ELEMENTOS DE LA DOCUMENTACIÓN DEL PROYECTO FINAL DE DESARROLLO DE SISTEMAS EN RED

Índice

[INTRODUCCIÓN 2](#_Toc11595195)

[REQUERIMIENTOS 2](#_Toc11595196)

[Contexto 2](#_Toc11595197)

[Clases de usuarios 2](#_Toc11595198)

[Requerimientos funcionales 2](#_Toc11595199)

[Requerimientos no funcionales 2](#_Toc11595200)

[ANÁLISIS Y DISEÑO 2](#_Toc11595201)

[Diseño arquitectónico 2](#_Toc11595202)

[Vista de casos de uso 2](#_Toc11595203)

[Vista lógica 5](#_Toc11595204)

[Vista de implementación 5](#_Toc11595205)

[Vista de procesos 5](#_Toc11595206)

[Vista de Despliegue 5](#_Toc11595207)

[Modelo de datos 5](#_Toc11595208)

[Prototipo UI Cliente 5](#_Toc11595209)

[CONSTRUCCIÓN 5](#_Toc11595210)

[Selección justificada de lenguajes de programación 5](#_Toc11595211)

[Estándares de codificación 5](#_Toc11595212)

[Selección justificada de tecnologías 5](#_Toc11595213)

[Selección justificada de prácticas de construcción 5](#_Toc11595214)

[Prácticas de codificación 5](#_Toc11595215)

[Trabajo en equipo 5](#_Toc11595216)

[Aseguramiento de la calidad 5](#_Toc11595217)

[Herramientas 5](#_Toc11595218)

[PRUEBA 5](#_Toc11595219)

[Plan general de prueba (unidad e integridad) 5](#_Toc11595220)

[Procedimiento de prueba 5](#_Toc11595221)

[Casos de prueba 5](#_Toc11595222)

[Resultado 5](#_Toc11595223)

[Conclusiones 5](#_Toc11595224)

