



**Universidad  
Andrés Bello**

**Proyecto “PET Manager”  
Seminario de Título 1  
Sebastián Inostroza Hurtado**

**INGENIERIA EN COMPUTACION E INFORMATICA**

**PROFESOR GUIA: PATRICIO CASTILLO**

**VIÑA DEL MAR - CHILE**

**10 DICIEMBRE 2018**

**FORMAR**

**TRANSFORMAR**



**Universidad  
Andrés Bello**

# Contenidos

<b>1. Introducción</b>	<b>5</b>
<b>2. Fundamentación del Problema</b>	<b>7</b>
2.1. Análisis de la situación actual.	7
2.2. Análisis de la problemática.	7
2.2.1. Técnica de los cinco por qué.	8
2.2.2. Técnica de Ishikawa.	9
2.3. Objetivos.	10
2.3.1. Objetivo general.	10
2.3.2. Objetivos específicos.	10
2.4. Diagrama de alto nivel.	11
2.5. Arquitectura de la solución.	12
Figura 5 - Diagrama arquitectura alto nivel V1.0	12
2.6. Solución ideal.	13
2.6.1. Alcances y Limitaciones.	13
2.6.2. Restricciones.	13
2.7. Alternativas de solución.	14
2.7.1. Procedimientos manuales.	14
2.7.2. Cambios en procedimientos actuales.	14
2.7.3. Alternativas disponibles en el mercado.	14
2.8. Solución propuesta.	15
2.8.1. Factibilidades.	15
2.8.2. Diseño de alto nivel.	16
2.8.3. Requerimientos de alto nivel.	19
<b>3. Planificación del proyecto.</b>	<b>21</b>
3.1. Metodología de desarrollo.	21
3.2. Planificación del proyecto.	21
3.3. Gestión de la configuración.	23
3.4. Gestión de cambios.	24
3.5. Gestión de riesgos.	25
Tabla 1.0 Riesgos de Proyecto	26
Riesgos técnicos	27
Tabla 1.1 Riesgos técnicos	27
Riesgos de negocio	28
Tabla 1.2 Riesgos negocio	28
3.6. Entorno de desarrollo.	29
<b>4. Resultados</b>	<b>31</b>

4.1. Análisis de resultados.	31
4.2. Diseño detallado.	39
4.3. Situación futura.	42
4.3.1. Resultados esperados.	43
<b>5. Conclusiones.</b>	<b>45</b>
5.1. Conclusión.	45
5.2. Problemas abiertos.	45
5.3. Trabajo futuro.	45

# Resumen

El año pasado se inauguró el primer centro especializado para la realización de los exámenes PET de la quinta región ubicado en Clínica Bupa Reñaca, ellos solicitan el radiofármaco al ciclotrón más cercano el cual que se encuentra ubicado en Santiago. El tiempo de despacho del radiofármaco es aproximadamente de 2 horas y varía mucho dependiendo del día y del tráfico, pues el tránsito es vía terrestre con límites de velocidad establecidas por ley, por lo que no hay una hora segura de llegada.

La Empresa del ciclotrón se compromete a entregar a las 10:30 am un mínimo de 110 mci (milicuries) valor el cual puede fluctuar para más o menos su valor, pues no siempre el radiofármaco llega a tal hora. Además puede existir la posibilidad de solicitar mayor cantidad de dosis para poder realizar mayor número de pacientes.

Es imprescindible tener un conteo específico del radiofármaco para poder organizar la agenda de pacientes y así poder garantizar que los pacientes citados se podrán realizar.

Actualmente se trabaja con un activímetro que mide la radiación que llega, y que se dosifica para ser inyectada a cada paciente. Esta manipulación de la dosis es realizada por los Tecnólogos Médicos, que deben ir calculando la dosis que se les inyecta a cada paciente e ir dosificando de mejor manera el Radiofármaco, con el fin de que cada uno de éstos pueda recibir la dosis adecuada. En una Pizarra se anota la Hora de Inyección del fármaco al paciente, la hora en la cual el paciente debe ingresar al examen , estos tiempos de deben adecuar a tiempos entre pacientes y del uso de la máquina PET.

El problema actual radica en que para tener un conteo preciso de radiofármaco, se deben calcular muchas variables que dificultan la fluidez del trabajo, además para tener un conteo preciso se debe medir el frasco de origen cada vez para ir chequeando que los valores de radiación serán adecuados para los pacientes, siendo al final del día un poco incierto la capacidad de hacer o no un paciente extra o incluso un paciente programado.

Ante esto, la necesidad de crear una aplicación que permita gestionar los tiempos de trabajo y entregue un valor de dosis sin necesidad de exponer al personal más de lo adecuado, es algo imperante en este servicio.

# Capítulo 1: Introducción

## 1. 1. Introducción

El PET (Tomografía por Emisión de Positrones ) es una técnica de diagnóstico no invasiva que permite realizar imágenes que muestran el metabolismo y el funcionamiento de tejidos y órganos, basándose en el consumo de glucosa.

Todos los tejidos en el cuerpo humano consumen glucosa, pero aquellos en los que se está produciendo una proliferación maligna consumen mucho más. Inyectando vía endovenosa al paciente una solución de glucosa con un marcador radioactivo (pero inocuo) y pasándolo por el PET, éste recoge la radiación que en ese momento emite.

Esa radiación permite al sistema operativo del PET dibujar un mapa del cuerpo en el que se recogen, si las hubiera, las mayores concentraciones de glucosa, de modo que el médico puede determinar la malignidad de un bulto ya conocido o incluso detectar metástasis en sus grados más incipientes.

El PET se utiliza fundamentalmente para comprobar la malignidad de los tumores. Sucede que las células tumorales, por su metabolismo oxidativo, necesita más glucosa. Para detectar dónde están o cómo actúan, se introducen radiofármacos como la fluorodesoxiglucosa, o f18-fdg, que permite detectar casi todos los tumores y cuya radiación decae a la mitad en un tiempo de 109,77 minutos aprox.

Las sustancias radiactivas se desintegran con mayor o menor rapidez, según de cual se trate. Se llama vida media al tiempo requerido para que la mitad de una sustancia desaparezca. Por ejemplo, la vida media del Radio es de 1620 años. De esta forma, si tenemos 500gr. de Radio, al cabo de 1620 años habrá sólo 250gr. 1620 años después, es decir dentro de 3240 años, habrá 125gr. Y así sucesivamente. A lo largo de cada semiperiodo, la radiactividad desciende primero a la mitad, luego a una cuarta parte, etc.

La mayoría de los radionucleidos emisores de positrones presentan vidas medias extremadamente cortas, lo cual necesita su producción cerca del sitio de utilización. Todos son producidos por ciclotrón en vez de reactor. Los más usados son el Flúor-18, el Carbono-11, el Oxígeno-15 y el Nitrógeno-13, todos los cuales son capaces de marcar moléculas orgánicas fisiológicamente utilizadas por el metabolismo celular. La única excepción para la necesidad de producción cerca del sitio de aplicación es el Flúor-18, el que por su mayor vida media puede ser usado en lugares distantes 1 – 2 horas del ciclotrón.

