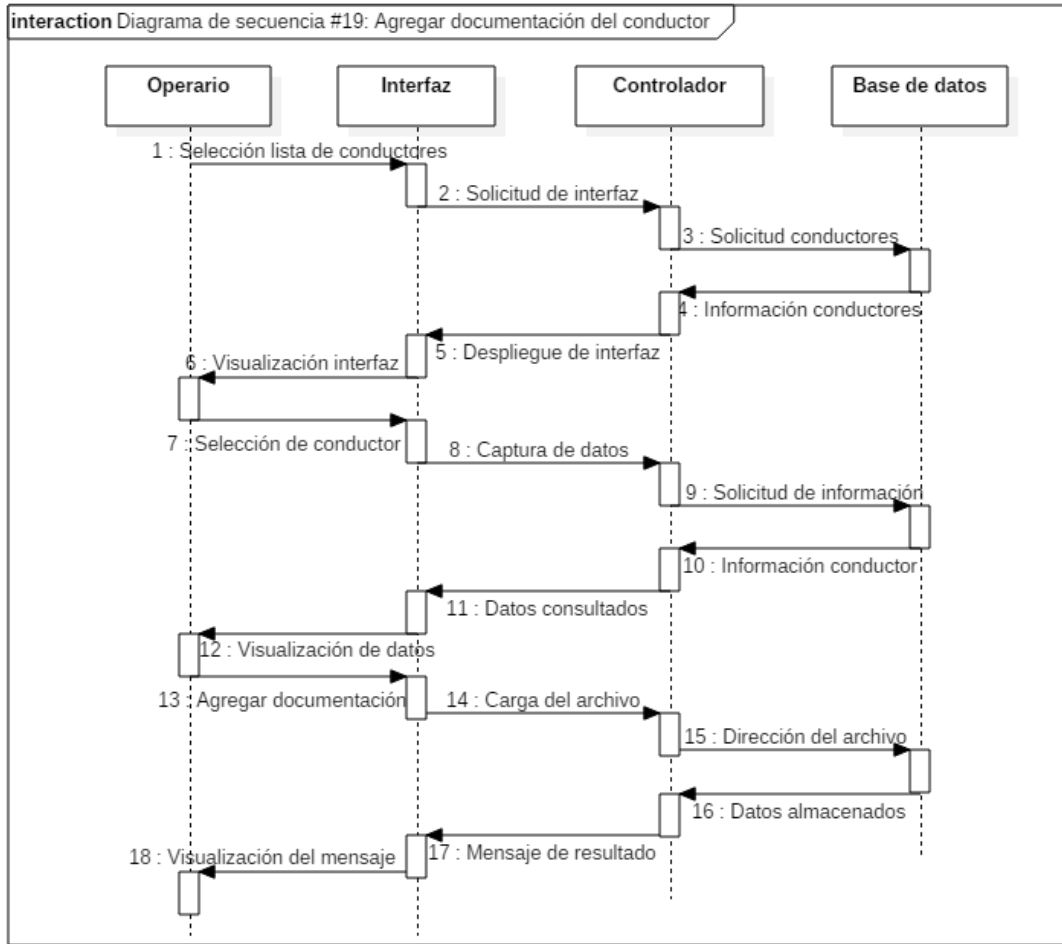


DIAGRAMA DE SECUENCIA #19: AGREGAR DOCUMENTACIÓN DEL CONDUCTOR

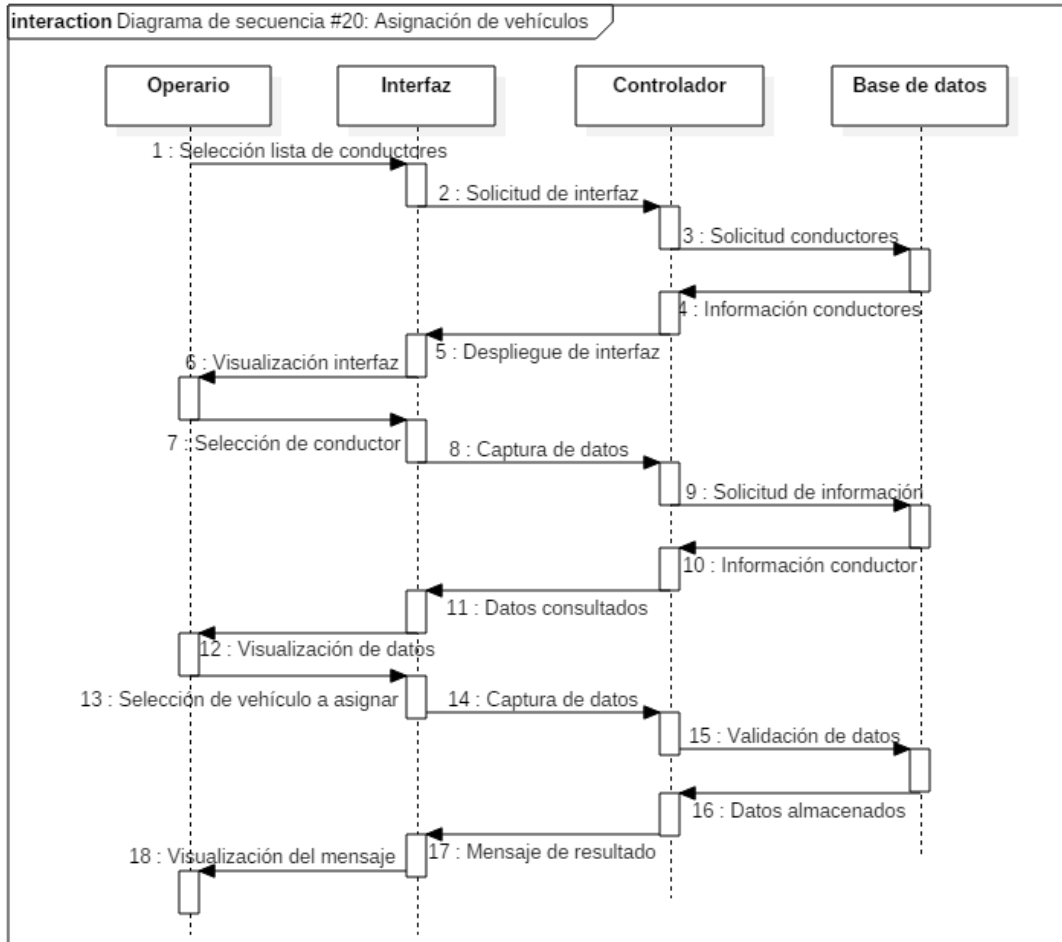


DIAGRAMA DE SECUENCIA #20: ASIGNACIÓN DE VEHÍCULOS

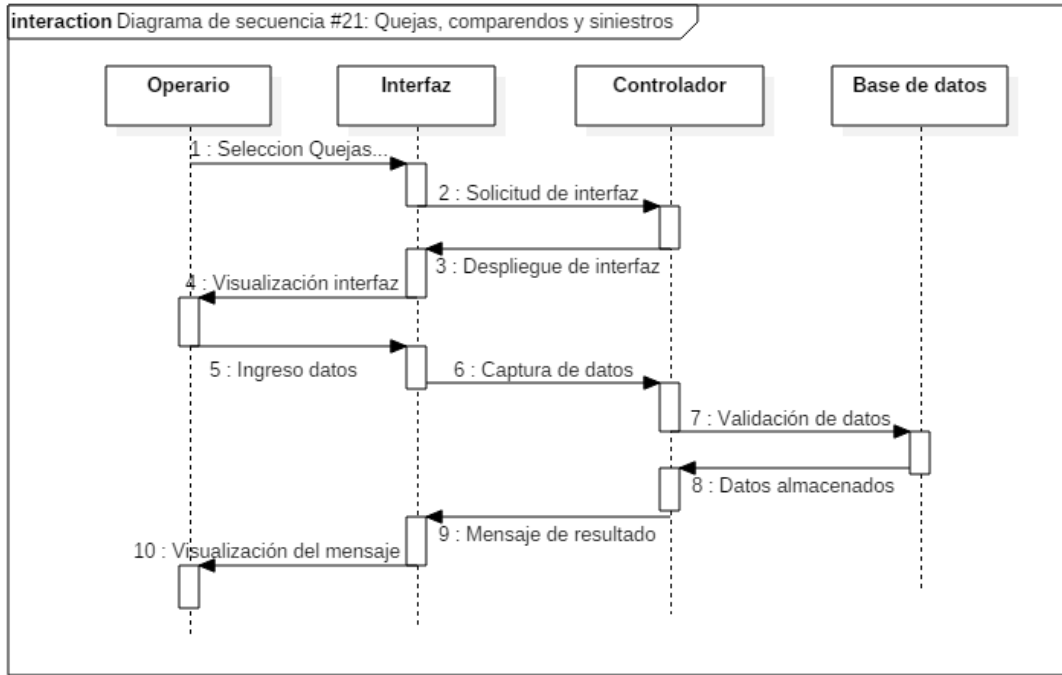


DIAGRAMA DE SECUENCIA #21: QUEJAS, COMPARENDOS Y SINIESTROS

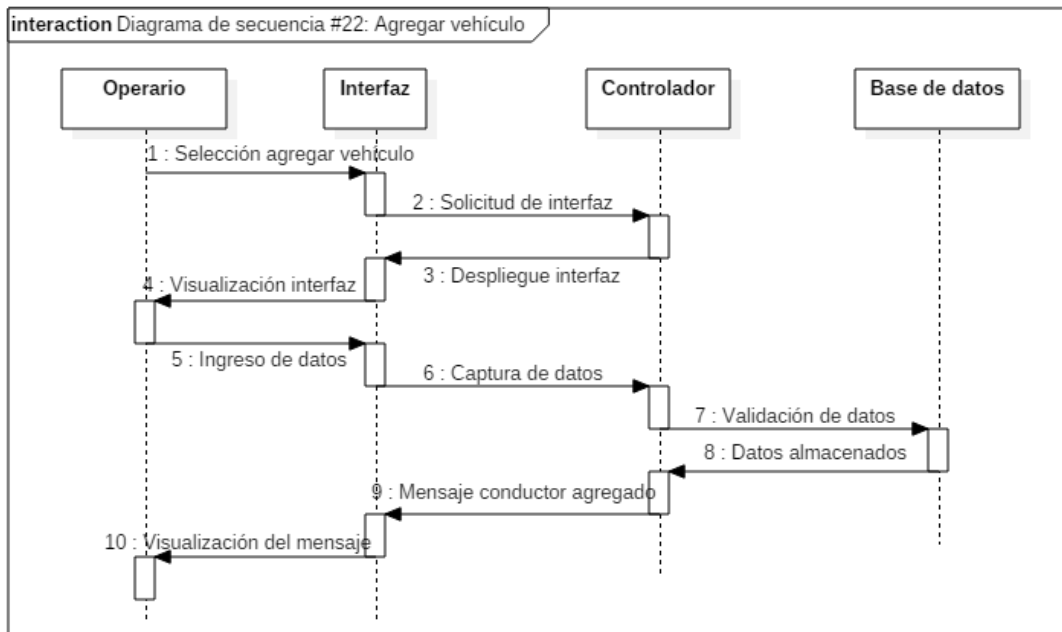