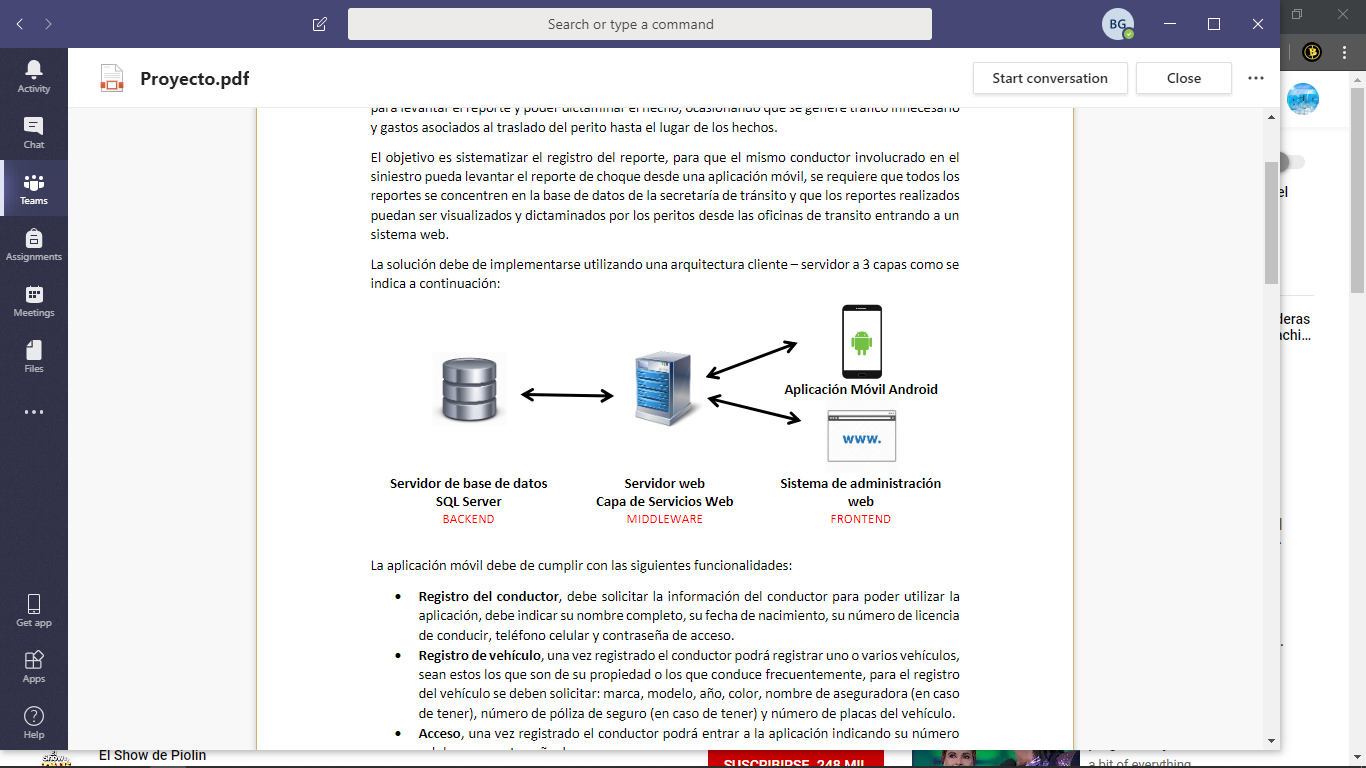
# INTRODUCCIÓN

# REQUERIMIENTOS

## Contexto

La secretaría de tránsito del estado de Veracruz, requiere implementa un sistema informático para llevar la gestión de los reportes de siniestros vehiculares, debido a que actualmente los reportes se hacen de manera manual y requieren que un perito calificado se traslade hasta el lugar del hecho para levantar el reporte y poder dictaminar el hecho, ocasionando que se genere tráfico innecesario y gastos asociados al traslado del perito hasta el lugar de los hechos.

El objetivo es sistematizar el registro del reporte, para que el mismo conductor involucrado en el siniestro pueda levantar el reporte de choque desde una aplicación móvil, se requiere que todos los reportes se concentren en la base de datos de la secretaría de tránsito y que los reportes realizados puedan ser visualizados y dictaminados por los peritos desde las oficinas de transito entrando a un sistema web.

La solución debe de implementarse utilizando una arquitectura cliente –servidor a 3 capas como se indica a continuación:

La capa de servicios web debe de contener todos los métodos necesarios para recibir y proveer de información tanto a la aplicación móvil, como a el sistema web.

La base de datos debe contener las tablas y campos necesarios para almacenar la información solicitada por los 2 aplicativos.

## Clases de usuarios

En este sistema se han identificado 2 tipos de usuarios específicos:

**Usuario Web**:

* Tendrá acceso al sistema web, dependiendo el rol que tenga (administrativo, perito, soporte y atención) introduciendo su usuario y contraseña.
* Visualizar reportes generados desde la aplicación móvil por un conductor. Cada reporte podrá ser visto de una forma más detallada, donde se mostrarán los datos e imágenes del accidente.
* Realizar un dictamen al reporte de acuerdo al análisis hecho por el perito.

**Conductor**:

* Tendrá acceso a la aplicación móvil. Proporcionando sus datos personales creará un registro en el sistema para, posteriormente, ingresar en la aplicación.
* El conductor será quien genere los reportes, brindando información del accidente, de los vehículos involucrados y los conductores.

## Requerimientos funcionales

Sistema web:

* Registro de usuarios con un rol definido (Administrativo, perito, Soporte, Atención) otorgando un usuario y clave de acceso.
* Acceso al sistema web mediante un usuario y clave de acceso de acuerdo al rol de usuario con el que se registró.
* Visualizar reportes filtrados por fecha, estatus (Pendiente, Dictaminado) y por ciudad donde se realizó el reporte (Coordenadas latitud y longitud).
* Dictaminar reporte haciendo una observación indicando a favor de quien fue el dictamen.