Para PET es la necesidad de una química automatizada de alta velocidad para asegurar que el marcado se pueda lograr con mínima exposición radiactiva de los operadores y una pérdida aceptable de actividad por decaimiento antes de ser administrada al paciente.

## Capítulo 2: Fundamentación del problema

## 2. Fundamentación del Problema

### 2.1. Análisis de la situación actual.

Actualmente, en clínica Bupa Reñaca, se realizan 5 pacientes confirmados diariamente, teniendo una lista de espera de 2 semanas para pacientes ambulatorios. Los pacientes urgentes u hospitalizados son incorporados dentro de estos 5 pacientes, pero bajo condiciones excepcionales se acepta como 6to paciente teniendo que tener extremado cuidado con la dosis a repartir para los pacientes, ocupando el mínimo para “asegurarnos” de que alcanzara la dosis al final de la jornada. Esta mínima cantidad influye con la calidad del examen, debiendo realizar un examen de mayor duración para compensar esta falta de dosis y al hacer el examen de mayor duración va repercutiendo en los pacientes futuros, pues la dosis para éstos va disminuyendo, creando una reacción en cadena, el cual concluye en que al final de la jornada puede sobrar o faltar radiofármaco y por ende no realizar el 6to paciente. La forma de asegurarnos de que podemos realizar cierta cantidad de pacientes es midiendo entre cada paciente la actividad (dosis) que va quedando en el contenedor principal y poder realizar un fraccionamiento correcto de las dosis entre los pacientes.

### 2.2. Análisis de la problemática.

Para identificar la problemática utilizamos la técnica de los 5 ¿Por qué? de forma extendida, buscando las posibles causas y el diagrama de Ishikawa que denotara de mejor forma los actores principales.

***“El número de pacientes diarios a examinar es impreciso ocupando las técnicas de trabajo actual y medidas de radioprotección”.***

## 2.2.1. Técnica de los cinco por qué.

Problema	Porque? 1	Porque? 2	Porque? 3	Porque? 4	Porque? 5	Solución
Imprecisión en numero de pacientes diarios	1- No hay certeza de la actividad del radiofármaco	1.1- No se realizan cálculos.	1.1.1- Cálculos Engorrosos	1.1.1.1- Muchas variables		Herramienta generadora de cálculos
			1.1.2- Toman mucho tiempo	1.1.2.1- Repetitivas.		Herramienta generadora de cálculos
		1.2- Medir la actividad del frasco contenedor	1.2.1- El fármaco emite radiación constantemente	1.2.1.1- Para Irradiar lo menos posible al operador	1.2.1.1.1- Leyes de radio protección.	Evitar Exposición
					1.2.1.1.2- Es perjudicial	
	2-No hay certeza de cuanta dosis llegara	2.1-El radiofármaco viene de Santiago	2.1.1-Producción solo en Santiago	2.1.1.1- Alto costo y mant.		X
			2.1.2- Transporte vía terrestre	2.1.2.1-Ley de Transporte		
	3-Pacientes pueden ser complejos o con mórbidos importantes	3.1- Mal acceso Venoso	3.1.1-Condición del Paciente			X
		3.2- Glicemia Elevada	3.2.1-Condición del Paciente			X
		3.3-Tener Mórbidos	3.3.1-Condición del Paciente			X
	4-Exámenes pueden tener mayor duración	4.1- Patologías específicas	4.1.1- Para estudio correcto			X
		4.2- Menos dosis al paciente	4.2.1-No se dosifico la dosis	4.2.1.1- No se realizaron cálculos.		Herramienta info. en tiempo real
		4.3- complicaciones durante el examen	4.3.1- Condición del Paciente			X

Figura 1 - Cuadro 5 Porques.



## 2.2.2. Técnica de Ishikawa.

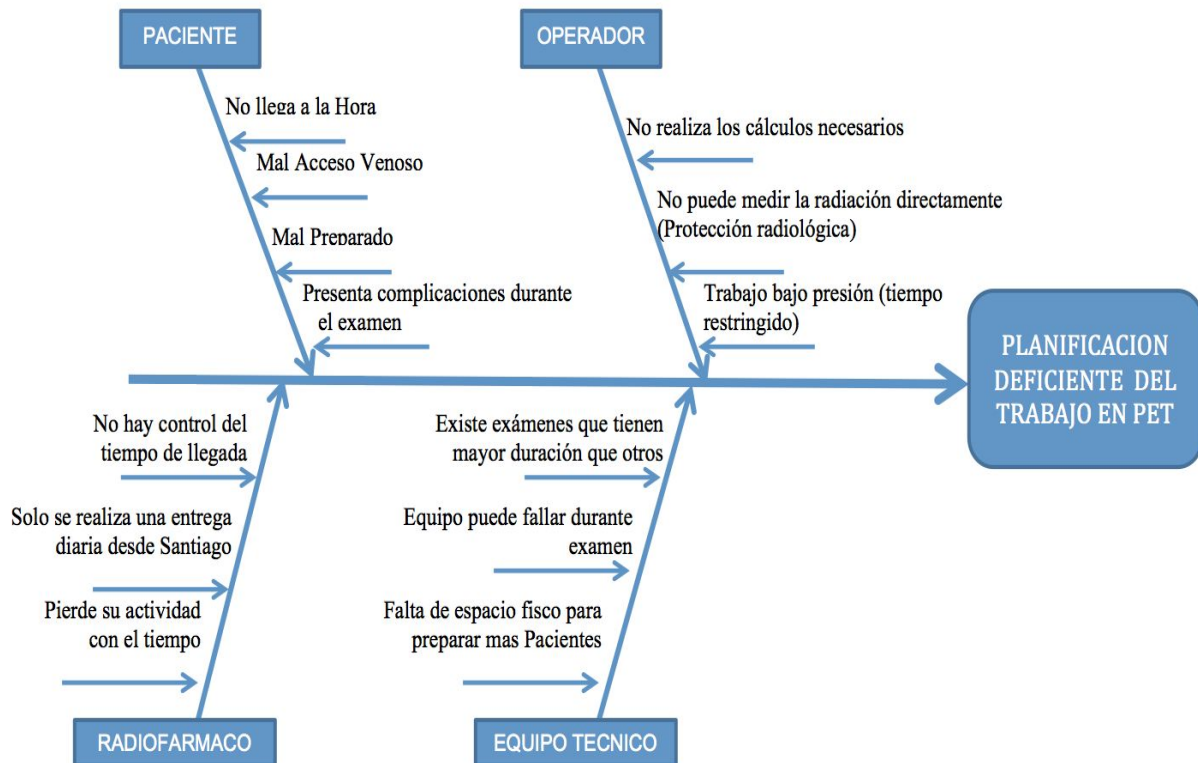


Figura 2 - Diagrama Ishikawa

## 2.3. Objetivos.

### 2.3.1. Objetivo general.

- “Tener de manera rápida y sencilla los valores de Actividad radiactiva del radiofármaco en cualquier momento de la jornada, sin necesidad de irradiar de manera adicional al personal Operador (Tecnólogos Médicos)”.