DIAGRAMA DE SECUENCIA #22: AGREGAR VEHÍCULO

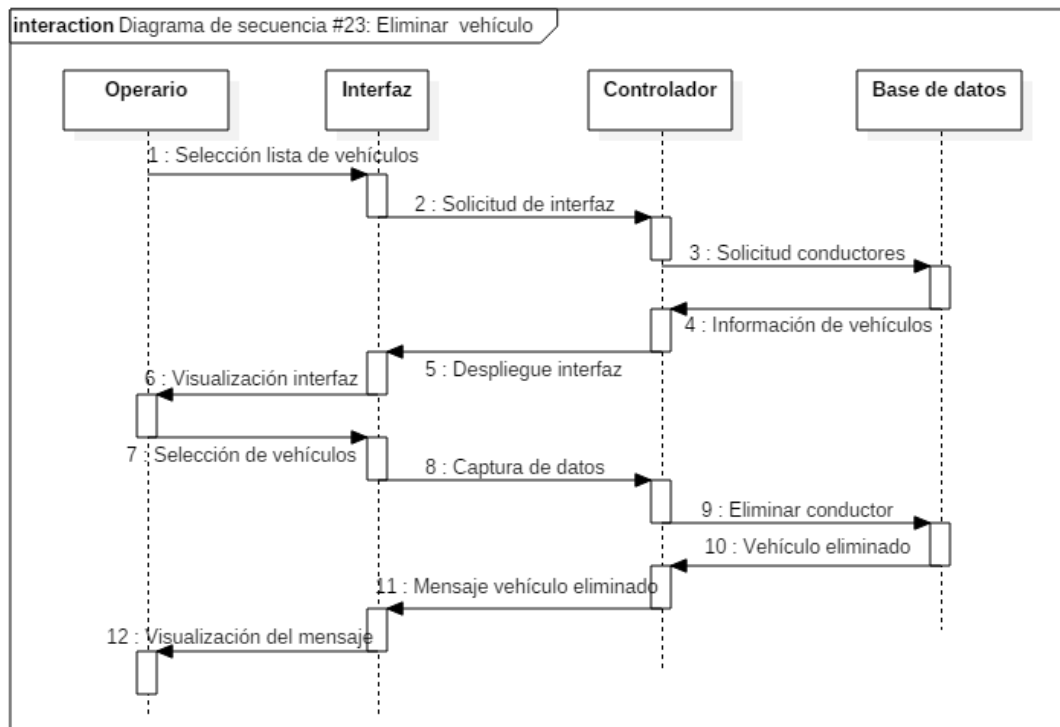


DIAGRAMA DE SECUENCIA #23: ELIMINAR VEHÍCULO

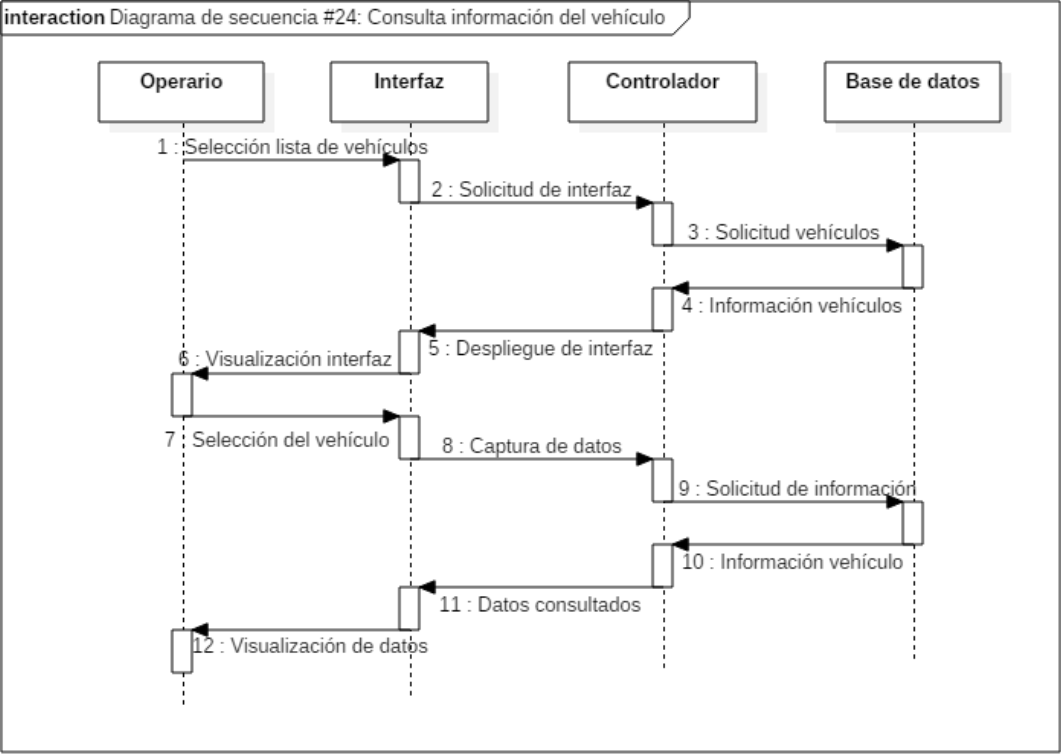


DIAGRAMA DE SECUENCIA #24: CONSULTA INFORMACIÓN DE VEHÍCULO

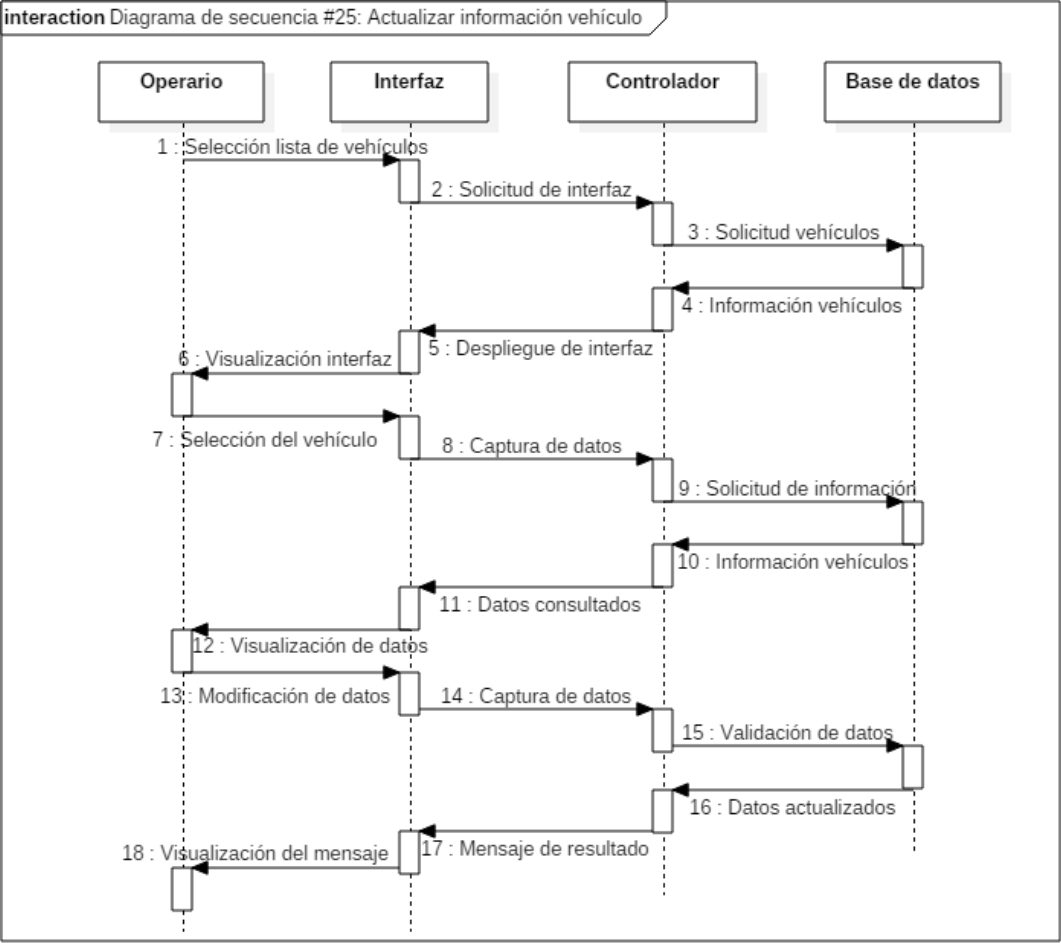


DIAGRAMA DE SECUENCIA #25: ACTUALIZAR INFORMACIÓN VEHÍCULO

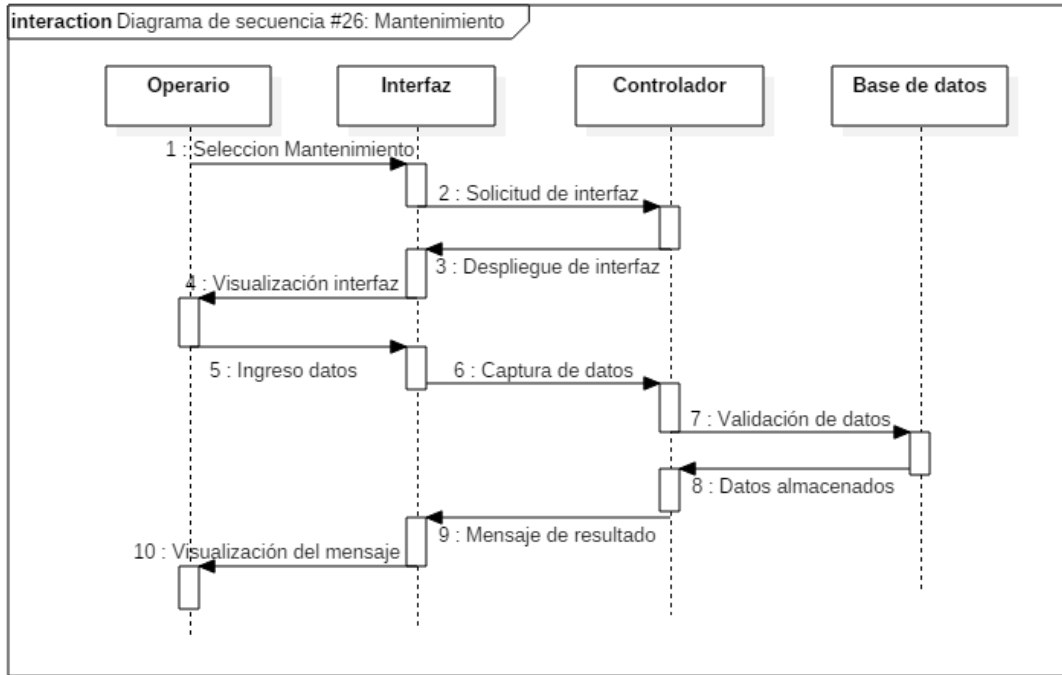