Aplicación móvil:

* Acceso a la aplicación móvil mediante el usuario y clave de acceso.
* Registro de conductor otorgando un usuario y clave de acceso.
* Actualización de datos personales del conductor.
* Registro de vehículos otorgando información del vehículo actualizada.
* Modificación de los vehículos registrados para mantener actualizada la información.
* Eliminación de vehículos en caso de que este ya no pertenezca al conductor quien lo registro.
* Generar reportes indicando la información del accidente y el punto donde fue originado con las coordenadas del dispositivo móvil (latitud y longitud).
* Ver historial de reportes generados por el conductor indicando el estado de su reporte.
* Ver el dictamen del reporte hecho por un perito desde el sistema web.

## Requerimientos no funcionales

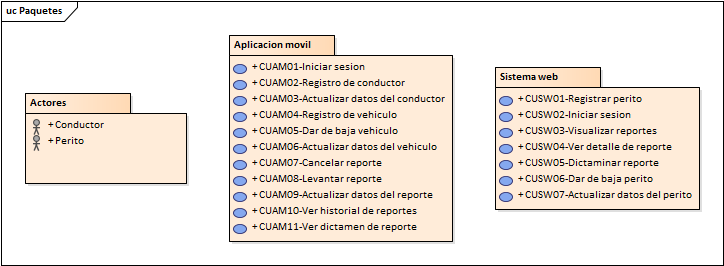
* **Disponibilidad**. El sistema deberá tener como prioridad estar siempre en línea para cualquier evento desafortunado del conductor.
* **Tiempo de respuesta**. La comunicación entre la aplicación móvil y el sistema será la más rápida posible, solo interviniendo factores externos de internet como conectividad o velocidad de internet.