### 2.3.2. Objetivos específicos.

- 1. Lograr que el valor calculado tenga un margen de error de no más de 5% con el Radiofármaco.
- 2. Disminuir la Dosimetría de los operadores manipuladores de dosis en un 5% a los meses de su implementación.
- 3. Aumentar un 5% el número de pacientes totales a realizar durante el primer semestre de su implementación.
- 4. Generar una visualización de la actividad radiactiva en tiempo real con una actualización de 1 minuto y una resolución menor a 1 segundo.
- 5. Generar una visualización del cálculo de la actividad del radiofármaco en un tiempo específico de la jornada, con una resolución menor a 1 segundo.
- 6. Lograr un cálculo de cantidad de ml con un margen de error de no más de 20% para cierta radiactividad requerida, Ej. se requiere una dosis de 8 mCi, según el cálculo se debe sacar del contenedor 1,5 mL para ese momento específico de la jornada.

Objetivo Especifico	Métrica	Unidad	Criterio de Éxito	Método
1	Margen de Error	%mCi	< 5	Comparar medición con activimetro
2	Disminuir Dosimetría	%mCi	> 5	Informes Dosimétricos CCHEN
3	Aumentar Producción	% Exámenes	> 5	Estadísticas producción Clínica Reñaca
4	Resolución calculo actual	Segundo	< 1	Medidor de resolución de cálculos
5	Resolución calculo proyectado	Segundo	< 1	Medidor de resolución de cálculos
6	Margen de Error	%mCi	< 20	Comparar Actividad dosificada con activimetro