DIAGRAMA DE SECUENCIA #26: MANTENIMIENTO

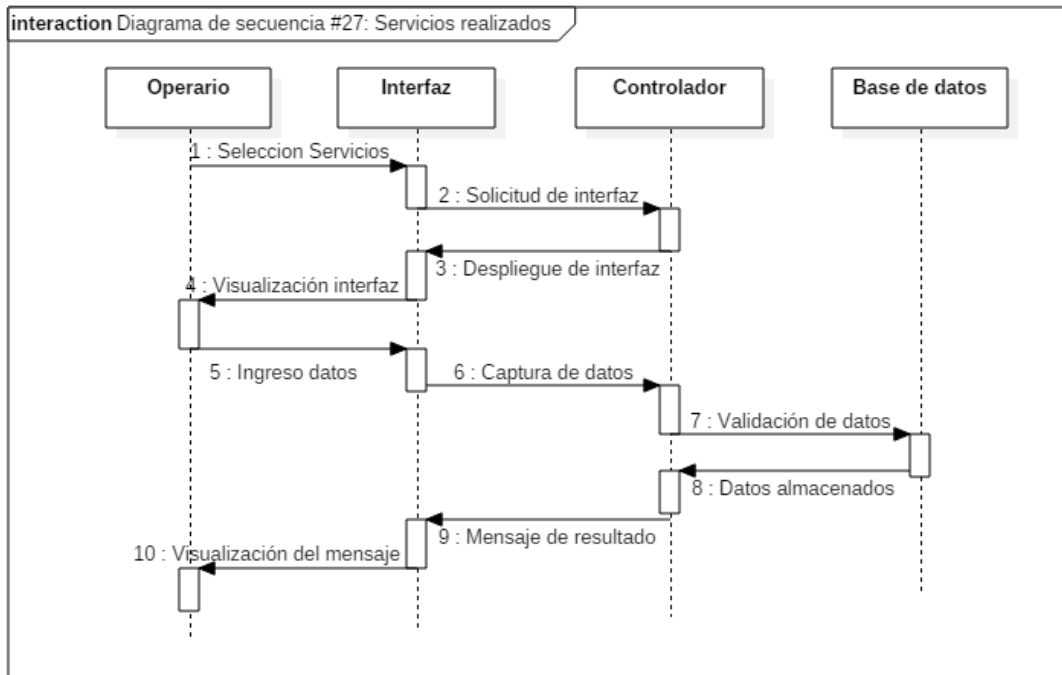


DIAGRAMA DE SECUENCIA #27: SERVICIOS REALIZADOS

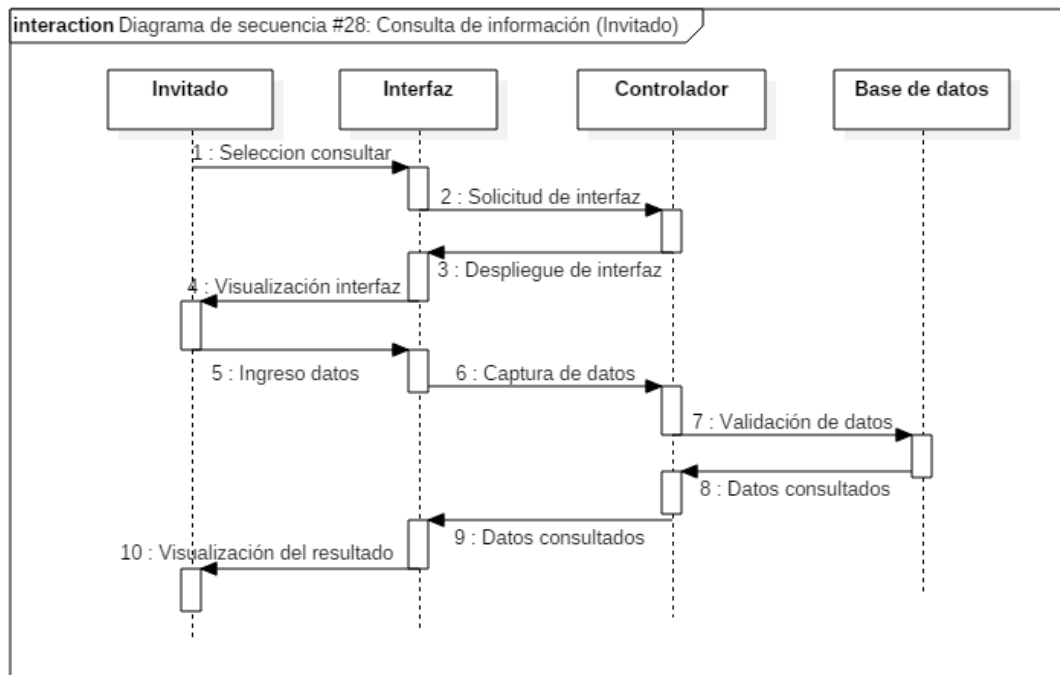


DIAGRAMA DE SECUENCIA #28: CONSULTA INFORMACIÓN (INVITADO)

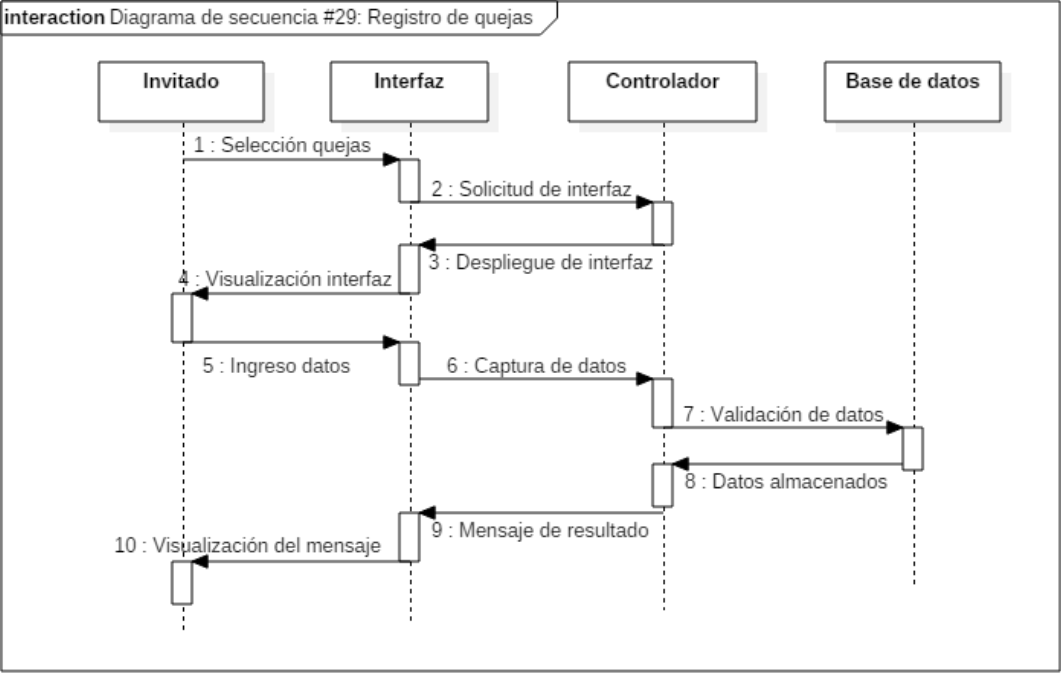
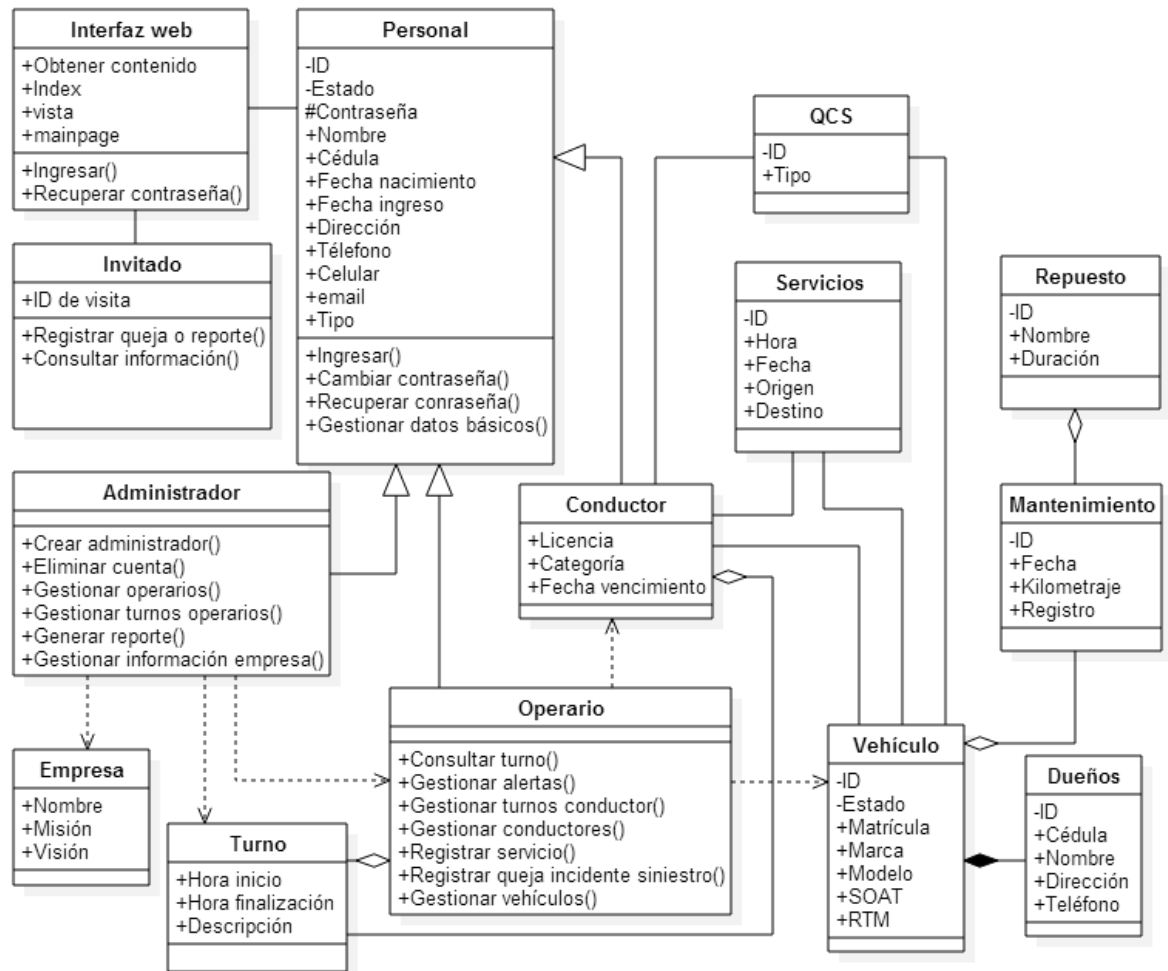
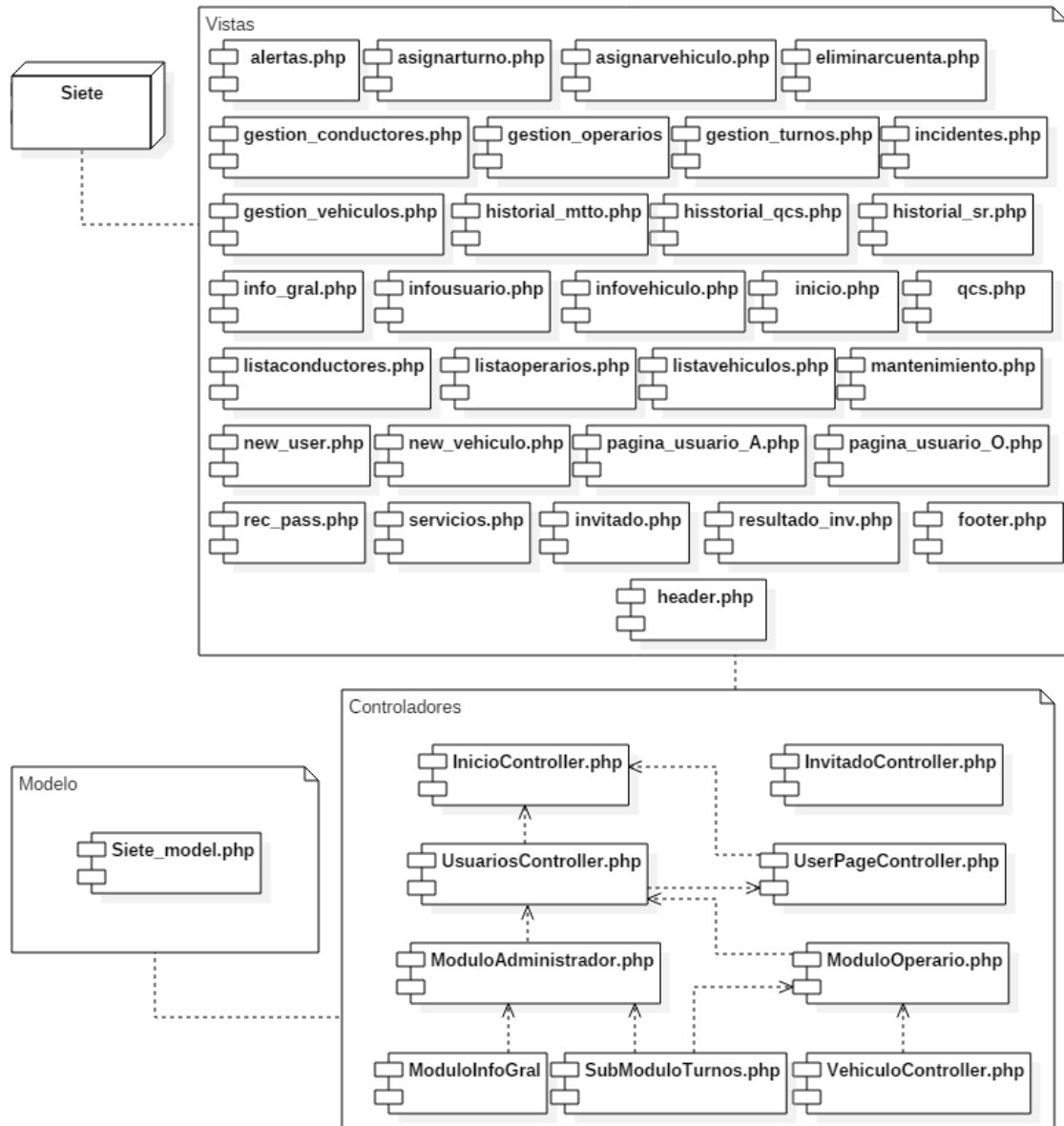