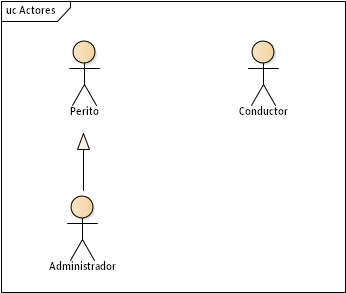
# ANÁLISIS Y DISEÑO

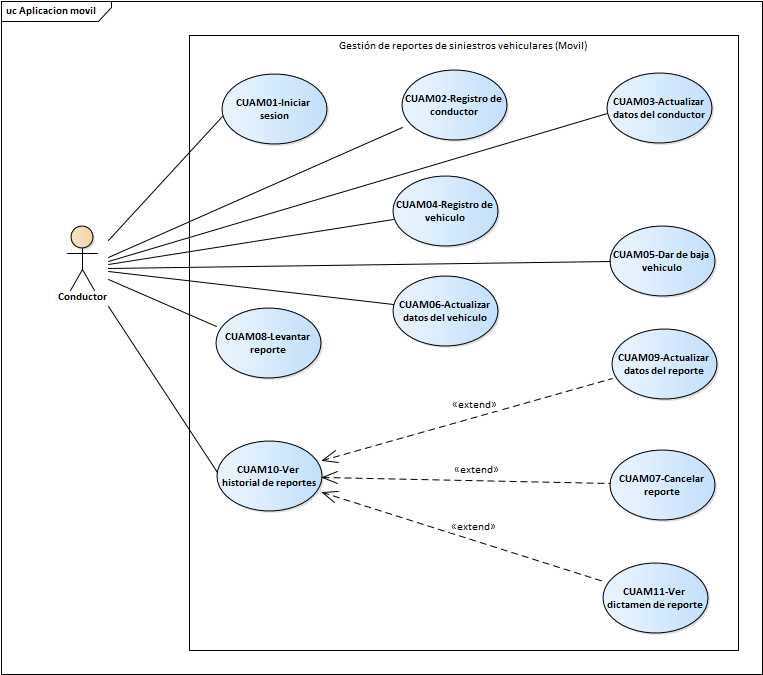
## Diseño arquitectónico

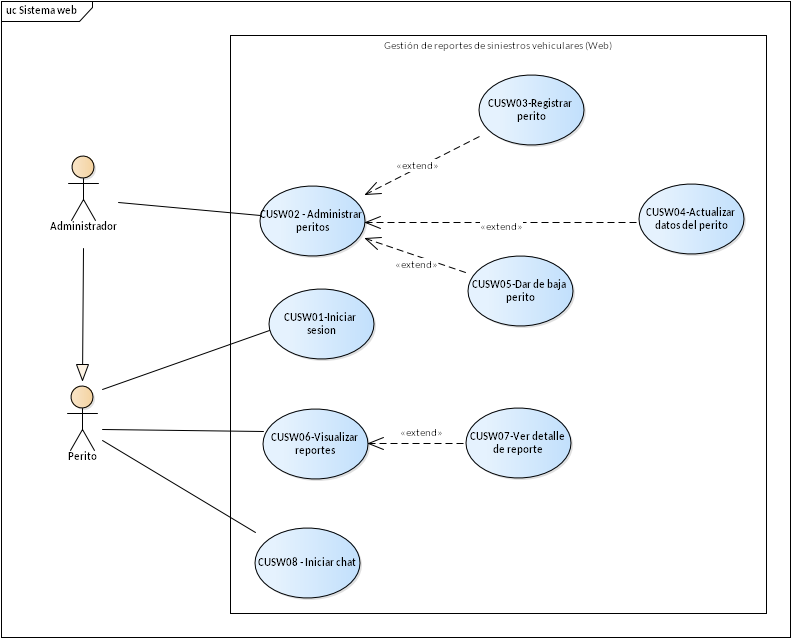
### Vista de casos de uso

#### Modelo de casos de uso









#### Descripciones de casos de uso

### Vista lógica

#### Diagrama UML de clases (servidor y cliente)

#### Diagrama UML de paquetes (servidor y cliente)

## Vista de implementación

#### Diagramas UML de componentes

### Vista de procesos

#### Diagramas UML de secuencia

#### Diagramas UML de robustez

### Vista de Despliegue

#### Diagrama UML de despliegue

## Modelo de datos

## Prototipo UI Cliente

# 

# CONSTRUCCIÓN

## Selección justificada de lenguajes de programación

La selección de lenguajes para el proyecto es escogida por el asesor de la experiencia educativa. El lenguaje de programación para la parte web es C# y para la parte móvil: Java.

Estos dos lenguajes de programación fueron enseñados con ejemplos prácticos dentro de las sesiones de clases.

## Estándares de codificación

Para el sistema web en C# no existe un estándar oficial sin embargo se siguen las nomenclaturas generales para el nombrado de métodos y variables “CamelCase” y en clases “lowerCase”, comentarios pertinentes, nombres descriptivos a métodos y variables.

Para la aplicación móvil se seguirá el estándar de codificación proporcionado por la página oficial de Android:

<https://source.android.com/setup/contribute/code-style>

## Selección justificada de tecnologías

Entornos de desarrollo:

**Android Studio** – IDE oficial de Android gratuito para el desarrollo de aplicaciones.

**Visual Studio Enterprise** – IDE para desarrollo del sistema web de paga. La licencia es otorgada por la UV.

Sistema gestor de base de datos:

**SQL Server** – Sistema gestor de bases de datos para almacenamiento de datos del sistema web y aplicación móvil

Frameworks:

**.NET Core** – Versión reducida de .NET Framework para la implementación de web api del sistema web.

**Volley** – Librería para conectarse a servicios web desde Android

## Selección justificada de prácticas de construcción

### Prácticas de codificación

#### 1 Introducción

En este punto se explica brevemente la funcionalidad de este estándar, así como algunos términos que podrían venir en el documento y nos ayuden a identificar partes del código ejemplo.

#### 2 conceptos básicos sobre archivos de origen

 En ésta sección se muestran las normas de la extensión del nombre origen del archivo (.java) , la codificación utf-8 y el uso de caracteres especiales:

* Como marcar un espacio
* No está permitido el tabulador para la sangría
* Que espacios están permitidos
* Cómo poner símbolos no ASCII

#### 3 Estructura del archivo fuente

Aquí se explica cómo es que debe ir estructurado un archivo fuente:

1. Licencia o copyright
2. Paquetes
3. Importar declaraciones
4. Clase de nivel superior

Siendo separados por una línea entre cada sección.

Para declarar un paquete o importar algo no es necesario tomar en cuenta el límite de caracteres permitidos o recomendados por línea.

No se debe importar cosas que no se necesiten. Las importaciones se separan; primero estáticas, línea en blanco y no estáticas en orden ASCII.