Figura 3 - Cuadro de Objetivos específicos

## 2.4. Diagrama de alto nivel.

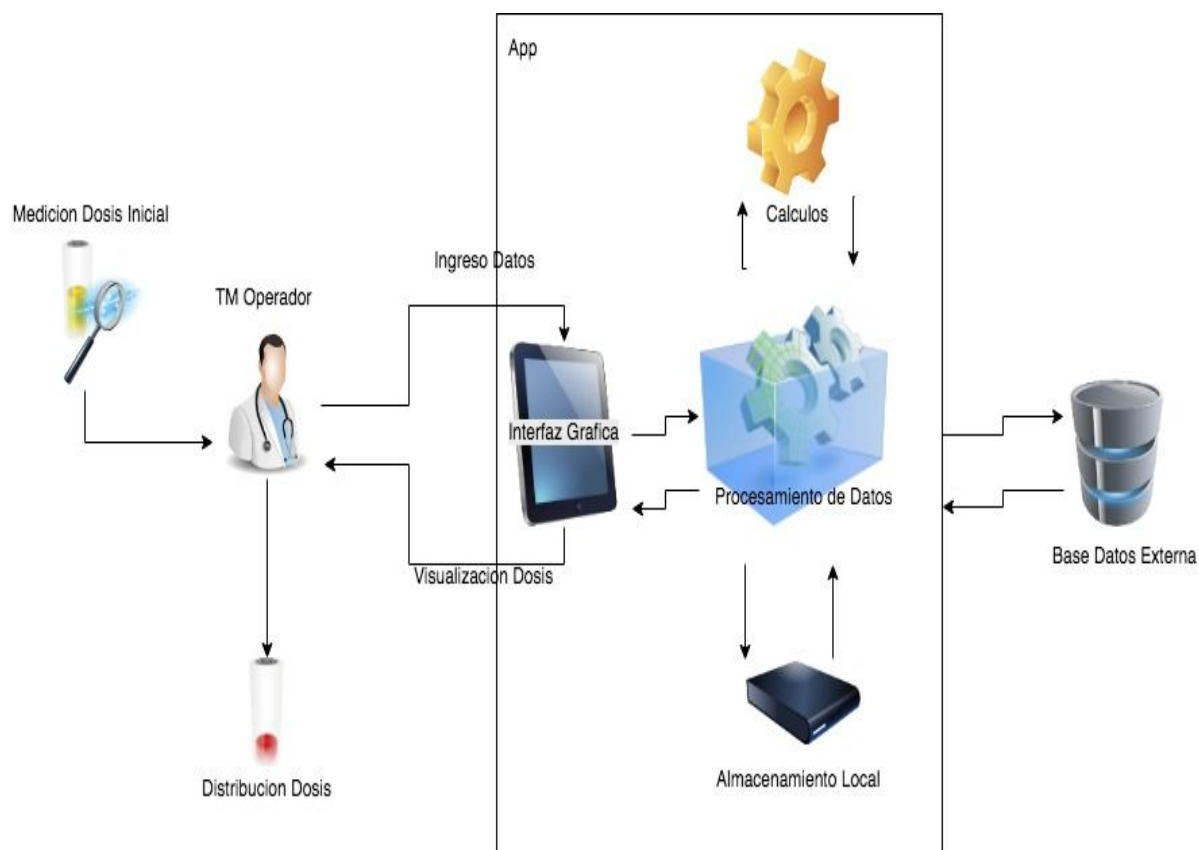


Figura 4 - Diagrama Alto nivel

## 2.5. Arquitectura de la solución.

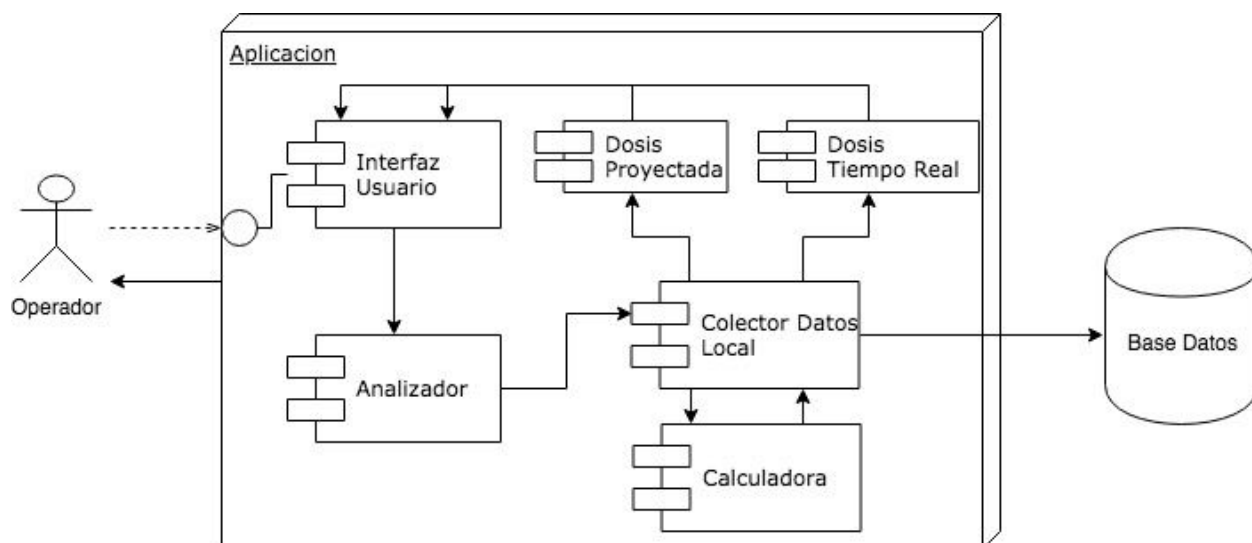


Figura 5 - Diagrama arquitectura alto nivel V1.0

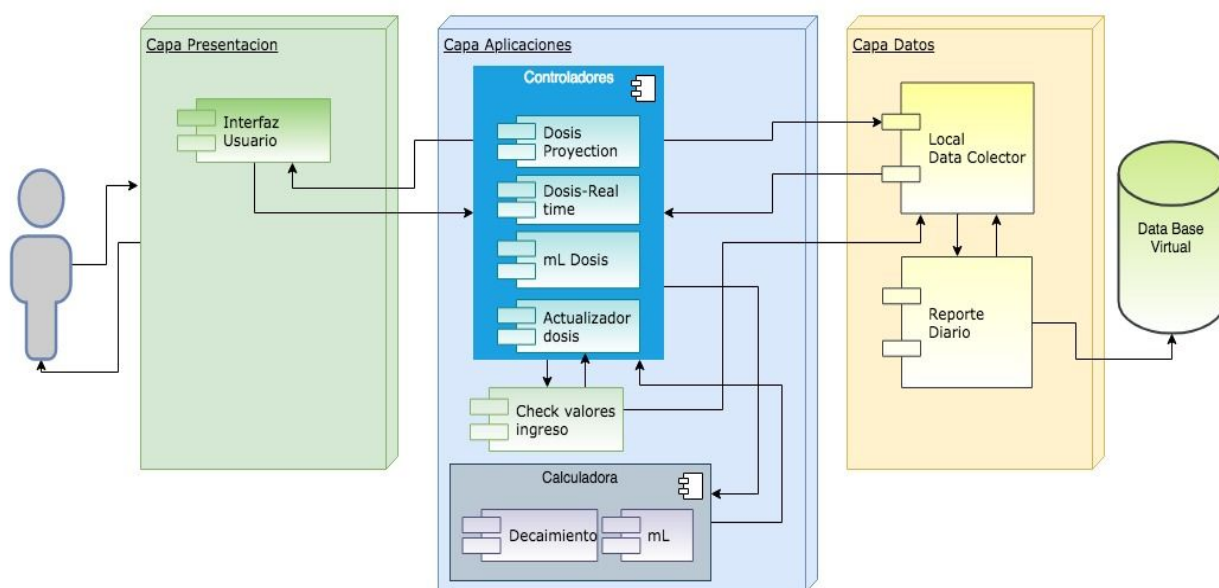


Figura 6 - Diagrama de arquitectura de alto nivel V1.1

## **2.6. Solución ideal.**

- Contar con un sistema robotizado de gestionamiento de dosis, que permita manipular y dosificar la dosis de manera inteligente y permita optimizar los recursos y el tiempo para cada examen.

### **2.6.1. Alcances y Limitaciones.**

- La Aplicación no será capaz de tomar los datos de manera automática.
- El ingreso de datos para la aplicación es manual y depende del operador.
- La Aplicación sólo podrá ser manipulada por los Tecnólogos Médicos capacitados.
- Para este proyecto, la Aplicación será desarrollada sólo para un tipo de dispositivo y sistema operativo.
- Para este proyecto, los reportes se enviarán como archivos a una ubicación predeterminada, dado que todavía no se cuenta con una base de datos.
- Para este proyecto, no se entregará información de Gestionamiento de pacientes, sólo datos calculados.

### **2.6.2. Restricciones.**

- No existe actualmente un plan de financiamiento para desarrollo de aplicaciones médica en Clínica Bupa Reñaca fuera del departamento informático.
- Se deberán respetar las leyes internas de manejo de información médica dentro del recinto.
- El Grupo de Trabajo es mínimo, por lo que los tiempos de ejecución para el desarrollo de la aplicación serán elevados.
- Para la utilización la aplicación deberá pasar por el comité de Ética de Clínica Bupa Reñaca.
- El riesgo de no cumplir con los objetivos propuestos en los tiempos estipulados generará desmotivación con el cliente y stakeholders, además de hacer menos plausible la asignación de recursos.

## 2.7. Alternativas de solución.

### 2.7.1. Procedimientos manuales.

- 1- Los cálculos pueden realizarse de manera manual, o estimar valores para gestionar la dosis en los pacientes.
- 2- Disminuir la calidad del examen ocupando el mínimo de radiofármaco por cada paciente.

### 2.7.2. Cambios en procedimientos actuales.

- 3- Aumentar el número de operadores y dejar destinado a uno que sólo se preocupe de la distribución y cálculos de dosis.
- 4- Pedir mayor cantidad de radiofármaco.
- 5- Instalar un Ciclotrón en la quinta región que realice despachos locales de radiofármaco.

### 2.7.3. Alternativas disponibles en el mercado.

- 6- Software y App Calculadoras de dosis disponibles que sólo realizan un cálculo básico de radiactividad inicial y final.
- 7- Activímetro Posee software que realiza un cálculo de decaimiento que no es posible aplicar a una agenda estructurada, al igual que las calculadoras solo se pueden realizar cálculos iniciales y finales.

Soluciones	Criterios						Valoración de la Solución
	Costos Implementación	Tiempos de Implementación	Impacto Negativo Productividad	Problemas Factibilidad técnica	Impacto Negativo Calidad actual	Dificultad cobertura Problema	
Nº1	3	1	1	2	1	1	9
Nº2	3	3	2	3	1	1	13
Nº3	1	2	3	1	3	3	13
Nº4	1	3	3	3	2	2	14
Nº5	1	1	3	1	2	2	10
Nº6	1	2	2	2	2	1	10
Nº7	2	1	2	2	2	1	10

Alta = 1  
Medio = 2  
Baja = 3

Figura 7 - Cuadro comparativo alternativas de solución

## 2.8. Solución propuesta.

- Creación de sistema capaz de realizar cálculos de decaimiento radiactivo para Flúor F-18.
- La aplicación será efectiva y eficiente, en términos de conducir a pocos errores y requerir poco tiempo de uso por parte del usuario.
- Herramienta será utilizada por Tecnólogos Médicos encargados de PET.
- Tendrá una Entrada de Datos (Actividad, Hora, mL).
- Generará Salida de Datos (Actividad) en tiempo real y proyectados a cualquier momento de la jornada de trabajo.
- Generará reportes de Distribución de dosis de la jornada para el día de trabajo.

### 2.8.1. Factibilidades.

Podemos clasificarlas en desde tres perspectivas:

#### Económica

- No se cuenta con recursos asignados, pero si los resultados son positivos podría abrirse financiamiento, por lo tanto el proyecto sería económicamente factible. Falta evaluación detallada.

#### Técnica

- No se cuenta con un equipo dedicado para la aplicación, pero se considera suficiente utilizar uno de los equipos de ofimática disponibles, resguardando el uso de recursos.

#### Operativa

- La Aplicación sólo podrá ser manipulada por los Tecnólogos Médicos capacitados.

