

## Informe de cierre del proyecto

*El equipo completa el informe de cierre del proyecto y lo envía al cliente. La información se compila a partir de la ejecución y recolección de información. El informe de cierre del proyecto analiza el cronograma y presupuesto para el período de informe.*

**Período del informe: 10/11/20 - 06/12/20**

**Equipo: Tomas Dilema, Bruno Fernández e Ignacio Olivera.**

### 1. Logros obtenidos para este período de informe

EDT ID: 5

Actividades ID: 5.1:

- ID: RFI1.2 Ayudas visuales
- ID: RFI1.5 Implementar funcionalidad de las listas "Agregar comida a la dieta"
- ID: RFI1.6 Implementar refresh de listas

EDT ID: 6

Actividades ID: 6.1.1:

- ID: RFPU1.0 Mejorar la cobertura del código
- ID: RFPU1.1 Implementar las pruebas para las nuevas funcionalidades

Actividades ID: 6.1.2:

- ID: RFC1.6 Disminuir la complejidad cognitiva
- ID: RFC1.7 Implementar las funcionalidades previstas

EDT ID: 7

Actividades ID: 7.1:

- ID: RFT1.0 Planificación y Diseño de Pruebas

Actividades ID: 7.2:

- ID: RFT1.1 Ejecución del set de pruebas y clases de equivalencias
- ID: RFT1.2 Ejecución del set de pruebas y partición de equivalencias
- ID: RFT1.3 Ejecución y set de pruebas de valores límite
- ID: RFT1.4 Ejecución y Set de Pruebas Testing exploratorio

Actividades ID: 7.3:

- ID: RFT1.5 Informe de Pruebas

EDT ID: 8

Actividades ID: 8.1

- ID: RBD1.0 Realización de builder e informe de logs

EDT ID: 9

Actividades ID: 9

- ID: RIF1.0 Ajustes del plan de riesgo
- ID: RIF1.1 Cambios de las líneas bases
- ID: RIF1.2 Reporte de los cambios
- ID: RIF1.3 Reporte estado de calidad final
- ID: RIF1.4 Realización del informe de cierre

### 2. Logros planificados, pero no completados en este período de informe

- Crear una clase de constantes para evitar la duplicación de código en las clases de dominio.
- El profesional tenga una lista de las comidas del usuario ingeridas.
- Quitar todos los bugs llevando la cantidad a 0 de SonarQube. Actualmente se bajó a 6 bugs de 10, que es passable en el SonarQube.