Cada clase de gran nivel tiene archivo propio. Los atributos o métodos de la clase estarán ordenados lógicamente y no por fecha. Si existen métodos de sobrecarga deberán ir secuenciados.

#### 4 formateo

El uso de llaves para estructuras como if, else, for, while, etcétera, no es opcional.

Para los bloques no vacíos los corchetes irán estilo egipcio:

* Sin salto de línea antes de la llave de apertura.
* Salto de línea después de la llave de apertura.
* Salto de línea antes del corsé de cierre.
* Salto de línea después del corchete de cierre, solo si ese corchete termina un enunciado o termina el cuerpo de un método, constructor o clase con nombre

Para los bloques vacíos pueden cerrarse inmediatamente si es que no pertenece a instrucción dependiente de otros bloques.

El indentado deberá ser de dos espacios.

Debe haber una declaración por línea.

El límite de una línea de código debe ser de al menos 100 caracteres, a excepciones de:

* Ser necesario por URLs o direcciones
* Declarar paquetes e importaciones
* Comentarios que pueden pegarse en caparazón

Cuando romper una línea:

* Símbolo que no es de asignación antes del símbolo, también aplica para:
* El separador de puntos ( .)
* Los dos puntos de una referencia de método ( ::)
* Un ampersand en un tipo vinculado ( )<T extends Foo & Bar>
* Una tubería en un bloque de captura ( ).catch (FooException | BarException e)
* Paréntesis abierto de un método o constructor ( ()
* Una coma ( ,)
* Nunca se rompe adyacente a la flecha lamba, sólo sr se rompe después

Cuando hay líneas de continuación puede haber hasta 4 espacios.

Siempre existe una línea en blanco vertical.

Espacios horizontales para:

* Palabras reservadas desde el primer paréntesis
* Palabras reservadas de una llave de cierre
* Antes de llave abierta
* Signos como & , | , : y ->
* Después ,:;o el paréntesis de cierre ( )) de un molde
* Ambos lados de //
* Tipo de dato
* Entre [] o …

Alineación horizontal no es requerida.

No declarar más de una variable por línea, solo en for. Declarar las variables cerca del punto donde se van a utilizar.

Los inicializadores de matrices pueden ser de tipo bloque. No se pueden declarar tipo C.

Cada swicht debe tener su default, pero para clases enum se puede omitir.

Debe haber una anotación por línea y se colocan luego del bloque de documentación.

El sangrado de los comentarios es el mismo al código y puede ser  /\*   \*/ o  // .

Orden de modificadores:

public protected private abstract default static final transient volatile synchronized native strictfp

para literales numéricos se utilizan mayúsculas y nunca minúsculas.

#### 5 nombrando

No se usan prefijos o sufijos especiales.

Nombres de paquetes van en minúsculas sin guiones bajos.

Los nombres de clases van en UpperCamelCase, clases que se están probando terminaran en Test.

Los nombres de métodos van en lowerCamelCase, pueden usar guion bajo.

Nombre de constantes es con mayúsculas con un único guion bajo.

Nombres de campos no constantes, parámetros, variables locales son tipo loweCamelCase.

#### 6 Prácticas de programación

Un método está marcado con la @Override anotación siempre que esté permitido.

Toda excepción se debe atrapar y no ignorar.

Cuando se debe calificar una referencia a un miembro de clase estática, se califica con el nombre de esa clase, no con una referencia o expresión del tipo de esa clase.

No se usan finalizadores.

#### 7 Javadoc

Forma general de java doc:

/ \*\*

 \* Aquí se escriben varias líneas de texto de Javadoc,

 \* se envuelve normalmente ...

 \* / public int method ( String p1 ) { ... }

### Trabajo en equipo

El sistema estará en constante respaldo gracias a git, un controlador de versiones gratuito, en el cual los integrantes del equipo podrán estar al tanto de cambios realizados en cada sistema.

Utilizando herramientas de interfaz gráfica como GitHub o por con sola con GitBash.

### Aseguramiento de la calidad

### Herramientas

Para Android Studio se instala el plugin CheckStyle el cual ayuda a seguir el estándar de codificación de java.