## 2.8.2. Diseño de alto nivel.

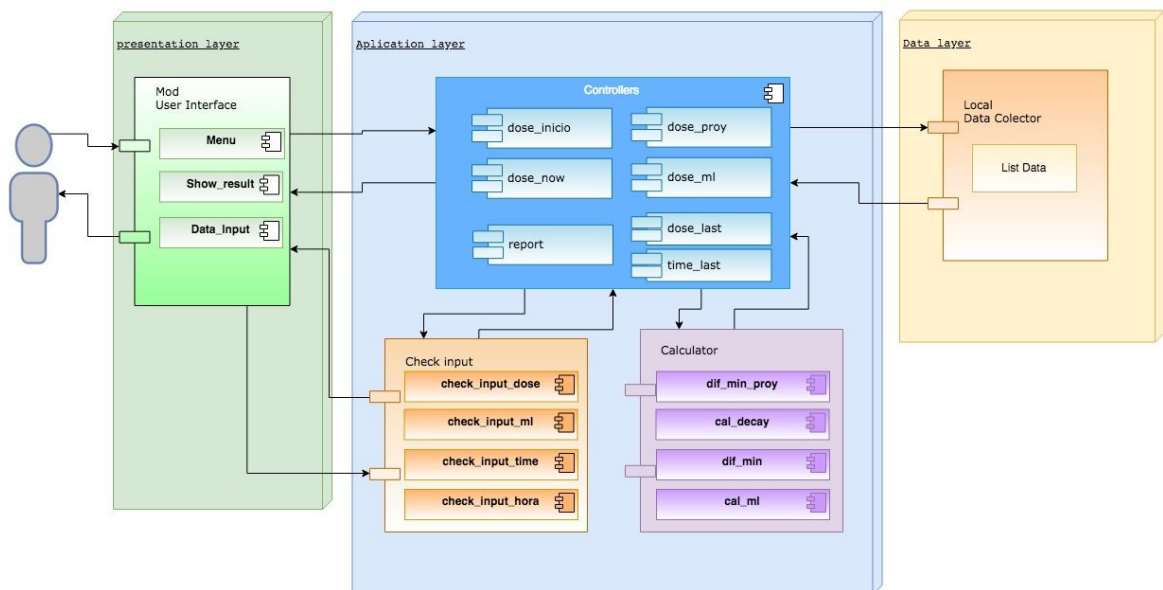


Figura 8 - Diagrama de alto nivel de la aplicación.

Algunos de los módulos principales a implementar son:

- Módulo **Cal\_decay**:
  - Implementación: Python 3.7.1
  - Librería gráfica: por definir
  - Base de datos: local
  - Objetivo: Realizar el cálculo del decaimiento radiactivo de un radiofármaco.
  - Datos de Entrada:
    - Requiere el valor numérico de Dosis inicial medida en milicuries (mCi) con dos decimales Ej 180.39
    - Requiere el valor entero de tiempo en Minutos (min) que transcurre desde la actividad inicial a la hora de medida requerida Ej 55.
  - Datos de Salida:
    - Valor de Decaimiento radiactivo de flúor en milicuries (mCi) de FDG en el periodo de tiempo establecido.

El Módulo cal\_decay está relacionado con cualquier módulo que solicite un cálculo de decaimiento radiactivo para el flour.



- Modulo **Check\_input:**

- Implementacion: Python 3.7.1
- Librería gráfica: por definir
- Base de datos: local
- Objetivo: Realizar filtrado de datos erróneos ingresados por el usuario.
- Datos de Entrada:
  - def check\_input\_dose / Ingreso de valor tipo float.
  - def check\_input\_ml / Ingreso de valor tipo float.
  - def check\_input\_time / Ingreso de valor tipo int para ingresar cantidad de minutos.
  - def check\_input\_hora / Ingreso de valor tipo str que transforma a estructura de tiempo, para luego filtrar en formato HH:MM .
- Datos de Salida:
  - def check\_input\_dose / retorna valor tipo float limitado a 2 decimales.
  - def check\_input\_ml / retorna valor tipo float.
  - def check\_input\_time / retorna valor entero.
  - def check\_input\_hora / retorna valor formato HH:MM .

El módulo Check\_input está relacionado con cualquier módulo que implique interacción del usuario para el ingreso de datos.

- Modulo **dif\_min:**

- Implementacion: Python 3.7.1
- Librería gráfica: por definir
- Base de datos: local
- Objetivo: Calcular los minutos que han transcurrido desde una hora determinada a la hora actual.
- Datos de Entrada:
  - Hora determinada en formato HH:MM
- Datos de Salida:
  - Minutos en formato Int.

Este Módulo se provee los minutos para distintos módulos de cálculos, se relaciona directamente con el módulo de dose\_now, módulo Local\_data\_colector, dose\_last y time\_last.

- Modulo **dif\_min\_proy**:
  - Implementacion: Python 3.7.1
  - Librería gráfica: por definir
  - Base de datos: local
  - Objetivo: Calcular los minutos que existen entre dos horas determinadas.
  - Datos de Entrada:
    - Hora determinada en formato HH:MM
  - Datos de Salida:
    - Minutos en formato Int.

Este Módulo se provee los minutos para distintos módulos de cálculos, se relaciona directamente con el modulo de dose\_proy, modulos check\_input.

### 2.8.3. Requerimientos de alto nivel.

#### Requerimientos Funcionales:

1. Realizar cálculos de decaimiento radiactivo para el Radiofármaco Flúor 18 FDG.
2. Permitir ingresar datos de Dosis inicial en mCi, mL y hora.
3. Mostrar el estado de decaimiento radiactivo actual.
4. Permitir ingresar datos de Dosis suministradas en mL y hora.
5. Mostrar cálculos de Dosis en mCi, Tiempo en formato Hora:minutos y volumen en mL.
6. Generar reporte diario con las Dosis utilizadas para cada examen realizado.
7. Presentar menú de selección de alternativas para las distintas tareas.

#### Requerimientos No Funcionales:

- La aplicación debe ser intuitiva y fácil de utilizar.
  - Se evaluará que el tiempo de aprendizaje por un nuevo usuario sea menor a 3 horas.
- Se debe resguardar las identidades de los pacientes.
  - A través de la aplicación y sus reportes no se puede acceder a información de pacientes.
- La aplicación debe ser liviana para un computador de ofimática de bajo costo.
  - Funcionará en Windows 7 32 bits, con menos de 500 MB de memoria y 2 GB de disco.
- Mostrar dosimetría actual de forma fácil de visualizar y entender.
  - Un usuario reconocerá la información ubicándose al menos a 3 metros de la pantalla.
- Permitir enviar reportes por correo de forma sencilla.
  - Se evaluará enviando un reporte sin requerir software especializado adicional.
- Mostrar mensajes de error informativos y orientados al usuario final.
  - Un usuario debe ser capaz de interpretar al menos 3 tipos de mensajes posibles.

## Capítulo 3: Planificación del proyecto

### 3. Planificación del proyecto.

#### 3.1. Metodología de desarrollo.

Metodología Ágil : SCRUM - Sprint cada 2 Semanas.

##### Roles

**Cliente:** Felipe Vera.  
**Product owner:** Patricio Castillo.  
**Scrum master:** Sebastián Inostroza  
**Equipo Scrum:** Sebastián Inostroza

#### 3.2. Planificación del proyecto.

Sprint cada 2 semanas.