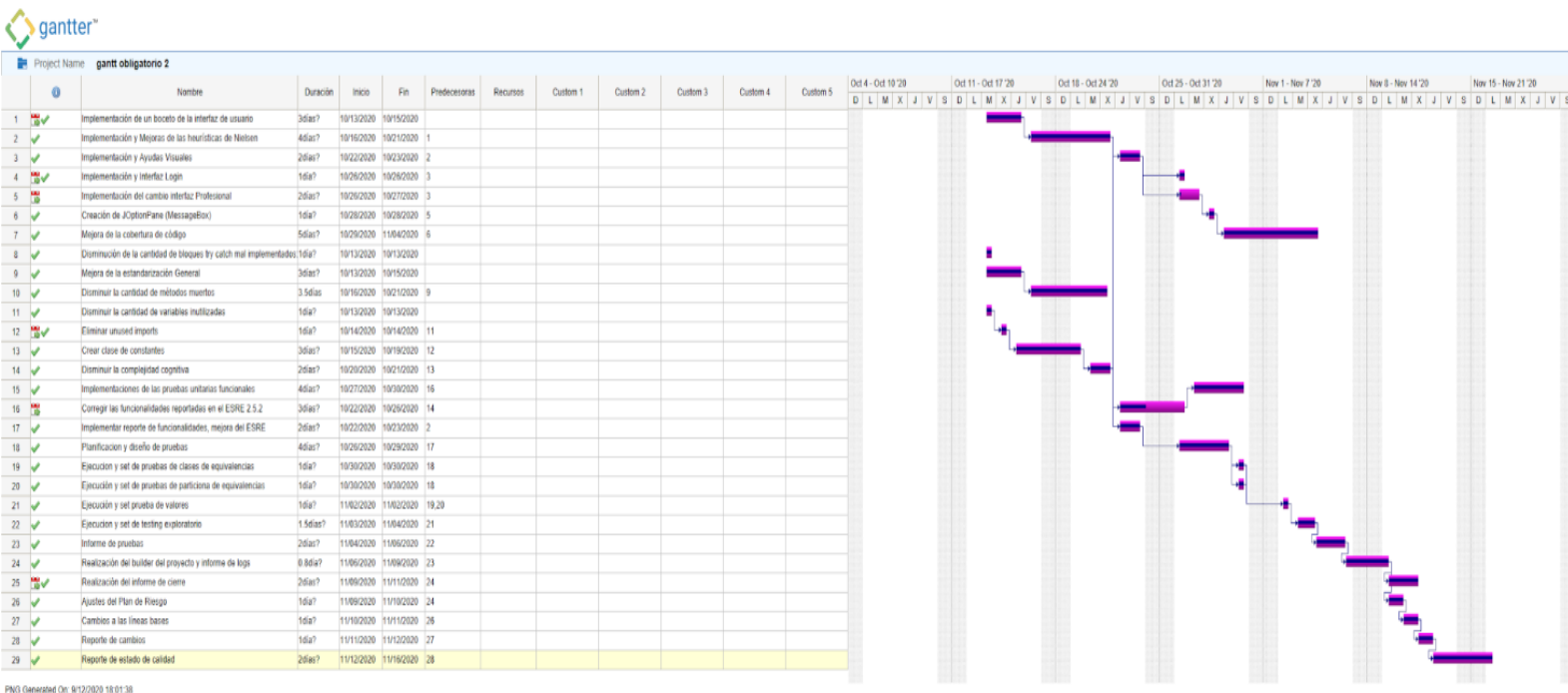
### 3. Causa raíz de las variaciones

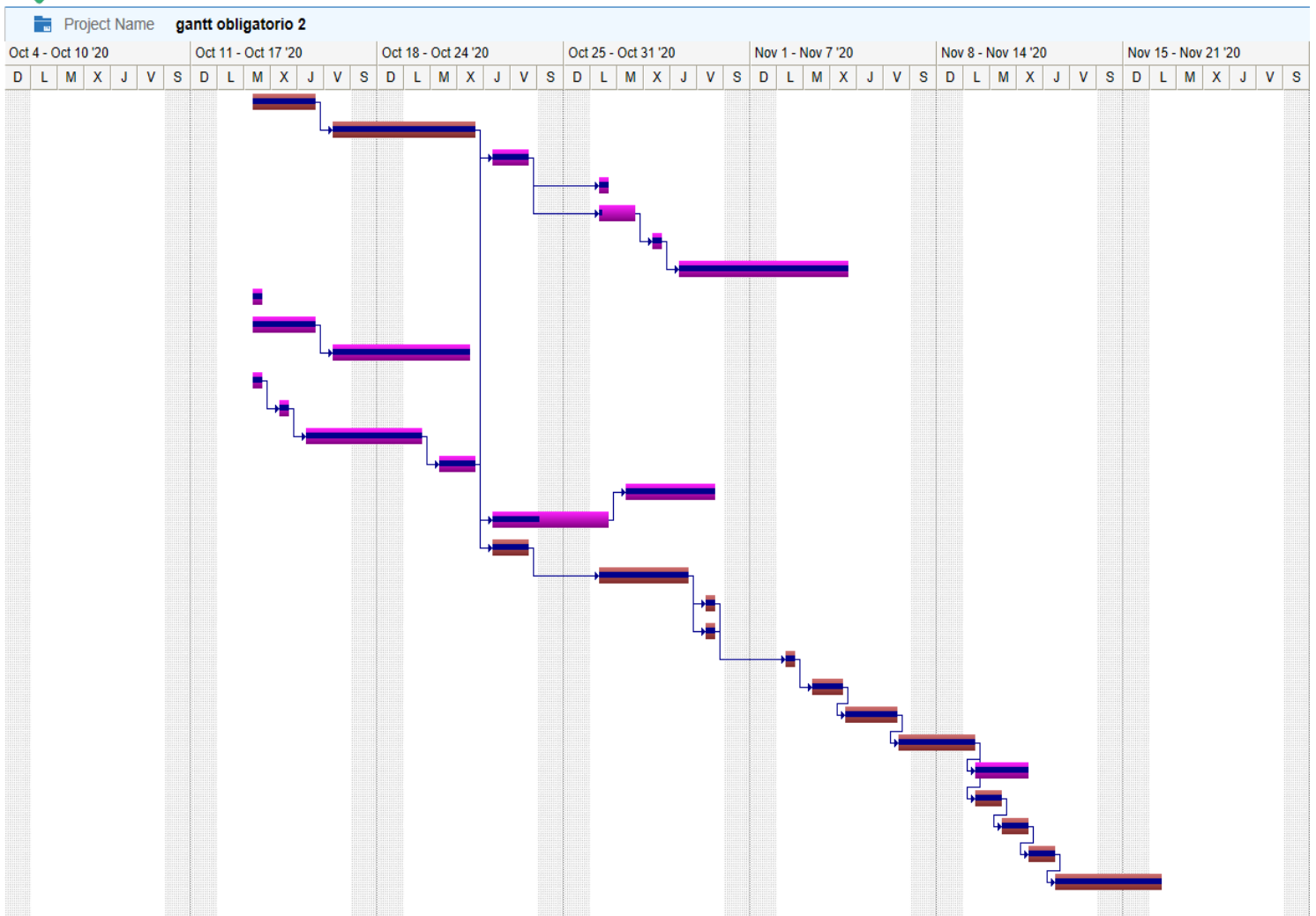
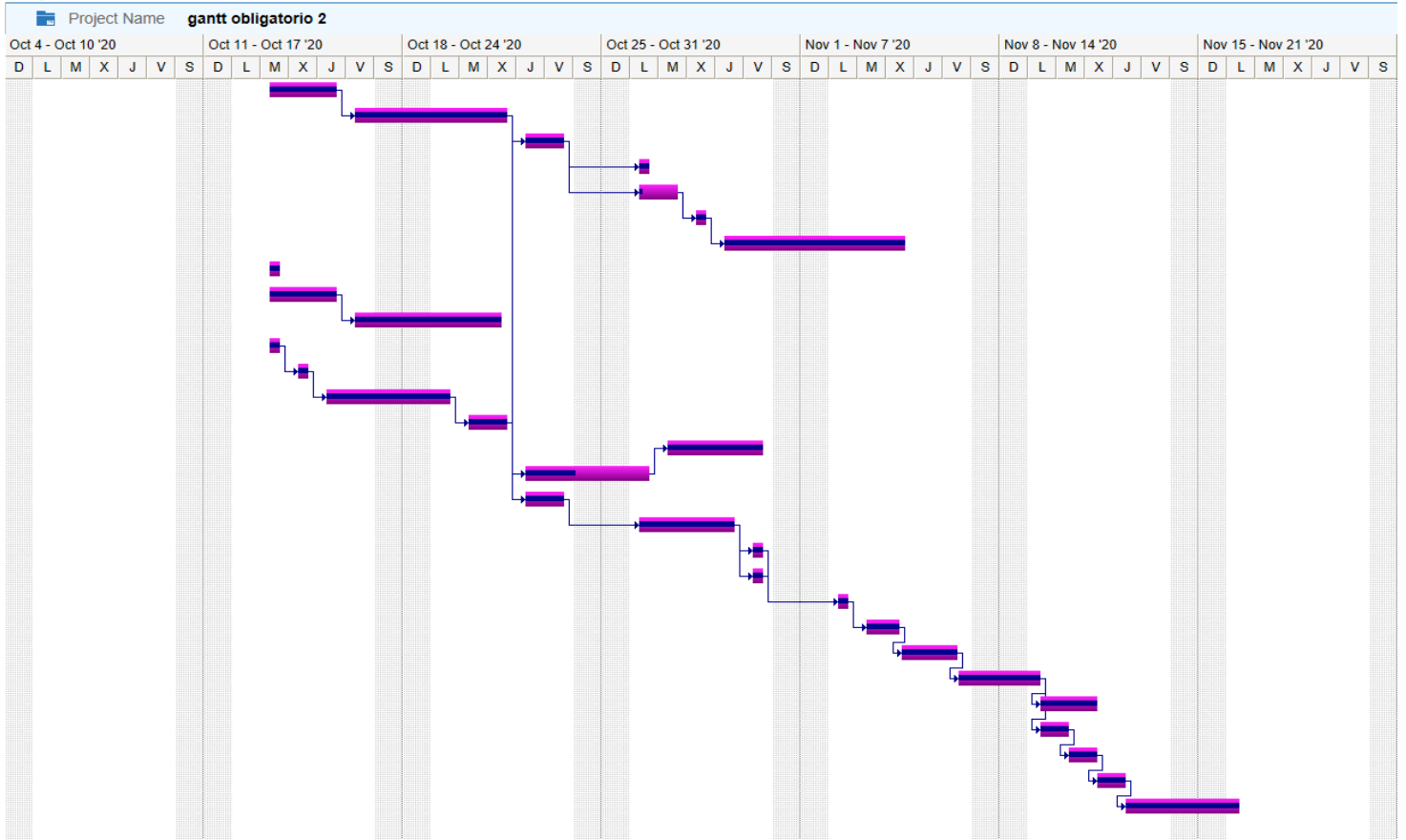
No se logró crear la clase constantes debido a la dificultad que planteaba poder implementarla en el programa para el tiempo que se poseía, luego la funcionalidad que permite al profesional tener una lista de las comidas ingeridas por el usuario no pudo ser implementada ya que requería una modificación masiva tanto del código fuente como de la interfaz, por lo que determinamos que no era posible implementarla sin descuidar otras funcionalidades de mayor relevancia para el proyecto. Por último, no se pudo bajar la cantidad de bugs en SonarQube debido a que dichos bugs requerían un mantenimiento superior al que se esperaba, así que se priorizo resolver aquellos bugs que se pudieran resolver de forma rápida y eficiente, para lograr un nivel aceptable de bugs que se pudieran resolver en futuras versiones del producto.