DIAGRAMA DE SECUENCIA #29: REGISTRO QUEJAS

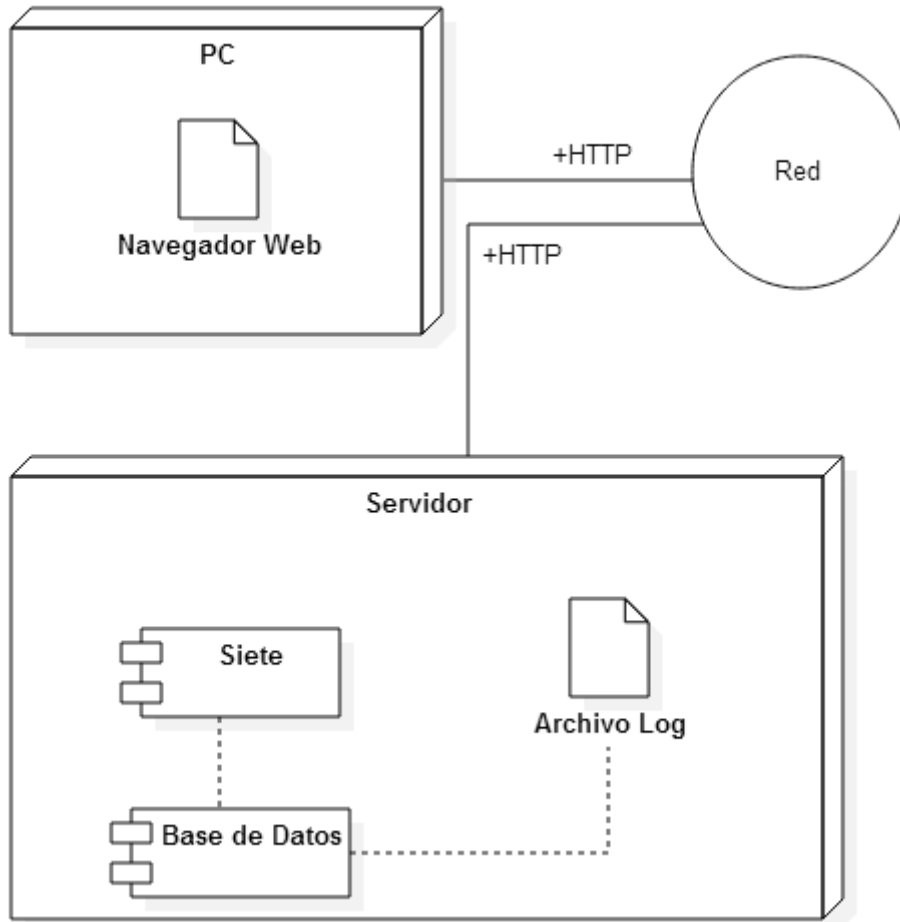
8.4 VISTA LOGICA (DIAGRAMA DE CLASES)



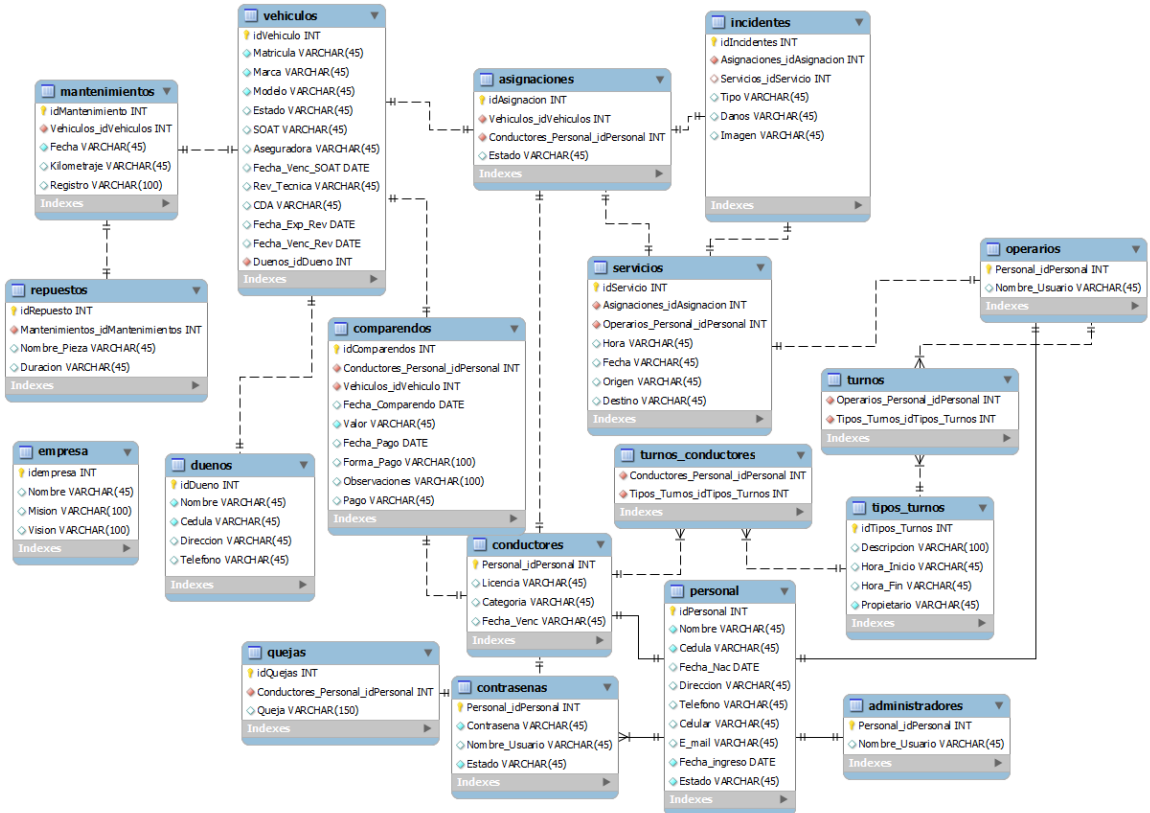
8.5 VISTA DE DESARROLLO (DIAGRAMA DE COMPONENTES)



8.6 VISTA FISICA (DIAGRAMA DE DESPLIEGUE)



8.7 MODELO RELACIONAL BASE DE DATOS



9. CONCLUSIONES

Realizar un correcto levantamiento de requerimientos, asegura el éxito durante el proceso de desarrollo de los productos, ya que se asegura las funcionalidades de manera específica que debe contener el sistema, los tipos de usuarios y sus privilegios, además brinda una vista inicial de que tan complejo y largo podría ser el ciclo de vida.

