

Leonardo Manrique

Ana M. Sánchez Mejía

Maria P. Nizo

Proyecto 3

Contenido	
Contexto.....	1
Requerimiento Funcionales.....	2
Requerimientos No Funcionales	2
Restricciones del Proyecto	2
¿Qué cosas salieron bien y qué cosas salieron mal?	3
¿Qué decisiones resultaron acertadas y qué decisiones fueron problemáticas?	4
¿Qué tipo de problemas tuvieron durante el desarrollo de los proyectos y a qué se debieron?	4
Conclusión.....	5

Contexto

A través de este documento se analizará las entregas realizadas a lo largo del segundo semestre del 2023 de la materia de Diseño y Programación. Para este proyecto se buscaba diseñar un sistema que permitiera llevar el control de una empresa de alquiler de vehículos, para diferentes actores del sistema y que se pudiera interactuar para reservas, inventario y pago. En general, la aplicación utilizaba cuatro tipos de usuarios: un administrador de la sede, el administrador general de la empresa, los empleados de la recepción, mantenimiento y los clientes, todos cuentan con usuario y contraseña establecidos a través de un registro en el sistema.

Para cada proyecto se requería en general una mejora en la programación realizada y en el de diseño o mejora del diseño con la entrega de un diagrama UML y un documento de diseño que permitiera sustentar los cambios realizados. En primer lugar, el proyecto 1 pedía la realización completa del programa implementando las funcionalidades requeridas con las restricciones. En segundo lugar, el proyecto 2 requería la implementación de una interfaz gráfica que permitiera ejecutar las acciones de los usuarios, adicionalmente toca realizar

gráfica de demanda de vehículos a lo largo del años, la realización de una interfaz a través de java swing. Por último, el proyecto 3 indicaba la realización métodos para las excepciones, incorporar nuevos requerimientos y una interfaz especial para los clientes.

Requerimiento Funcionales

- Registrar un nuevo vehículo.
- Dar un vehículo de baja.
- Configurar información de cada sede
- Modificar información de precios para los vehículos.
- Configurar información de seguros.
- Agregar un usuario al sistema.
- Modificar un usuario al sistema.
- Eliminar un usuario del sistema.
- Reportar a un vehículo a mantenimiento.
- Crear un cliente interno.
- Revisar vehículo.
- Crear la reserva especial.
- Entrega de un vehículo.
- Devolución de un vehículo.
- Recibir el pago.
- Registrar un nuevo conductor.
- Bloquear la tarjeta.
- Reserva del vehículo.
- Pago de seguros.
- Pago de 30%.
- Emitir recibos en PDF
- Realizar un descuento del 10%
- Crear usuarios.

Requerimientos No Funcionales

- La aplicación debe correr soportando varios usuarios participantes a la vez.
- La aplicación debe correr de forma rápida, con la mayor eficiencia posible.

Restricciones del Proyecto

1. Una de las reglas de dominio es que todos deben ingresar con un LogIn y una contraseña y dependiendo del usuario podrá realizar ciertas actividades en la aplicación.
2. Se asume que no existe un usuario malicioso que pueda modificar la información de la aplicación.
3. Otra regla de dominio es que la aplicación debe guardar toda la información para generar archivos log.

4. También se tiene que todo cliente debe realizar una reserva por internet, no se acepta que lleguen y creen la reserva en la sede. En las sedes solamente se atienden solicitudes extra de la reserva y se entrega el vehículo.
 5. Los únicos que pueden reportar que un vehículo necesita mantenimiento son los empleados y no los clientes.
 6. Se asume que todos los clientes pagan con tarjeta de crédito.
- Los trabajadores solo pueden cambiar el estado de los vehículos que están en su sede.

¿Qué cosas salieron bien y qué cosas salieron mal?

	¿Qué salió bien?	¿Qué salió mal?
Proyecto 1	<ul style="list-style-type: none"> Los diagramas realizados permiten entender lo que se iba a realizar. La información que la consola imprimía indicaba un avance. <p>La persistencia de la información fue consistente.</p>	<ul style="list-style-type: none"> Se pudo haber mejorado la realización del UML, algunas relaciones estuvieron mal realizadas y se pudo simplificar a través del uso de otras relaciones. <p>Algunas de las clases les faltaron atributos</p>
Proyecto 2	<ul style="list-style-type: none"> Las partes del interfaz realizadas estuvieron bien diseñadas. La implementación coincide con el diseño. 	<ul style="list-style-type: none"> El documento está muy básico. No se completó el diagrama de alto nivel Toca revisar las excepciones para hacer un buen proyecto El documento de diseño pudo haberse mejorado. <p>Toca hacer un botón de cerrar sesión.</p>

Proyecto 3	<ul style="list-style-type: none"> La forma de guardar la información por carpetas. 	<ul style="list-style-type: none"> La conexión de un usuario a través de git siguió sin ser algo claro como funcionalidad.
------------	--	---

¿Qué decisiones resultaron acertadas y qué decisiones fueron problemáticas?

	Decisiones Acertadas	Decisiones Problemáticas
Proyecto 1	<ul style="list-style-type: none"> El diseño de las clases permitió realizar bien el proyecto. 	<ul style="list-style-type: none"> No se tuvo en cuenta la opción de modificar la alineación No se tuvo en cuenta las restricciones como la forma de guardar la información
Proyecto 2	<ul style="list-style-type: none"> La separación entre la lógica y la interfaz. La implementación coincide con el diseño. 	<ul style="list-style-type: none"> Algunas de las excepciones no se manejaron de la mejor manera. No se actualiza de una vez el panel para seleccionar el vehículo a eliminar, es decir si cree uno y lo quiero borrar, tengo que volver a iniciar la app, lo mismo sucede en varias ocasiones como seguros, etc.
Proyecto 3	<ul style="list-style-type: none"> La modificación de las carpetas en las que se guardará la información Se realizaron pruebas para verificar ciertos métodos Se actualizó el diagrama UML 	

¿Qué tipo de problemas tuvieron durante el desarrollo de los proyectos y a qué se debieron?

	Problemas
--	-----------

Proyecto 1	<ul style="list-style-type: none"> • No se tuvo en cuenta de antemano los atributos y funcionalidades implicadas, lo que ocasiono que para la entrega límite no se pudiera completar todo esto. • La forma en la que se asociaban las clases
Proyecto 2	<ul style="list-style-type: none"> • Dados los errores y vacíos de la entrega 1, se alargó el tiempo dedicado al proyecto 1 y hubo detalles que no se tuvieron en cuenta en este proyecto como hacer un buen documento de diseño
Proyecto 3	<ul style="list-style-type: none"> • En el proyecto 2 faltaron excepciones que manejar en el modelo además de realizar los requerimientos necesarios en el proyecto 3 y las fallas se tuvo que trabajar en esto.
	<ul style="list-style-type: none"> • La parte de hacer el uso de diferentes plataformas de pago se nos dificulto un poco

Conclusión

En conclusión, las entregas realizadas presentaron problemas por falta de tiempo y revisión, sin embargo, la programación realizada, la interfaz y el diseño permitieron realizar una aplicación que manejará en general las funcionalidades pedidas. Se debe tener en cuenta siempre la persistencia de la información.