### 4. Impacto de la variación en los próximos hitos o en la fecha de finalización del proyecto

El hecho de no haber podido hacer la clase de constantes implica la generación de malas prácticas de codificación, lo que baja la calidad del producto y afecta seriamente al futuro mantenimiento del mismo. También no haber implementado la funcionalidad antes mencionada implica en un impedimento para el profesional, haciendo más difícil su trabajo y pudiendo resultar en un rechazo al software. Por último, no haber podido eliminar todos los bugs encontrados en el SonarQube puede afectar al correcto funcionamiento del software, generando también rechazo y trabajo extra en la próxima entrega del mismo.

Además no se tuvo en cuenta la cantidad de tiempo suficiente que llevaría analizar el estado final del sistema, tanto código como la interfaz. La implementación de las nuevas funcionalidades (en progreso) fueron actividades críticas como se mencionó en el informe de avance que influyeron también directamente en otras actividades.





## 5. Fondos gastados (presupuesto ejecutado) en este período de informe

EDT	ENTREGABLE	PV	EV	AC(U\$S)	CV	CPI
<b>ID: 5.1 Diseño interfaz</b>		882	882	614	264	1.44
ID: RFI1.2 Ayudas visuales		114	114	114	0	1
ID: RFI1.5 Implementar funcionalidad de las listas “Agregar comida a la dieta”		384	384	300	84	1.28
ID: RFI1.6 Implementar refresh de listas		384	384	200	184	1.92
<b>HITO</b>	<b>Se completaron todas las interfaces, paneles y popUps, así como también las mejoras anteriormente mencionadas.</b>					
<b>ID: 6.1.1 Mejora de las pruebas unitarias ATRIP y nivel de cobertura</b>		834	834	834	0	1
ID:RFPU1.0 Mejorar las coberturas		162	162	162	0	1
ID: RFPU1.1 Implementar las pruebas para las nuevas funcionalidades		672	672	672	0	1
<b>HITO</b>	<b>Se completaron todas las pruebas de las funcionalidades nuevas.</b>					
<b>ID: 6.1.2 Mejora de estandarización, diseño y patrones</b>		400	400	230	230	1.74
ID: RFC1.6 Disminuir la complejidad cognitiva		400	400	230	230	1.74