# PRUEBA

## Plan general de prueba (unidad)

Se generarán una serie de 5 casos de prueba por cada caso de uso, probando 1 flujos exitosos, 2 flujos de alternos y 2 flujo de excepción.

## Procedimiento de prueba

Se planteará entradas al sistema, salidas esperadas y salidas obtenidas para finalmente mostrar si pasó o no la prueba.

## Casos de prueba

|  |  |  |  |
| --- | --- | --- | --- |
| **Caso de uso:** | Iniciar Sesión | **ID:** | CUAM01 |
| **Requisitos** | **Entrada** | **Salida esperada** | **Salida obtenida** |
| Deberá existir un usuario registrado con el teléfono 2281721501 y clave 321 | Teléfono = 2281721501  Password = 321 | Logeo a la aplicación móvil |  |
| Teléfono = 2281721501  Password = 321 | Error al conectar con el servidor |  |
| Ninguna | Seleccionar botón registrarse | Despliegue de la pantalla Registro01 |  |
| Celular sin conexión a internet | Teléfono = 2281721501  Password = 321 | Error al conectar con el servidor |  |
| Servidor caido | Teléfono = 2281721501  Password = 321 | Error al conectar con el servidor |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Caso de uso:** | Registro de conductor | **ID:** | CUAM02 |
| **Requisitos** | **Entrada** | **Salida esperada** | **Salida obtenida** |
| Ninguno | Telefono = 2283017645  Password = 123456  Password2 = 123456  Nombre = Luis  Apellido paterno = Bonilla  Apellido materno = Ramírez  Fecha = 26/08/1996  No. Licencia = 12345678 | Registro exitoso |  |
| Teléfono = 2281721501  Password = 123456  Password2 = 654321 | Contraseñas no coinciden |  |
| Teléfono = 123  Password = 123456  Password2=123456 | Ingrese un celular válido |  |
| Celular sin conexión a internet | Telefono = 2283017645  Password = 123456  Password2 = 123456  Nombre = Luis  Apellido paterno = Bonilla  Apellido materno = Ramírez  Fecha = 26/08/1996  No. Licencia = 12345678 | Error al conectar con el servidor |  |
| Servidor caido | Telefono = 2283017645  Password = 123456  Password2 = 123456  Nombre = Luis  Apellido paterno = Bonilla  Apellido materno = Ramírez  Fecha = 26/08/1996  No. Licencia = 12345678 | Error al conectar con el servidor |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Caso de uso:** | Actualizar datos del conductor | **ID:** | CUAM03 |
| **Requisitos** | **Entrada** | **Salida esperada** | **Salida obtenida** |
| Usuario logeado en la aplicación | Password = 123456  Password2 = 123456  Nombre = Luis  Apellido paterno = Bonilla  Apellido materno = Ramírez  Fecha = 26/08/1996  No. Licencia = 12345678 | Actualización exitosa |  |
| Password = 123456  Password2 = 654321  Nombre = Luis  Apellido paterno = Bonilla  Apellido materno = Ramírez  Fecha = 26/08/1996  No. Licencia = 12345678 | Contraseñas no coinciden |  |
| Password = Password2 =  Nombre =  paterno =  Apellido materno =  Fecha = 26/08/1996  No. Licencia = | Asegúrese de ingresar todos los datos |  |
| Celular sin conexión a internet | Password = 123456  Password2 = 123456  Nombre = Luis  Apellido paterno = Bonilla  Apellido materno = Ramírez  Fecha = 26/08/1996  No. Licencia = 12345678 | Error al conectar con el servidor |  |
| Servidor caido | Password = 123456  Password2 = 123456  Nombre = Luis  Apellido paterno = Bonilla  Apellido materno = Ramírez  Fecha = 26/08/1996  No. Licencia = 12345678 | Error al conectar con el servidor |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Caso de uso:** | Registro de vehículo | **ID:** | CUAM04 |
| **Requisitos** | **Entrada** | **Salida esperada** | **Salida obtenida** |
| Usuario logeado en la aplicación | Placas = KLS-CM-0293  Marca = TOYOTA  Modelo = Corolla  Año = 2001  Color = Gris plateado  Nombre aseguradora = Banamex  No. Poliza = 213876123 | Registro exitoso |  |
| Placas =  Marca =  Modelo =  Año =  Color =  Nombre aseguradora =  No. Poliza = | Asegúrese de ingresar todos los datos |  |
| Seleccionar botón menú opciones | El sistema despliega el menú desplegable |  |
| Celular sin conexión a internet | Placas = KLS-CM-0293  Marca = TOYOTA  Modelo = Corolla  Año = 2001  Color = Gris plateado  Nombre aseguradora = Banamex  No. Poliza = 213876123 | Error al conectar con el servidor |  |
| Servidor caido | Placas = KLS-CM-0293  Marca = TOYOTA  Modelo = Corolla  Año = 2001  Color = Gris plateado  Nombre aseguradora = Banamex  No. Poliza = 213876123 | Error al conectar con el servidor |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Caso de uso:** | Actualizar datos del vehículo | **ID:** | CUAM05 |
| **Requisitos** | **Entrada** | **Salida esperada** | **Salida obtenida** |