El análisis temprano de la arquitectura ayuda a reducir las probabilidades de fallo que pueden tener como consecuencia fracasos en los proyectos.

Las metodologías ágiles son útiles para el desarrollo rápido de software, teniendo ventajas enormes como entregar prototipos funcionales en poco tiempo, los cuales atraviesan el ciclo de vida y se encuentran debidamente documentados, esto ayudando a asegurar la calidad del producto.

La utilización de herramientas que facilitan las tareas de codificación como es el caso de los framework, aportan ventajas de tiempo en el desarrollo puesto que el objetivo principal de estos es no solo brindar una manera organizada de codificar a través de patrones (por ejemplo MVC), si no servir como estructura de trabajo y plantilla de codificación.

Cuando se realiza un prototipo es necesario tener en cuenta que este sea escalable, debido a su eventual parametrización evolución y producción.

El diseño de software ofrece una vista global de lo que es el sistema y sus escenarios de uso, mostrándolo de manera lógica, física, dinámica y estructural.

La realización de pruebas de caja negra (pruebas funcionales), dentro de cada iteración o parte funcional del prototipo, garantizan el cumplimiento de los requerimientos iniciales.

La elaboración de manuales técnicos y de usuarios, debe ser precisa, clara y acorde con el perfil del lector.

El correcto desarrollo de un producto de software debe atravesar y completar todas las etapas del ciclo de vida: análisis, diseño, codificación y pruebas.

10.RECOMENDACIONES

- Ante una eventual implantación del sistema de información es necesario conocer en un nivel de detalle alto aspectos del funcionamiento operativo de la empresa.
- El sistema de información siete podría complementarse con aplicaciones móviles para usuario que ofrezcan servicios de contacto de vehículos y geolocalización.
- El uso de cualquier sistema de información debe estar sujeto a una adecuada política de seguridad de la información, para asegurar la integridad, disponibilidad y confidencialidad de los datos, tal como es recomendado y evaluado en la norma ISO/IEC 27000- series sobre seguridad de la información.

11. BIBLIOGRAFÍA

ALVAREZ, J. Ingeniería de software. Técnicas de, 2008.

APARICIO, Alexandra. Ingeniería de Software. EN: Datateca, Universidad Nacional Abierta y a Distancia[en línea], 2012.

ESPINOSA, Adrian Trueba, et al. Automatización de la codificación del patrón modelo vista controlador (MVC) en proyectos orientados a la Web. CIENCIA ergo-sum, 2012, vol. 19.

FALGUERAS, Benet Campderrich. Ingeniería del software. Editorial UOC, 2002.

FIGUEROA, R. G., SOLÍS, C. J., & CABRERA, A. A. Metodologías tradicionales vs. Metodologías ágiles. *Universidad Técnica Particular de Loja, Escuela de Ciencias en Computación*. [En línea], Disponible en: <http://adonisnet.files.wordpress.com/2008/06/articulo-metodologia-de-sw-formato.doc>.

GARCÍA PEÑALVO, Francisco José; BRAVO MARTÍN, Sergio; CONDE GONZÁLEZ, Miguel Ángel. Ingeniería del Software. 2008.

GONZÁLEZ, Yanette Díaz; ROMERO, Yenisleidy Fernández. Patrón Modelo-Vista-Controlador. *Revista Telemática*, 2012, vol. 11, no 1.

HUMPHREY, Watts S. A discipline for software engineering. Addison-Wesley Longman Publishing Co., Inc., 1995.

JACOBSON, Ivar; BOOCH, Grady; RUMBAUGH, James. El proceso unificado de desarrollo de software. Reading: Addison Wesley, 2000.

KNIBERG, Henrik. Scrum and XP from the Trenches: How we do Scrum. Lulu.com, 2007.

KRUCHTEN, Philippe. Planos Arquitectónicos: El Modelo de 4+ 1 Vistas de la Arquitectura del Software. *IEEE Software*, 1995, vol. 12.

LARMAN, Craig. UML y Patrones. Pearson, 1999.

PEÑALVO, Francisco José García; GONZÁLEZ, Miguel Ángel Conde; MARTÍN, Sergio Bravo. Ingeniería del software.

PRESSMAN, Roger S. Ingeniería del Software: Un enfoque práctico. Mikel Angoar, 1997.

SCHWABER, Ken. Scrum development process. En *Business Object Design and Implementation*. Springer London, 1997.

SOMMERVILLE, Ian; GALIPIENSO, María Isabel Alfonso. Ingeniería del software. Pearson Educación, 2005.

[Consultado el 10 julio de 2015]:< http://www.codeigniter.com/user_guide/>

[Consultado el 13 julio de 2015] disponible en:< www.dsdm.org>

[Consultado el 15 julio de 2015] disponible en: <www.adaptivesd.com>

[Consultado el 15 julio de 2015] disponible en: <<http://alistair.cockburn.us/crystal/crystal.html>>

[Consultado el 15 julio de 2015] disponible en: <www.extremeprogramming.org/rules.html>

[Consultado el 18 julio de 2015] Documentación de java script disponible en: <<https://www.javascript.com/>>

[Consultado el 15 agosto de 2015] disponible en: <<http://normasicontec.org/>>

[Consultado el 15 agosto de 2015] disponible en: disponible en: <<http://www.w3.org/>>

[Consultado el 18 agosto de 2015] disponible en: <<http://www.it-mentor.com.ar/>>

[Consultado el 20 agosto de 2015] disponible en: <http://www.literateprogramming.com/quotes_ad.html>

ANEXO 1
PRODUCT BACKLOG

product backlog sistema de información para empresa de taxis (SIETE)
SPRINT 1
Administración de perfiles
el sistema debe permitir el ingreso de tres diferentes tipos de usuarios: administrador: operario: invitado: nota: el invitado solo podrá realizar consultas básicas relacionadas con los vehículos y conductores. <ul style="list-style-type: none">• historias de usuario numero 2
ingreso al sistema Los usuarios administrador y operario ingresan al sistema con un ID y una contraseña. <ul style="list-style-type: none">• historia de usuario numero 1
recuperación de contraseña y cambio de contraseña <ul style="list-style-type: none">• historia de usuario numero 3• historia de usuario numero 4
creación de usuarios <ul style="list-style-type: none">• historia de usuario numero 5• historia de usuario numero 6
módulo administrador: <ul style="list-style-type: none">• historia de usuario numero 7• historia de usuario numero 8• historia de usuario numero 9• historia de usuario numero 10• historia de usuario numero 11

<p>módulo gestión de operarios:</p> <ul style="list-style-type: none"> • historia de usuario numero 15 • historia de usuario numero 16 • historia de usuario numero 17
<p>SPRINT 2</p>
<p>módulo para operario</p> <ul style="list-style-type: none"> • historia de usuario numero 12 • historia de usuario numero 13
<p>módulo gestión conductores</p> <ul style="list-style-type: none"> • historia de usuario numero 18 • historia de usuario numero 19 • historia de usuario numero 20 • historia de usuario numero 21
<p>SPRINT 3</p>
<p>Módulo de gestión de vehículos</p> <ul style="list-style-type: none"> • historia de usuario numero 22 • historia de usuario numero 23 • historia de usuario numero 24 • historia de usuario numero 25
<p>módulo ingresos</p> <ul style="list-style-type: none"> • historia de usuario numero 26
<p>módulo servicios realizados</p> <ul style="list-style-type: none"> • historia de usuario numero 27
<p>vista como usuario invitado</p> <ul style="list-style-type: none"> • historia de usuario numero 14

ANEXO 2

PLAN DE PRUEBAS

INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
SISTEMA DE INFORMACIÓN EMPRESA DE TAXIS ESPECIALIZADO (SIETE)
PLAN DE PRUEBAS
APLICATIVO SIETE

Fecha	24/08/2015
Responsable	EQUIPO DE TRABAJO
Editor	
Versión	1

NOMBRE	SISTEMA DE INFORMACIÓN PARA EMPRESA DE TAXIS (SIETE)
INTRODUCCIÓN	El sistema de información siete se desarrolla bajo una metodología ágil, con el marco de trabajo SCRUM, el cual divide las entregas del producto final por sprint, las funcionalidades que deben ser evaluadas y probadas para cada sprint se encuentran definidas en el documento product backlog, el cual es uno de los artefactos de dicho marco de trabajo.
ALCANCE	Teniendo en cuenta los requerimientos descritos en el product backlog para cada sprint, se evaluaron a través de diferentes casos de prueba, la funcionalidad y operabilidad, usabilidad de los módulos descritos para el sistema SIETE, los cuales son: módulo administrador, módulo gestión operarios, módulo operario, módulo gestión de conductores, módulo gestión de vehículos, módulo ingresos, módulo servicios realizados, además de la vista como usuario invitado, la creación e ingreso de usuarios al sistema, los privilegios y roles de cada uno.
ELEMENTOS	Los documentos que se necesitaran para la realización de las pruebas son: product backlog con funcionalidades requeridas para el sprint listado de requerimientos
ESTRATEGIA	Se utilizara la estrategia de caja negra en donde se evaluaran los requerimientos funcionales que el software deberá cumplir para brindar el funcionamiento deseado

CARACTERÍSTICAS	Se evaluarán todas las características funcionales del software que se describieron en las historias de usuario; además de esto se evaluarán todos los casos alternos de los casos de uso ya que se considera que tener una falla en estas secciones del programa podría llevar a inconsistencias en la información.
CATEGORIA DE LA CONFIGURACIÓN	La prueba deberá ser suspendida cuando se presente fallas de componentes cuya funcionalidad dependan otras funciones como es el caso de: Fallar al momento de autenticarse, no poder crear un usuario. Otra causa de suspensión de las pruebas es cuando el servidor no se encuentre en funcionamiento o no exista conexión con la base de datos.
ENTREGABLES	Los documentos que se entregaran son: Plan de pruebas, casos de prueba, reporte de pruebas
RECURSOS	PC con acceso a internet acceso a documentación del producto
RIESGOS	<ol style="list-style-type: none"> 1) El desarrollador del proyecto no cumple con la fecha de entrega del código. 2) El servidor se encuentra caído al momento de realizar las pruebas. 3) No se cumplen con más de 4 pruebas de funcionalidad general del software. 4) Los tiempos de carga de el aplicativo son demasiado lentos