Estimación: Hito 3 - Entrega - Mockup Básico Funcional.

Listo	Nombre de la tarea	Fecha de Inicio	Fecha final
<input type="checkbox"/>	Seminario Titulo 1	11/06/18	01/07/19
<input checked="" type="checkbox"/>	Reunion 1	11/06/18	11/06/18
<input checked="" type="checkbox"/>	Reunion 2	11/20/18	11/20/18
<input type="checkbox"/>	Sprint 1	11/26/18	12/10/18
<input type="checkbox"/>	Sprint 2	12/10/18	12/24/18
<input type="checkbox"/>	Entrega Prototipo	01/04/19	01/04/19
<input type="checkbox"/>	Sprint 3	12/24/18	01/04/19
<input type="checkbox"/>	Hito 1	11/16/18	11/16/18
<input type="checkbox"/>	Hito 2	12/07/18	12/07/18
<input type="checkbox"/>	Hito 3 - Mockup	01/04/19	01/04/19

Figura 9 - Cuadro planificacion proyecto.

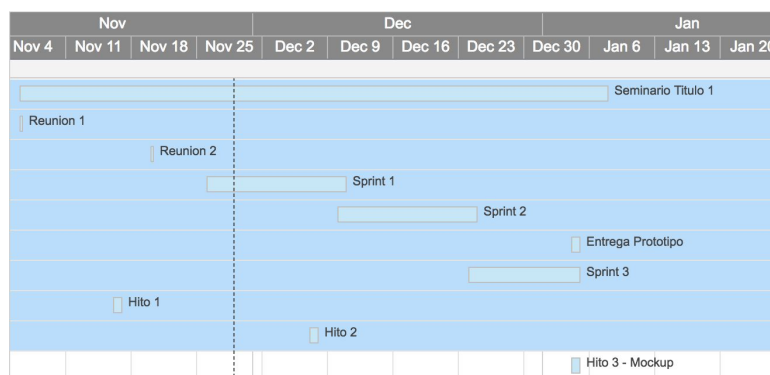


Figura 10 - Cuadro planificacion proyecto desplegado.

- Requerimientos Propuestos vs. Cumplimiento por Sprint.
1. Realizar cálculos de decaimiento radiactivo para el Radiofármaco Flúor 18 FDG.
  2. Permitir ingresar datos de Dosis inicial en mCi, mL y hora.
  3. Mostrar el estado de decaimiento radiactivo actual.
  4. Permitir ingresar datos de Dosis suministradas en mL y hora.
  5. Mostrar cálculos de Dosis en mCi, Tiempo en formato Hora:minutos y volumen en mL.
  6. Generar reporte diario con las Dosis utilizadas para cada examen realizado.
  7. Presentar menú de selección de alternativas para las distintas tareas.

Backlog	Sprint 1	Sprint 2	Sprint 3	Metrica %
R1	50	80	100	Generar el cálculo de decaimiento y comparar con valor real in situ
R2	20	80	100	Permitir el ingreso de datos iniciales a través de interfaz gráfica en formatos correspondientes sin errores
R3	20	80	100	Generar y mostrar el calculo a traves de vista en la aplicación
R4	20	80	100	Permitir el ingreso de datos a través de interfaz gráfica en formatos correspondientes
R5	10	100	100	Visualizar resultados en los formatos especificados en la vista de resultados
R6	10	50	100	Archivo que registre las dosis iniciales y sus modificaciones a los largo de la jornada
R7	5	70	100	Generar menú que permite navegar por aplicación en vistas interfaz gráfica

**Figura 11 - Tabla planificación requisitos del proyecto vs sprint.**

### 3.3. Gestión de la configuración.

#### Gestión de líneas base

Cada vez que se finalice una etapa será liberada una línea base del proyecto.

Cada vez que se termine un Sprint y este sea validado tanto por el Scrum Master como por el cliente será liberada una línea base del proyecto.

#### Presentación a evaluaciones (Hitos)

Durante el desarrollo del proyecto, se realizarán presentaciones ante una comisión para evaluar el avance del proyecto, para este propósito será liberada una línea base del proyecto.

Cada línea base que posea el proyecto será documentada en el repositorio de la aplicación de github en la carpeta "docs".

Los Archivos serán:

- Presentación (PPTX, PDF, etc) .
- Documento de aceptación (txt, DOC, PDF, XLSX, etc).

Los Archivos serán subidos a un repositorio en GITHUB con direccion:

<https://github.com/Sebasinos/ProyectoPET.git>

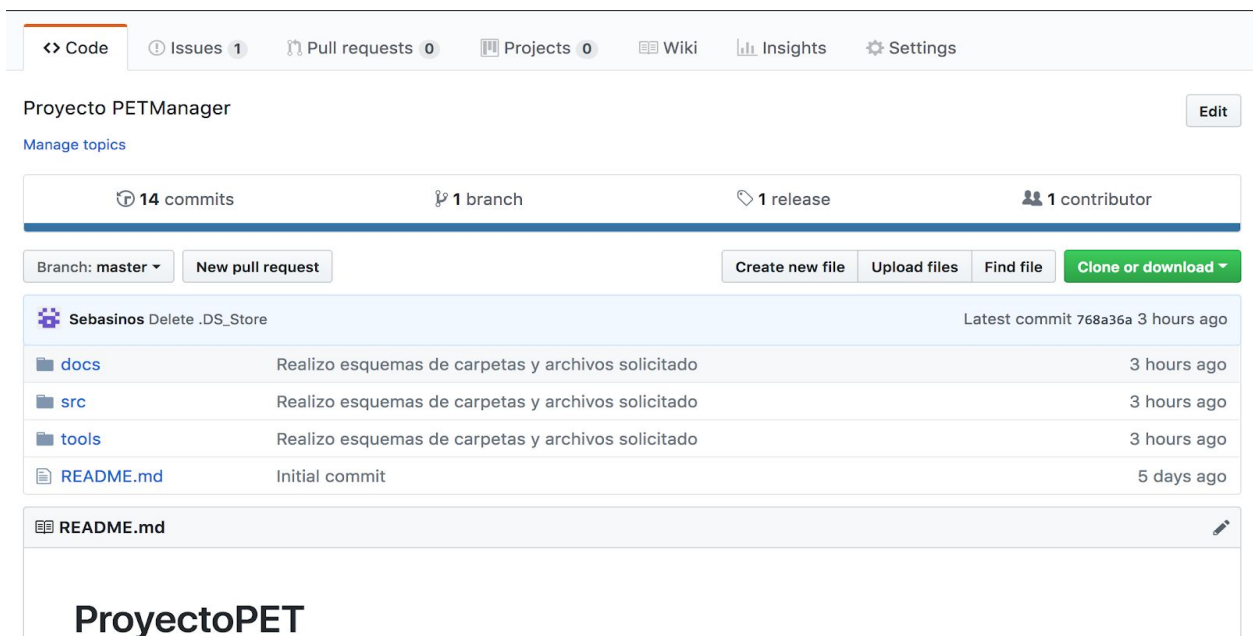


Figura 12 - Visualización del repositorio Github.