HITOS	Mejora de las estandarización del código					
<b>ID: 7.1 Diseño de prueba de testing detallado</b>		168	168	168	0	1
ID: RFT1.0 Planificación y Diseño de Pruebas		168	168	168	0	1
<b>HITOS</b>						
<b>ID: 7.2 Ejecución del set de pruebas de Testing</b>		894	894	751	143	1.19
ID: RFT1.1 ejecución del set de pruebas y clases de equivalencias		162	162	162	0	1
ID: RFT1.2 Ejecución del set de pruebas y partición de equivalencias		162	162	162	0	1
ID: RFT1.3 Ejecución y set de pruebas de valores límite		243	243	243	0	1
ID: RFT1.4 Ejecución y Set de Pruebas Testing exploratorio		243	243	100	143	2.43
ID: RFT1.5 Informe de Pruebas		84	84	84	0	1
<b>Hitos</b>	<b>Todos los informes de las nuevas pruebas realizadas de caja negra.</b>					
<b>ID: 8.1 Informe del proceso (logs)</b>		20	20	20	0	1
ID: RBD1.0 Realización de builder e informe de logs		20	20	20	0	1
<b>HITOS</b>						

<b>ID: 9 Informes de cierre</b>		<b>220</b>	<b>220</b>	<b>180</b>	<b>40</b>	<b>1.22</b>
ID: RIF1.0 Ajustes del plan de riesgo		40	40	10	30	4
ID: RIF1.1 Cambios de las líneas bases		20	20	10	10	2
ID: RIF1.2 Reporte de los cambios		20	20	20	0	1
ID: RIF1.3 Reporte estado de calidad final		100	100	100	0	1
ID: RIF1.4 Realización del informe de cierre		40	40	40	0	1
<b>HITOS</b>						
<b>SubTotal 06/12</b>		<b>2856</b>	<b>2856</b>	<b>2797</b>	<b>1358</b>	<b>31.96</b>

## 6. Situación del presupuesto del proyecto

Como se mencionó anteriormente en el informe de avance el presupuesto en esta etapa del proyecto fue de U\$S 3500 comparado a lo gastado en las actividades del proyecto fue de

U\$S 2797, pero esto sin pensar en el gasto de internet, alquiler y sueldos. Estimamos mal el costo de las actividades, nuestra estimación fue de U\$S 1800 pero el presupuesto abarcaba más de lo suficiente para poder realizar libremente en un rango normal las actividades sin exceder demasiado. Claramente las etapas de testing nos llevaron tiempo, muchos recursos y exhaustivos análisis, todo esto generó un gasto pero pudimos cubrirlo, y más aún estimamos más de lo que fue realmente gastado.

El precio total de las actividades eran: 7191,8\$. Claramente pudimos bajar todos estos costos, pero esto incluía los costos de alquiler, gastos de transporte de integración del sistema, sueldos, y gastos fijos de telefonía.

## 7. Situación de los riesgos identificados

No existieron nuevos riesgos identificados, sin embargo se agregó la acción de mitigación en caso de la no posible implementación de Java Fx y de las no posible realización de ciertas del proceso de mantenimiento. Estas dos ocurrieron, pero sobre todo no se pudo completar algunas actividades de calidad de código por miedo de generar nuevos problemas en ciertas funcionalidades, muchas de ellas se refactorizaron pero requerían cambios de diseño de codificación y implementación de patrones de diseño.

En caso de una tarea será necesario modificar la de actividades para agregar la tarea nueva y su estimación correspondiente de esfuerzo. Esto causara la modificación de los diagramas de cronograma (Gantt) agregando la tarea nueva, con su duración, y posponiendo las tareas que le siguen a esta. Además habría que reajustar el presupuesto inicial ya que ahora tenemos una nueva tarea que implica más horas de trabajo y más personal a disposición de esa nueva tarea.