| Deberá existir un vehículo registrado con los siguientes datos:  Placas = KLS-CM-0293  Marca = TOYOTA  Modelo = Corolla  Año = 2001  Color = Gris plateado  Nombre aseguradora = Banamex  No. Poliza = 213876123 | Placas = KLS-CM-0293  Marca = Verso  Modelo = Corolla  Año = 2001  Color = Gris plateado  Nombre aseguradora = Banamex  No. Poliza = 1231231231 | Modificación exitosa |  |
| Placas =  Marca =  Modelo =  Año =  Color =  Nombre aseguradora =  No. Poliza = | Asegúrese de ingresar todos los datos |  |
| Placas = KLS-CM-0293  Marca = Verso  Modelo = Corolla  Año = 2  Color = Gris plateado  Nombre aseguradora = Banamex  No. Poliza = 1231231231 | Ingrese un año válido |  |
| Celular sin conexión a internet | Placas = KLS-CM-0293  Marca = TOYOTA  Modelo = Corolla  Año = 2001  Color = Gris plateado  Nombre aseguradora = Banamex  No. Poliza = 213876123 | Error al conectar con el servidor |  |
| Servidor caido | Placas = KLS-CM-0293  Marca = TOYOTA  Modelo = Corolla  Año = 2001  Color = Gris plateado  Nombre aseguradora = Banamex  No. Poliza = 213876123 | Error al conectar con el servidor |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Caso de uso:** | Dar de baja vehículo | **ID:** | CUAM06 |
| **Requisitos** | **Entrada** | **Salida esperada** | **Salida obtenida** |
| Deberá existir un vehículo registrado con los siguientes datos:  Placas = KLS-CM-0293  Marca = TOYOTA  Modelo = Corolla  Año = 2001  Color = Gris plateado  Nombre aseguradora = Banamex  No. Poliza = 213876123 | Conductor selecciona la opción eliminar y confirma la eliminación | Eliminación exitosa |  |
| El conductor selecciona el menú de más opciones | El sistema despliega el menú de más opciones |  |
| Conductor selecciona la opción eliminar y cancelar la eliminación | El sistema cierra la ventana de confirmación sin eliminar el vehículo |  |
| Celular sin conexión a internet | Conductor selecciona la opción eliminar y confirma la eliminación | Error al conectar con el servidor |  |
| Servidor caido | Conductor selecciona la opción eliminar y confirma la eliminación | Error al conectar con el servidor |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Caso de uso:** | Levantar Reporte | **ID:** | CUAM07 |
| **Requisitos** | **Entrada** | **Salida esperada** | **Salida obtenida** |
| El conductor deberá estar logeado | Placas = JDD-AS-1284  Marca = TOYOTA  Modelo = Yaris  Color = Azul  Nombre = Luis Gerardo  Poliza de seguro = 1728132  4 fotos en el reporte | El sistema despliega la lista de vehículos registrados por el conductor |  |
| Placas =  Marca =  Modelo =  Color =  Nombre =  Poliza de seguro =  4 fotos en el reporte | Asegúrese de ingresar todos los datos. |  |
| Placas = JDD-AS-1284  Marca = TOYOTA  Modelo = Yaris  Color = Azul  Nombre = Luis Gerardo  Poliza de seguro = 1728132 | Por favor, introduzca 4 fotografías como mínimo |  |
| Celular sin conexión a internet | El conductor selecciona siguiente y selecciona su vehículo | Error al conectar con el servidor |  |
| Servidor caido | El conductor selecciona siguiente y selecciona su vehículo | Error al conectar con el servidor |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Caso de uso:** | Ver historial reportes | **ID:** | CUAM08 |
| **Requisitos** | **Entrada** | **Salida esperada** | **Salida obtenida** |
| El conductor deberá estar logeado | El conductor selecciona la opción historial reportes | El sistema despliega la lista de los reportes registrados por el conductor |  |
| El conductor selecciona mis datos | El sistema despliega la pantalla mis datos |  |
| El conductor selecciona el botón atrás en su dispositivo móvil | El sistema despliega la pantalla Login |  |
| Celular sin conexión a internet | El conductor selecciona la opción historial reportes | Error al conectar con el servidor |  |
| Servidor caido | El conductor selecciona la opción historial reportes | Error al conectar con el servidor |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Caso de uso:** | Ver dictamen reporte | **ID:** | CUAM09 |
| **Requisitos** | **Entrada** | **Salida esperada** | **Salida obtenida** |
| El conductor deberá estar logeado | El conductor selecciona la opción ver dictamen desde la vista del reporte | El sistema despliega la ventana Dictamen |  |
| El conductor selecciona mis datos | El sistema despliega la pantalla mis datos |  |
| El conductor selecciona el botón atrás en su dispositivo móvil | El sistema despliega la pantalla MenuPrincipal |  |
| Celular sin conexión a internet | El conductor selecciona la opción ver dictamen desde la vista del reporte | Error al conectar con el servidor |  |
| Servidor caido | El conductor selecciona la opción ver dictamen desde la vista del reporte | Error al conectar con el servidor |  |