ANEXO 3

casos de prueba sprint 1

CASOS DE PRUEBA

APLICATIVO SIETE

Fecha 24/08/2015

Responsable Paola Andrea Agudelo -Denier Adolfo
Alcalde

Editor

Versión 1

TIPO	ID CASO DE PRUEBA	H U	DESCRIPCIÓN GENERAL	DESCRIPCIÓN PASO A PASO	ENTRADA	SALIDA ESPERADA	SE DEBE CUMPLIR		SALIDA OBTENIDA	OBSERVACIONES
							SI	NO		
FUNCIONALIDAD GENERAL	pf1	1	Ingreso a la página	1)se ingresa a la página: 2)se verifica campo ingreso usuarios registrados 3) se verifica campo	URL	página de inicio del aplicativo con los campos requeridos	x		Los campos requeridos se encuentran en la pagina de inicio	

				ingreso usuario invitado						
	pf2	2	tipos de perfiles	1) Se ingresa a la página: 2)se verifica que existen tres tipos de perfiles	Datos	tipos de perfiles: administrador operario invitado	x		A través del perfil administrador se verifica que existe el perfil operario	El perfil invitado no se encuentra funcional para este sprint
	pf3	3	Recuperación de contraseña	1) Se ingresa a la página: 2) se ingresa el ID de usuario 3)se selecciona opción recuperar contraseña 4) Se visualiza un mensaje que indica que el link de recuperación fue enviado al correo. 5) Se espera	Datos	mensaje de confirmación envío del enlace de recuperación .	x		La recuperación de contraseña no pudo ser realizada.	

				la respuesta del sistema.						
	pf4	4	Cambio de contraseña	1) Se ingresa a la página: 2) se ingresa el ID de usuario y la contraseña 3) Dentro del perfil de usuario se selecciona opción cambio contraseña 4) Se ingresa la contraseña actual 5) se ingresa la nueva contraseña 6) Se espera la respuesta	Datos	mensaje confirmación cambio de contraseña exitoso	x		El cambio de contraseña pudo ser realizado con éxito.	La contraseña exige seguridad de caracteres alfanumericos de acuerdo con estandares de seguridad o con proyectos para seguridad de proyectos como OWASP.

				del sistema.						
	pf5	5	Verificación de usuario administrador inicial	1) Se ingresa a la página: 2) se ingresa el ID de Administrador y la contraseña proporcionados a el primer administrador 3) Se espera la respuesta del sistema.	Datos	visualización perfil administrador inicial	x		Al realizar el primer ingreso al aplicativo, este arroja un mensaje informando la contraseña y administrador por defecto, al igual que el mensaje que recuerda actualizar los datos al ingresar.	

	pf6	5	Creación del usuario administrador	1) Se ingresa a la página: 2) Se ingresa con el ID y contraseña de administrador 3) se elige la opción crear nuevo administrador 4) se ingresan los datos del nuevo administrador 5) Se espera la respuesta del sistema con la creación exitosa del usuario administrador	Datos	Mensaje de creación exitosa de la cuenta.	x		el aplicativo despliega la interfaz para diligenciar los datos del nuevo administrador. Proporciona una contraseña inicial para el nuevo usuario	al realizar el ingreso con estos datos generados para el nuevo administrador el ingreso al perfil se realizó con éxito
--	-----	---	------------------------------------	---	-------	---	---	--	--	--

	pf7	6	creación del usuario operario	1) Se ingresa a la página: 2) Se ingresa con el ID y contraseña de administrador 3) se elige el modulo gestión de operarios 4) se elige la opción crear nuevo operario 5) se ingresan los datos del nuevo operario 6) Se espera la respuesta del sistema con la creación exitosa del nuevo operario	Datos	visualización del mensaje de nuevo operario creado su ID de usuario y contraseña temporal	x		el aplicativo despliega la interfaz para diligenciar los datos del nuevo operario. Proporciona una contraseña inicial para el nuevo usuario	No fue posible ingresar al perfil de este usuario creado, mensaje de error en conexión con base de datos.
	pf78	7	perfil administrador	1) Ingresar al sistema como el usuario administrador.	Datos	visualización del campo requerido en el perfil	x		El campo se encuentra disponible dentro del perfil	

				2) visualización de campos para datos basicos 3) respuesta del sistema						
	pf9	7	perfil administrador	1) Ingresar al sistema como el usuario administrador. 2) visualización de campos para consultas 4) respuesta del sistema	Datos	Visualización de campo consultas	x		se visualiza el campo información general	
	pf10	7	perfil administrador (acceso al modulo gestión de operarios)	1) Ingresar al sistema como el usuario administrador. 2) acceso al modulo operarios 3) se visualizan los campos : crear y eliminar	Datos	Visualización de los campos requeridos dentro del submodulo	x		se accede al modulo gestión de operarios, donde se encuentran las opciones, nuevo operario y lista de operarios.	

				operario, consultar datos de operarios ,asignar turnos a operarios 3) se espera respuesta del sistema						
	pf11	7	perfil administra dor	1) Ingresar al sistema como el usuario administrad or. 2) acceso al módulo operarios 3) se elige la opción crear, eliminar, consultar operario 4)se ingresan los datos del operario a crear, eliminar , consultar 5) la acción	Datos	Visualización de la acción realizada dentro del submódulo de manera exitosa.	x		se elige la opción lista de operarios, el sistema permite eliminar y consultar operarios	

				escogida se realiza						
	pf12	7	perfil administrador	<p>1) Ingresar al sistema como el usuario administrador.</p> <p>2) acceso al módulo operarios</p> <p>3) se elige la opción asignar turnos</p> <p>4) se ingresan los datos del nuevo turno</p> <p>5) la acción escogida se realiza</p>	Datos	Visualización de la acción realizada dentro del submodulo de manera exitosa.	x		Se crea el nuevo turno, y posteriormente es asignado al usuario.	el campo aparece como un campo de texto donde se realiza la descripción del turno.

	pf13	7	perfil administrador	1) Ingresar al sistema como el usuario administrador. 2) acceso al submodulo información de la empresa 3) la acción escogida se realiza	Datos	Acceso al perfil empresa y a sus funcionalidades.	x		se visualiza información en el sistema	no se visualiza información de la empresa.
	pf14	7	perfil administrador	1) Ingresar al sistema como el usuario administrador. 2) opción eliminar usuarios 3) se elige el usuario a eliminar 4) la acción escogida se realiza	Datos	Confirmación de modificación de los datos.	x		el sistema solicita la contraseña para eliminación del usuario.	no fue posible la eliminación de un usuario operario, debido a que la contraseña suministrada no fue aceptada en el sistema-

	pf15	7	perfil administrador	1) Ingresar al sistema como el usuario administrador. 2) opción eliminar cuenta 3) la cuenta es eliminada de manera lógica	Datos	Mensaje de confirmación de eliminación de la cuenta	x		mensaje de eliminación de la cuenta de manera correcta.	
	pf16	7	perfil administrador	1) Ingresar al sistema como el usuario administrador. 2) opción eliminar cuenta 3) ingresar datos de otro administrador 4) se espera la respuesta negativa del sistema	Datos	Mensaje de error por acción invalida	x		mensaje de error del servidor	no es posible eliminar cuentas de otros usuarios(la contraseña no es aceptada)

	pf17	8	Gestión de datos básicos	1) Ingresar al sistema como un usuario registrado 2) opción datos básicos 3) gestión de información de correo y fecha de nacimiento 4) respuesta del sistema	Datos	Mensaje de confirmación de realización de gestión de los datos	x		mensaje de error en conexión con base de datos	No es posible ingresar al perfil de usuarios registrados con la contraseña suministrada por el sistema.
	pf18	9	submódulo de información general	1) Ingresar al sistema como el usuario administrador. 2) acceso submódulo de información general 3) generar informes (4 tipos) 4) se espera la respuesta del sistema	Datos	Informe (s) solicitados por el usuario administrador	x		mensaje con la posibilidad de generar un reporte(número de usuario dentro del sistema)	el reporte se presenta en formato pdf con la posibilidad de ser impreso.

	pf19	11	único administrador	1) Ingresar al sistema como el usuario administrador. 2) elegir opción eliminar cuenta 3) respuesta del sistema	Datos	Mensaje de error por acción invalida ya que el sistema posee un único administrador	x		mensaje de advertencia sobre único administrador y no opción de eliminación de cuenta por este motivo	
PRUEBAS DE SEGURIDAD	ps1	1	Ingreso a una cuenta con contraseña incorrecta	1) Se ingresa a la página: 2) Se ingresa los datos de un usuario registrado y una contraseña incorrecta.	Datos	Generar un mensaje de error por parte del sistema.	x		Mensaje de no conexión con la base de datos.	
	ps2	1	Ingresar a una cuenta sin digitar una contraseña	1) Se ingresa a la página: 2) Se ingresan ID de un usuario registrado pero no una contraseña.	Datos	Generar un mensaje de error por parte del sistema.	x		Mensaje de campo obligatorio	

	ps3	3	Ingreso no autorizado mediante links	1) Acceder al sistema como cualquier usuario. 2) Acceder a otro tipo de usuario mediante los links	Datos	Generar un mensaje de error por parte del sistema.	x		Mensaje de error al encontrar la pagina	
	ps4	3	Ingreso de cuenta no existente	1) Se ingresa a la página: 2) Se ingresa un ID de usuario que no se encuentre registrado.	Datos	Generar un mensaje de error por parte del sistema.	x		Mensaje de no conexión con la base de datos.	
	ps5	5	Modificación de información privada	1) Se ingresa como administrador al sistema. 2) Se selecciona un usuario 3) se modifican datos	Datos	El sistema no debe habilitar los campos para la modificación .	x		no se activa la opción de modificación sobre los datos del operario	

ANEXO 4
REPORTE DE PRUEBAS SPRINT 1
APLICATIVO SIETE
VERSIÓN 1

SEGUIMIENTO FALLAS ENCONTRADAS SPRINT 1

N°	ID CP	Falla o Error encontrado	Estado Final
1	pf3	no se realizó la recuperación de contraseña	Las pruebas se han realizado sobre un servidor local, quedando esta funcionalidad como pendiente para pruebas una vez el sistema se encuentre implementado en un servidor web.
2	pf7	El usuario operario fue creado con éxito y aparece en el listado de operarios, sin embargo no es posible acceder con los datos suministrados al perfil.	El error fue corregido y ahora es posible acceder al perfil con los datos suministrados inicialmente.
3	pf14	no es posible eliminar usuario operario debido a errores en el servidor cuando se ingresa la contraseña	el error fue corregido inicializando variables en la bd y ahora es posible eliminar la cuenta de un operario, digitando la contraseña de administrador.
4	pf16	no es posible eliminar cuentas de otros usuarios(la contraseña no es aceptada)	el error fue corregido y la contraseña es aceptada, permitiendo eliminar cuentas de usuarios.