En el caso de haber problemas con la implementación de Java Fx, se reestructura el diagrama Gantt, se habla con el equipo de diseñadores, y se deja estipulado el uso de Java Swing.

## 8. Evaluación de la calidad de los productos

Todas las actividades de mantenimiento que se realizaron para este informe de implementaciones de nuevas funcionalidades, les establecimos métricas de calidad.

**Corrección.** El software cumple con los objetivos principales del cliente, su forma de gestión de los alimentos, del historial, y de las comida ingeridas está realizado, sin embargo el software presenta algunas fallas sobre esto, el profesional tiene que revisar qué alimentos está consumiendo el usuario, el usuario que tipo de profesional es el encargado de su plan de alimentación, todo esto son cosas que los stakeholders pueden quedar insatisfecho.

Nivel de corrección: 60%

**Confiabilidad.** Este es un punto, en el que el programa cumple a raja tabla con las especificación, cumple un 100% sus objetivos sin irse de las ramas.

Nivel de confiabilidad: 90%

**Eficiencia.** Cantidad de recursos de cómputo y de código requeridos por un programa para llevar a cabo su función.

Nivel de eficiencia: 90%

**Integridad.** Gracias a la implementación de la interfaz login, y el identificador de una contraseña, este apartado tiene un óptimo nivel.

Nivel de Integridad: 89%

**Usabilidad.** En este ítem, como mencionamos anteriormente creemos que el usuario puede realizar el flujo normal, realizar las acciones correspondientes sin tener errores, sin que él precise ayuda en todo el momento sobre qué está haciendo el sistema, se le mantiene informado en cada momento, en cada acción.

Nivel de usabilidad: 90%

**Facilidad de recibir mantenimiento.** En este apartado fue cuando se nos llamó, fue muy difícil las correcciones de todos los errores que presenta el sistema, las tecnologías avanzadas, la calidad de código no era la más óptima y necesita verse para que no se degradara, la refactorización, el análisis más exhaustivo del código, y la aplicación de patrones de diseño más convencionales pueden llevar a que el sistema tenga una mejor facilidad de mantenimiento.

Nivel de facilidad de recibir mantenimiento: 75%

**Flexibilidad.** Con los cambios que hicimos, las implementaciones de los handlers de validaciones y gestión de logueo, quitar la responsabilidad de muchas cosas del sistema, podemos decir que la flexibilidad aumentó considerablemente.

Nivel de flexibilidad: 86%

**Portabilidad.** La portabilidad es bastante alta, comparando con el sistema inicial se siguió manteniendo, se cambia el nombre del archivo de serialización por repositorio, para manejar mejor con los tipos de datos que se quieren guardar.

Nivel de portabilidad: 86%

**Reusabilidad.** Este apartado fue muy importante para nosotros, se optó por cambiar un 50% de todos los paneles pero reusando partes existentes de los que estaban, podemos decir que la reusabilidad es bastante elevada y comparada con el estado inicial mucho más.

Nivel de reusabilidad: 90%

**Interoperabilidad.** La capacidad de acoplar el sistema con otro es bastante baja, básicamente existe una franja muy pequeña, de pasar a un sistema a un sistema legacy, justamente este apartado está en una franja crítica, y creo que se merece un análisis exhaustivo, y poner el énfasis en este ítem.

Nivel de interoperabilidad: 20%

## **9. Comentarios**

Este informe fue realizado basándonos en el informe de avance entregado anteriormente, y dejando abierta la posibilidad de que después del cierre del proyecto hecho por nosotros, el mismo pueda ser tomado por otro equipo de desarrollo, y que ellos realicen lo que nosotros no pudimos.

En este informe también se realizó las actividades necesarias para el cierre administrativo del proyecto que fueron:

- Auditar el éxito o fracaso del proyecto.
- Gestionar el intercambio y la transferencia de conocimiento.
- Identificar las lecciones aprendidas.
- Archivar la información del proyecto para su uso futuro por parte de la organización.