### 3.4. Gestión de cambios.

La Gestión de cambios será realizada con la herramienta GIT - “PullRequest”, que permite llevar un registro de todos los cambios que se han realizado en el repositorio del proyecto. Los Commits serán obligatorios cada vez que se finalice el sprint planificado, no impidiendo realizar avances o actualizaciones en cualquier momento de la duración del proyecto.

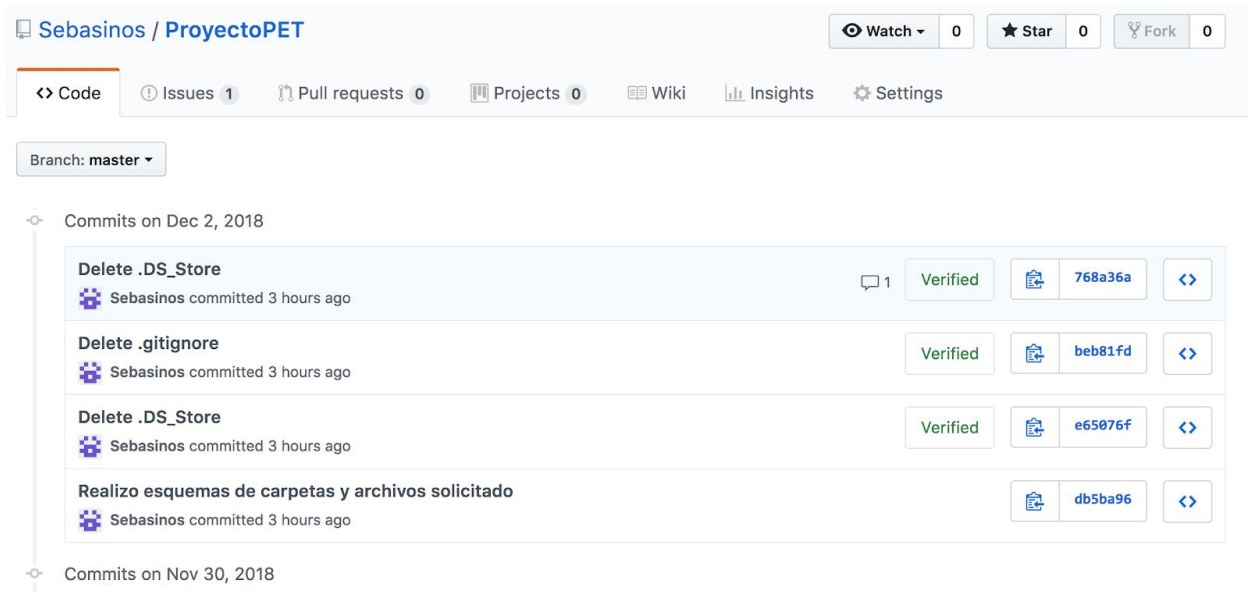


Figura 13 - Visualización de cambios en GITHUB



### 3.5. Gestión de riesgos.

Los riesgos serán clasificados de la siguiente manera:

- Riesgo de proyecto.
- Riesgo técnico.
- Riesgo de negocio.
- Riesgos específicos.

Después de ser clasificados se evaluará el impacto del riesgo en una escala de 0 a 1. Donde 0 es muy bajo y 1 es muy alto impacto.

Además de determinar el impacto del riesgo, hay que agregar una ponderación de probabilidad de ocurrencia que se evaluará en una escala de 0 a 1. Donde 0 es muy baja y 1 es muy alta probabilidad. El resultado nos dará una ponderación de amenaza del riesgo.  
 $\text{Porcentaje de amenaza} = (\text{Probabilidad} * \text{impacto del riesgo}) * 100$   
Cada riesgo incluye un plan de mitigación, un plan de contingencia y el sprint cuando se gatilla el riesgo.

A continuación se mostraran los riesgos identificados del proyecto. Cabe destacar que la identificación de riesgos es una tarea constante a lo largo de todo el proyecto.

#### Riesgos de Proyecto

ID	Riesgo	Pro bab ilid ad	Im pa cto	% de am ena za	Mitigación	Contingencia	Fecha Sprint Relacio nado
RP1	Incumplimiento de fechas	0,8	0,2	16	Cumplir con la planificación del proyecto rigurosamente	Aumentar las horas diarias de trabajo	10-12-18

RP2	Los Requerimientos cambian en cada Sprint	0,8	1	80	Establecer el Alcance junto con el cliente y respetar los ítems acordados	Por cada Cambios se deberá analizar la factibilidad y si es factible modificar los plazos	
RP3	Mala planificación de sprint	0,6	0,4	24	Planificar el sprint con más riesgo primero	Re-planificación de sprint	
RP4	Mala estimación de sprint	0,7	0,5	35	Usar técnicas de estimación validadas	Re-estimar los sprint	
RP5	Plataforma de gestión de la configuración no está disponible, se cae.	0,2	0,1	2	Llevar una gestión de la configuración de forma manual.	Usar gestión de la config. Manual	
RP6	Plataforma de control de versiones no está disponible, se cae.	0,2	0,1	2	Llevar un control de versión local adicional.	Usar el control de versión local	
RP7	Requerimientos no cumplen con las expectativas del cliente	0,5	0,8	40	Realizar reuniones semanales para recibir feedback	Aumentar las HH para modificar aquellos requerimientos	

**Tabla 1.0 Riesgos de Proyecto**

## Riesgos técnicos

ID	Riesgo	Pr ob abi lid ad	Impa cto	% de ame naza	Mitigación	Contingencia	Sprint Relaciona do
RT 1	Un mal diseño de software implica volver a diseñar e implementar	0,5	0,6	30	Evaluación periódica del modelo actual	Realizar un nuevo diseño, verificarlo validarlo y volver a implementar	24-11-18
RT 2	Metodología no acorde al proyecto	0,1	0,4	4	Evaluar la mejor metodología	Adaptar nueva metodología en base al resto de la planificación	
RT 3	Inexperiencia en el lenguaje de programación o herramientas	0,8	0,9	72	Aumentar las horas de trabajo diarias dedicada al auto aprendizaje	Aumentar las horas diarias de trabajo	25-11-18
RT 4	Complejidad de herramienta no permite al equipo de desarrollo avanzar	0,7	0,7	49	Realizar investigación, estudiar libros o ver tutoriales, documentaciones.	Cambiarla herramienta de desarrollo.	9-12-18

**Tabla 1.1 Riesgos técnicos**

## Riesgos de negocio

ID	Riesgo	Pro babi lida d	Impa cto	% de ame naza	Mitigación	Contingencia	Fecha Sprint Relacionado
R N1	Directivos rechazan implementación de aplicación	0,2	0,8	16	Abordar las causas que provocaron rechazo.	Realizar un nuevo estudio e implementar las mejoras	04-01-19
RN 2	Incumplimiento de fecha de entrega de la aplicación.	0,5	0,8	40	Aumentar el trabajo de cada sprint, solicitar reuniones extras para cumplir con los requerimientos de stakeholders	Llegar a acuerdo con directivos por nuevos plazos	04-01-19
RN 3	Fecha de presentación de la aplicación no definida por parte del cliente	0,7	0,2	14	Coordinar con cliente fecha para presentación formal de la aplicación antes de finalizar el proyecto.	Establecer fecha de presentación y documentar en proyecto antes del sprint 1	25-11-18