5	pf1	todos los ingresos con datos erróneos arrojan mensajes de conexión con base de datos y no mensajes de datos inválidos.	El error fue corregido, el sistema arroja mensajes de campo obligatorio, o datos inválidos para los datos inválidos ingresados.
---	-----	--	---

INDICADORES TOMADOS	
Numero de fallas encontradas en la aplicación	5
Numero de errores encontrados en la documentación	0
Tiempo tomado por los desarrolladores para solucionar una falla (horas)	2

ANEXO 5

casos de prueba sprint 2

CASOS DE PRUEBA

APLICATIVO SIETE

Fecha 05/09/2015

Responsable Paola Andrea Agudelo- Denier Adolfo Alcalde

Editor _____

Versión 1

TIPO	ID CASO DE RPUEBA	HU	DESCRIPCIÓN GENERAL	DESCRIPCIÓN PASO A PASO	ENTRADA	SALIDA ESPERADA	SE DEBE CUMPLIR		SALIDA OBTENIDA	OBSERVACIONES
							SI	NO		
funcionalidad	pf1	12	usuario operario	1)se ingresa a la página: 2)se ingresan datos de ID y operario 3) se accede a datos básicos 4) se realiza gestión sobre datos básicos 5) se espera respuesta del sistema	Datos	Mensaje de gestión de datos básicos realizados.	x		Visualización de gestión de datos realizado con éxito.	Los datos susceptibles a cambios son datos de contacto.
	pf2	13	usuario operario	1)se ingresa a la página: 2)se ingresan datos de ID y operario 3) se accede a revisión de alertas 4) se revisan las alertas pendientes y sus características 5) opción desactivar la alerta 6) se espera respuesta del sistema	Datos	listado de alertas pendientes en el sistema	x		Visualización en la interfaz de inicio de alertas, se despliega información detallada de cada alerta.	No se encuentra disponible la opción para desactivar la alerta.

pf3	12	usuario operario	<p>1)se ingresa a la página: 2)se ingresan datos de ID y operario 3) se accede a realizar una consulta sobre un conductor o vehiculo 4) se ingresan los datos a ser consultados 5) se espera respuesta del sistema con los datos de la consulta realizada</p>	Datos	visualización de datos consultados	x		visualización datos conductor	No se encuentra disponible una opción para búsqueda de información, el sistema despliega toda la información que tiene disponible sobre el usuario elegido.
pf4	12	usuario operario(acceso al módulo gestión de operarios)	<p>1)se ingresa a la página: 2)se ingresan datos de ID y operario 3) se accede al módulo gestión de operarios 4) se visualizan los campos: consultas turnos 5) se espera respuesta del sistema</p>	Datos	visualización de los campos dentro del módulo.	x		visualización del campo requerido	
pf5	12	usuario operario(acceso al módulo gestión de operarios)	<p>1)se ingresa a la página: 2)se ingresan datos de ID y operario 3) se accede a modulo gestión de operarios 4) se selecciona consultar turnos 5) se visualiza los turnos asignados a el operario</p>	Datos	visualización de los turnos asignados por el administrador	x		visualización de turnos asignados al operario junto a una descripción del mismo	
pf6	18	usuario operario(acceso al módulo gestión de conductores)	<p>1)se ingresa a la página: 2)se ingresan datos de ID y operario 3) se accede a modulo gestión de conductores 4) se selecciona gestión de turnos 5) se ingresan los datos del nuevo turno 6) se espera respuesta del sistema</p>	Datos	visualización de turnos creados y la opción para adicionar uno nuevo	x		Visualización de turnos existentes la posibilidad de adicionar, eliminar y modificar turnos.	

pf7	18	usuario operario(acceso al módulo gestión de conductores)	1)se ingresa a la página: 2)se ingresan datos de ID y operario 3) se accede a modulo gestión de conductores 4) se selecciona adicionar nuevo conductor 5) se ingresan los datos del nuevo conductor 6) se espera respuesta del sistema	Datos	visualización de mensaje de nuevo conductor adicionado	x		visualización mensaje de creación exitosa del conductor	
pf8	19	usuario operario(acceso al módulo gestión de conductores, submódulo lista de conductores)	1)se ingresa a la página: 2)se ingresan datos de ID y operario 3) se accede a modulo gestión de conductores 4) se accede al submódulo lista de conductores 5) se selecciona una opción: eliminar consultar asignar turno 6) se realiza la gestión sobre la totalidad de las opciones disponibles 7) se espera la respuesta del sistema		visualización de gestión en las opciones disponibles	x		las opciones dentro del submodulo se encuentran operativas	
pf9	20	usuario operario(acceso al módulo gestión de conductores, submódulo comparendos, siniestros y quejas)	1)se ingresa a la página: 2)se ingresan datos de ID y operario 3) se accede a modulo gestión de conductores 4) se accede al submódulo comparendos,siniestros, quejas 5) se selecciona una opción: comparendos siniestros quejas estadísticas generales 6) se realiza la gestión sobre la totalidad de las opciones disponibles 7) se espera la respuesta del sistema		visualización del módulo	x			El módulo no se encuentra disponible en el sistema

pf10	20	usuario operario(acceso al módulo gestión de conductores, submódulo comparendos, siniestros y quejas)	1)se ingresa a la página: 2)se ingresan datos de ID y operario 3) se accede a modulo gestión de conductores 4) se accede al submódulo comparendos, siniestros, quejas 5) se selecciona una opción: siniestros 6) se habilita la opción para cargar imágenes 7) se espera la respuesta del sistema		carga correcta de la imagen al sistema	x			El módulo no se encuentra disponible en el sistema
pf11	12	usuario operario(acceso al módulo gestión de vehiculos)	1)se ingresa a la página: 2)se ingresan datos de ID y operario 3) se visualiza el modulo gestión de vehiculos 4) se espera la respuesta del sistema	Datos	visualización del módulo	x			la visualización del módulo no se encuentra disponible en el sistema
pf12	18	usuario operario(acceso al módulo gestión de conductores)	1)se ingresa a la página: 2)se ingresan datos de ID y operario 3) se accede a modulo gestión de conductores 4) se selecciona eliminar conductor 5) se espera respuesta del sistema	Datos	visualización de mensaje de conductor eliminado	x		visualización mensaje de eliminación correcta de la cuenta del conductor	
pf13	12	usuario operario (acceso a módulo servicios prestados, ingresos)	1)se ingresa a la página: 2)se ingresan datos de ID y operario 3) se accede a los módulos servicios realizados e ingresos 4) se espera respuesta del sistema		visualización de los módulos	x			los módulos no se encuentran disponibles

pruebas de seguridad	ps1	12	Ingresar de un documento ID demasiado corto en la gestión de datos	1) Se ingresa a la página: 2) Se ingresan los datos de contacto 3) se ingresa un numero de documento corto	Datos	Generar un mensaje de error por parte del sistema.	x		visualización mensaje "cédula muy corta"	
	ps2	12	Ingresar un correo electronico sin formato @	1) Se ingresa a la página: 2) Se ingresan los datos de gestión de usuarios 3)se ingresa un correo con un formato incorrecto	Datos	Generar un mensaje de error por parte del sistema.	x		visualización mensaje " error en correo electrónico"	
	ps3	12	Dejar campos sin información en la gestión de datos basicos	1) Se ingresa a la página: 2) Se ingresan los datos de gestión de usuarios 3)no se ingresan datos de contacto como dirección ,telefono 4) se espera respuesta del sistema	Datos	Generar un mensaje de error por parte del sistema.	x		El sistema permite dejar campos sin validar como la información de contacto	la información de contacto, puede ser actualizada pero en todos los casos debe encontrarse disponible en el aplicativo
	ps4	18	Ingreso de datos incorrectos en la licencia de conducción	1) Se ingresa a la página: 2) Se ingresa al módulo gestión de conductores. 3) se registra un nuevo conductor 4) se ingresan datos erroneos en la licencia	Datos	Generar un mensaje de error por parte del sistema.	x		visualización de mensaje de error en número de la licencia	es posible dejar campos sin registro.

ANEXO 6
REPORTE DE PRUEBAS SPRINT 2
APLICATIVO SIETE
VERSIÓN 1

INGENIERÍA DE SISTEMAS Y COMPUTACIÓN

REPORTE **2**
APLICATIVO

Fecha 22/09/2015

Proyecto SIETE

SEGUIMIENTO FALLAS ENCONTRADAS SPRINT 1

N°	ID CP	Falla o Error encontrado	Estado Final
1	pf2	No es posible desactivar las alertas	Se decidio dejar esto como un requerimiento futuro.
2	pf3	no es posible realizar una búsqueda para filtrar solo la información que deseo consultar del conductor, el sistema despliega la totalidad de información que alberga.	Se decidio dejarlo de esta manera ya que se, especifico que la información era muy poca para hacer un filtrado de datos.
3	pf9	no se encuentra disponible el módulo siniestros quejas y reclamos	El módulo se encuentra disponible en el sistema para el sprint 3.

4	pf10	no es posible cargar imágenes al aplicativo ya que este no tiene la opción disponible	Se decidio dejar este requerimiento dentro de desarrollos futuros.
5	pf11	no es posible ingresar al módulo gestión vehiculos	El módulo se encuentra disponible y es posible acceder a el.
6	pf11	no se encuentra disponible el módulo servicios prestados e ingresos	El módulo servicios se encuentra disponible, el módulo ingresos se decidio dejarlo para una eventual parametrización.
7	ps3	Es posible dejar datos sin diligenciar dentro de la información de contacto de un usuario	Está información no es obligatoria.
8	ps4	es posible dejar datos sin diligenciar en la información sobre la documentación del conductor	Está información no es obligatoria.