## Resultado

Resultados en la aplicación móvil

|  |  |  |
| --- | --- | --- |
| **Nombre del probador:** | Luis Gerardo Bonilla Ramírez | |
| **Nombre Caso de Uso** | **ID** | **Resultados:** |
| Iniciar sesión | CUAM01 | |  |  | | --- | --- | | Prueba 1 | PASO | | Prueba 2 | PASO | | Prueba 3 | PASO | | Prueba 4 | NO PASO | | Prueba 5 | NO PASO | |
| Registro de conductor | CUAM02 | |  |  | | --- | --- | | Prueba 1 |  | | Prueba 2 |  | | Prueba 3 |  | | Prueba 4 |  | | Prueba 5 |  | |
| Actualizar datos del conductor | CUAM03 | |  |  | | --- | --- | | Prueba 1 |  | | Prueba 2 |  | | Prueba 3 |  | | Prueba 4 |  | | Prueba 5 |  | |
| Registro de vehículo | CUAM04 | |  |  | | --- | --- | | Prueba 1 |  | | Prueba 2 |  | | Prueba 3 |  | | Prueba 4 |  | | Prueba 5 |  | |
| Actualizar datos del vehículo | CUAM05 | |  |  | | --- | --- | | Prueba 1 |  | | Prueba 2 |  | | Prueba 3 |  | | Prueba 4 |  | | Prueba 5 |  | |
| Dar de baja vehículo | CUAM06 | |  |  | | --- | --- | | Prueba 1 |  | | Prueba 2 |  | | Prueba 3 |  | | Prueba 4 |  | | Prueba 5 |  | |
| Levantar reporte | CUAM07 | |  |  | | --- | --- | | Prueba 1 |  | | Prueba 2 |  | | Prueba 3 |  | | Prueba 4 |  | | Prueba 5 |  | |
| Ver historial reportes | CUAM08 | |  |  | | --- | --- | | Prueba 1 |  | | Prueba 2 |  | | Prueba 3 |  | | Prueba 4 |  | | Prueba 5 |  | |
| Ver dictamen de reporte | CUAM09 | |  |  | | --- | --- | | Prueba 1 |  | | Prueba 2 |  | | Prueba 3 |  | | Prueba 4 |  | | Prueba 5 |  | |

# Conclusiones