INDICADORES TOMADOS	
Numero de fallas encontradas en la aplicación	8
Numero de errores encontrados en la documentación	0
Tiempo tomado por los desarrolladores para solucionar una falla (horas)	

Anexo 7
casos de prueba sprint 3
CASOS DE PRUEBA
APLICATIVO SIETE

Fecha 26/09/2015

Responsable Paola Andrea Agudelo

Editor _____

Versión 2

TIPO	ID CASO DE RPUEBA	HU	DESCRIPCIÓN GENERAL	DESCRIPCIÓN PASO A PASO	ENTRADA	SALIDA ESPERADA	SE DEBE CUMPLIR		SALIDA OBTENIDA	OBSERVACIONES	
							SI	NO			
funcionalidad	pf1	18	usuario operario(acceso al módulo gestión de conductores)	1)se ingresa a la página: de ID y operario accede módulo gestión conductores opción actualizar datos del conductor 5) se ingresan los datos a actualizar respuesta del sistema	2)se ingresan datos 3) se 4) se elige 6) se espera	Datos	Mensaje de actualización de datos	x		Datos actualizados de manera exitosa	
	pf2	19	usuario operario(acceso al módulo gestión de conductores)	1)se ingresa a la página: de ID y operario accede módulo gestión conductores opción agregar documentos 5) se cargan un documento del sistema	2)se ingresan datos 3) se 4) se elige 6) se espera respuesta	Datos	Mensaje de carga exitosa del documento		x	La opción no se encuentra disponible en el sistema.	

pf3	19	usuario operario(acceso al módulo gestión de conductores)	1)se ingresa a la página: de ID y operario 2)se ingresan datos de ID y operario 3) se accede módulo gestión conductores, lista de conductores 4) se elige opción Asignar vehículo 5) el sistema despliega la lista de vehículos 6) se elige un vehículo de la lista 7) se espera respuesta del sistema	Datos	Mensaje de asignación exitosa del vehículo	x		Datos asignados de manera exitosa	
pf4	20	usuario operario(acceso al módulo gestión de conductores)	1)se ingresa a la página: de ID y operario 2)se ingresan datos de ID y operario 3) se accede módulo gestión conductores 4) se elige opción comparendos, quejas, siniestros 5) se elige la opción queja comparendo o siniestro y la acción registrar 6) se ingresan los datos 7) se espera respuesta del sistema	Datos	Visualización registro exitoso	x		El registro de la queja, el siniestro y el comparendo se realizo con éxito.	
pf5	20	usuario operario(acceso al módulo gestión de conductores)	1)se ingresa a la página: de ID y operario 2)se ingresan datos de ID y operario 3) se accede módulo gestión conductores 4) se elige opción comparendos, quejas, siniestros 5) se elige la opción queja comparendo o siniestro y la acción historial 6) se espera respuesta del sistema	Datos	Visualización de datos	x		La visualización del historial de QCS fue realizada con éxito.	
pf6	20	usuario operario(acceso al módulo gestión de conductores)	1)se ingresa a la página: de ID y operario 2)se ingresan datos de ID y operario 3) se accede módulo gestión conductores 4) se elige opción comparendos, quejas, siniestros 5) se elige la opción siniestro 6) se habilita la carga de imagenes 7) se carga una imagen 8) se espera respuesta del sistema	Datos	Visualización carga exitosa de imágenes	x		No se encuentra disponible la opción para cargar imágenes.	

pf7	20	usuario operario(acceso al módulo gestión de conductores)	1)se ingresa a la página: de ID y operario accede módulo gestión conductores visualiza opción para consulta de estadísticas 5) se espera respuesta del sistema	2)se ingresan datos 3) se 4) se	Datos	Visualización estadística	x		Las estadísticas disponibles en el sistema son para el apartado de quejas, comparendos y reclamos.
pf8	22	usuario operario(acceso al módulo gestión de vehículos)	1)se ingresa a la página: de ID y operario accede a modulo gestión de vehículos 4) se selecciona opción adicionar vehículo 5) se ingresan los datos del nuevo vehículo 6) se espera respuesta del sistema	2)se ingresan datos 3) se	Datos	Visualización de adición exitosa del vehículo	x		Visualización del mensaje de adicción exitosa.
pf9	22	usuario operario(acceso al módulo gestión de vehículos)	1)se ingresa a la página: de ID y operario accede a modulo gestión de vehículos 4) se selecciona opción eliminar vehículo 5) se ingresan los datos del vehículo 6) se espera respuesta del sistema	2)se ingresan datos 3) se	Datos	Visualización de eliminación exitosa del vehículo	x		La eliminación del vehículo se realiza.
pf10	23	usuario operario(acceso al módulo gestión de vehículos , submódulo datos del vehículo)	1)se ingresa a la página: de ID y operario accede a modulo gestión de vehículos se elige consultar vehículo campos datos basicos, conductor asignado, documentos del vehiculo, información básica sobre el dueño espera respuesta del sistema	2)se ingresan datos 3) se 4) 5) se visualizán	Datos	Visualización de consulta requerida	x		Se encuentran todos los datos dentro del módulo

pf11	24	usuario operario(acceso al módulo gestión de vehículos , submódulo mantenimiento)	1)se ingresa a la página: de ID y operario 2)se ingresan datos de ID y operario 3) se accede a modulo gestión de vehículos 4) se accede al submodulo mantenimiento 5) se visualizán los campos historial, registro de mantenimiento realizado 6) se espera respuesta del sistema	Datos	Visualización de datos consultados	x		Es posible adicionar el nuevo mantenimiento, se ingresan los datos de fecha, Kilometraje, registro y la opción para agregar un repuesto.
pf12	24	usuario operario(acceso al módulo gestión de vehículos , submódulo mantenimiento)	1)se ingresa a la página: de ID y operario 2)se ingresan datos de ID y operario 3) se accede a modulo gestión de vehículos 4) se accede al submodulo mantenimiento 5) se visualizán los campos: historial, registro de mantenimiento realizado 6) se espera respuesta del sistema	Datos	Visualización gestión correcta de datos	x		Visualización de historial de mantenimientos realizados con posibilidad de acceder a detalles de los mismos.
pf13	25	usuario operario(acceso al módulo gestión submódulo reportes)	1)se ingresa a la página: de ID y operario 2)se ingresan datos de ID y operario 3) se accede a modulo gestión de vehículos 4) se accede al submodulo reportes 5) se visualizán un formulario para escoger los datos requeridos dentro del reporte 6) se eligen datos a generar en el reporte se espera respuesta del sistema	Datos	Visualización de campos requeridos en el módulo	x		Reporte generado con éxito.

pf14	25	usuario operario(acceso al módulo gestión submódulo reportes)	1)se ingresa a la página: de ID y operario 2)se ingresan datos 3) se accede a modulo gestión de conductores 4) se accede al submodulo reportes 5) se visualizán un formulario para escoger los datos requeridos dentro del reporte 5) se eligen datos a generar en el reporte 6) se espera respuesta del sistema	Datos	Visualización de campos requeridos dentro del módulo	x		Reporte generado con éxito.	
pf15	25	usuario operario(acceso al módulo gestión submódulo reportes)	1)se ingresa a la página: de ID y operario 2)se ingresan datos 3) se accede a modulo gestión de vehículos 4) se accede al submodulo reportes 5) se visualizán un formulario para escoger los datos requeridos dentro del reporte 5) se eligen datos a generar en el reporte 6) se espera respuesta del sistema	Datos	se espera que el formato de salida sea del tipo pdf	x		El reporte se encuentra en el formato requerido.	
pf16	26	usuario operario acceso al Módulo ingresos	1)se ingresa a la página: de ID y operario 2)se ingresan datos 3) se accede a modulo ingresos 4) se visualizán los campos ingreso por vehículo, ingresos por conductor, tablas de ingresos por fechas, ingresos por cliente y /o particulares, estadísticas 5) se elige la opción revisión o consulta 6) se espera respuesta del sistema	Datos	Visualización de campos consultados		x	No se encuentra disponible dentro de la versión prototipo	los ingresos dependen directamente del tipo de parametrización que se realice
pf17	26	usuario operario acceso al Módulo ingresos	1)se ingresa a la página: de ID y operario 2)se ingresan datos 3) se accede a modulo ingresos 4) se visualizán los campos ingreso por vehículo, ingresos por conductor, tablas de ingresos por fechas, ingresos por cliente y /o particulares, estadísticas 5) se elige la opción ingreso 6) se ingresan los datos 7) se espera respuesta del sistema	Datos	Visualización de ingreso exitoso de datos		x	No se encuentra disponible dentro de la versión prototipo	los ingresos dependen directamente del tipo de parametrización que se realice

pf18	27	modulo servicios realizados	1)se ingresa a la página: de ID de usuario 2)se ingresan datos 3) se accede a modulo servicios realizados 4) se visualizán los campos: ingreso servicio, historial 5) se elige la opción a ingresar 6) se ingresan los datos 7) se espera respuesta del sistema	Datos	Visualización de ingreso exitoso dde datos	x		El servicio es registrado con éxito	
pf19	27	modulo servicios realizados	1)se ingresa a la página: de ID de usuario 2)se ingresan datos 3) se accede a modulo servicios realizados 4) se visualizán los campos: ingreso servicios,historial 5) se elige la opción historial, existe la posibilidad de filtrar por fechas conductor o vehículo (matrícula o cédula) 6) se ingresan los datos 7) se espera respuesta del sistema	Datos	Visualización de datos consultados	x		Los datos ingresados: por cédula o matrícula son visualizados.	
pf20	14	usuario invitado	1)se ingresa a la página: ingreso como invitado 2)se visualiza opción 3) se accede al sistema como invitado 4) se visualizán los campos: consulta, registro quejas 5) se elige la opción: consultar vehículo 6) se ingresan los datos 7) se espera respuesta del sistema	Datos	Visualización de datos consultados	x		La consulta se realiza con éxito.	
pf21	14	usuario invitado	1)se ingresa a la página: ingreso como invitado 2)se visualiza opción 3) se accede al sistema como invitado 4) se visualizán los campos: consulta, registro quejas y reportes 5) se elige la opción: registrar queja 6) se ingresan los datos 7) se espera respuesta del sistema	Datos	Visualización registro de queja y reporte exitoso	x		La queja se registra con éxito.	

pruebas de seguridad	ps1	18	Ingreso de datos invalidos en la actualización	1) Se ingresa a la página: 2) Se modifican datos basicos del conductor 3) se deja un campo sin diligenciar	Datos	Generar un mensaje de error por parte del sistema.	x	No es posible hacer modificaciones en datos basicos, solo es posible modificar datos de contacto.
	ps2	22	ingreso de datos incompletos en SOAT	1) Se ingresa a la página: 2) No se ingresan todos los datos en el apartado de SOAT	Datos	Generar un mensaje de error por parte del sistema.	x	No es posible dejar campos sin diligenciar, referentes al SOAT

ANEXO 8

MATRIZ DE TRAZABILIDAD

CU \ HU	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
1																													
2																													
3																													
4																													
5																													
6																													
7																													
8																													
9																													
10																													
11																													
12																													
13																													
14																													
15																													
16																													
17																													
18																													
19																													
20																													
21																													
22																													
23																													
24																													
25																													
26																													
27																													
Diagrama CU	Usuario Registrado			Usuario Administrador						Usuario Operario																	Usuario Invitado		
NOTAS:	- El número identificador de caso de uso es el mismo que hace										Convencione		Relación HU/CU					Relación Inexistente											

	referencia a los diagramas de secuencia. - Esta tabla muestra la relación existente entre las HU y los CU, para ver la relación entre CU diríjase a las tablas de especificación de CU.	s		
--	--	---	--	--