**Tabla 1.2 Riesgos negocio**

## Riesgos específicos Orientados a programación

ID	Riesgo	% de amenaza	Mitigación	Contingencia	Fecha Sprint Relacionado
1	No se pueda realizar Cálculo de decaimiento en la aplicación.	30	Capacitar al team en programación de fórmulas físicas del lenguaje de python	Aumentar las horas diarias de trabajo y aumentar las herramientas de aprendizaje	26-11-18
2	Datos ingresados generen error en la aplicación	30	Dar instrucciones específicas de ingreso de datos a los usuarios	Crear herramienta controladora de datos	28-11-18
3	No conocer herramientas que ayuden a calcular formatos de tiempo	72	Investigar sobre herramientas y librerías de python para trabajo con formatos de tiempo	Aumentar las horas de autoaprendizaje - visualización de foros y videos	30-11-18
4	Generar herramientas para orden en el código y realización de pruebas	49	Tener orden de archivos y control en repositorio	Crear archivo que implemente todos los módulos para pruebas	1-12-18

**Tabla 1.3 Riesgos específicos**

### 3.6. Entorno de desarrollo.

El ambiente de desarrollo del producto software será el computador personal del desarrollador.

Hardware  
PC

Modelo	MacBook Pro (Retina, 13-inch, Late 2013)
Procesador	2,4 GHz Intel Core i5
Memoria	8 GB 1600 MHz DDR3
S.O	OS X El capitan Version 10.11.6 (15G22010)

**Tabla 2.0 Hardware**

Herramientas de construcción

Tipo	Nombre
Ambiente de desarrollo integrado/Compilador	IDLE Python
Lenguaje de programación	Python 3.7.1
Gestor Grafico	Por definir

**Tabla 2.1 Herramientas de desarrollo**

## Capítulo 4: Resultados

## 4. Resultados

### 4.1. Análisis de resultados.

Se abordarán los resultados obtenidos hasta ahora del proyecto, el margen de error de las estimaciones, la arquitectura y el diseño planificado inicial, versus el obtenido. Los módulos realizados hasta la fecha actual y alguno resultados de los módulos.

Se decidió con el equipo de proyecto priorizar el funcionamiento de la aplicación y luego de desarrollar todos los módulos funcionales agregar una interfaz que pueda ser aceptada por los stakeholders.

Se inició la programación del código realizando el cálculo de decaimiento radiactivo para el radiofármaco Flúor-18 FDG, logrando el funcionamiento del código con parámetros ingresados de manera manual.

```
import math

Act_ini = float(raw_input("Ingrese Actividad Inicial: "))
tiempo = float(raw_input("Ingrese tiempo: "))
des_fluor = float(109.771)
Act_fin = Act_ini * (math.exp(-(0.693 * float(tiempo) / des_fluor)))

print "El decaimiento es", Act_fin
```

Figura 14 - Código fórmula decaimiento radiactivo.

```
===== RESTART: /Users/Sebas/Desktop/Codigo PET/Decaimiento.py =====
Ingrese Actividad Inicial: 120
Ingrese tiempo: 30
El decaimiento es 99.29522252044228
>>> |
```

Figura 15 - resultado formula decaimiento radiactivo.



El código fue actualizado para poder ser implementado en conjunto con otros módulos, quedando de la siguiente manera:

- Módulo **cal\_decay**:

```
def cal_decay(dose_act,minut_decay):  
    dose_act=float(dose_act)  
    des_fluor =float(109.771)  
    minutos=float(minut_decay)  
    act_fin=dose_act*(math.exp(-(0.693*minutos/des_fluor)))  
    return act_fin
```

Figura 16 - Módulo actualizado fórmula decaimiento radiactivo.

Éste Módulo retorna el valor de actividad restante luego de ingresar la actividad inicial y el tiempo transcurrido en formato de minutos, valor que es ocupado en conjunto con otros módulos según el requerimiento solicitado.

Para la realización de las funcionalidades propuestas se decidió implementar dos nuevos módulos de cálculos, estos son :

- Módulo **dif\_min**: que calcula la diferencia de tiempo entre una hora determinada por el usuario y la hora actual del sistema, dando un resultado en minutos.

```
def dif_min(time_x):  
    now= time.strftime('%H:%M')  
    start_dt = dt.datetime.strptime(now, '%H:%M')  
    end_dt = dt.datetime.strptime(time_x, '%H:%M')  
    diff = (start_dt - end_dt)  
    minutos= int(diff.seconds/60)  
    return minutos
```

Figura 17 - Módulo dif\_min .

- Módulo **dif\_min\_proy**: que calcula la diferencia de tiempo entre dos horas determinadas por el usuario, dando el resultado en minutos. En otras palabras es la cantidad de minutos que han transcurrido entre una hora y otra.

```
def dif_min_proy(time_1,time_2):
    start_dt = dt.datetime.strptime(time_2, '%H:%M')
    end_dt = dt.datetime.strptime(time_1, '%H:%M')
    diff = (start_dt - end_dt)
    minutos= int(diff.seconds/60)
    print (minutos)
    return minutos
```

Figura 18 - Módulo dif\_min\_proy .

Ambos Módulos se conectan con otros según los requerimientos solicitados.

Para la implementación del Módulo Chequeo de datos (Check\_output) se analizó crear restricciones individuales para cada tipo de dato que debe ingresar el usuario, con el fin de disminuir la probabilidad de fallo de la aplicación.

Los módulos implementados se presentan a continuación:

- **Módulo check\_output\_dose** : Encargado de controlar el ingreso adecuado de datos de tipo dosis, limitando los datos a tipo numérico con dos decimales. Si los valores son ingresado de manera incorrecta, se retornara una opción de ingreso nuevamente de datos con la restricción solicitada.

```
def check_input_dose():
    while True:
        dosis_u = input("Ingrese Dosis con punto y solo 2 decimales")
        try:
            dosis_u =float(dosis_u)
            dosis_u="%.2f" % (2, dosis_u)
            print (dosis_u)
            return dosis_u
        except ValueError:
            print ("Debe ingresar valores separador por PUNTO y solo dos decimales")
```

Figura 19 - Módulo Check\_input\_dose.

- **Módulo check\_input\_ml:** Encargado de controlar el ingreso adecuado de datos de tipo ml, limitando los datos a tipo numérico con dos decimales. Si los valores son ingresado de manera incorrecta, se retornara una opción de ingreso nuevamente de datos con la restricción solicitada.

```
def check_input_ml():
    while True:
        ml_u = input("Ingrese cantidad de ml con punto y solo 2 decimales")
        try:
            ml_u = float(ml_u)
            ml_u = "%.2f" % (2, ml_u)
            print (ml_u)
            return ml_u

        except ValueError:
            print ("Debe ingresar valores separador por PUNTO y solo dos decimales si tiene.")
```

Figura 20 - Módulo Check\_input\_ml.

- **Modulo check\_input\_hora:** Encargado de controlar el ingreso adecuado de datos de tipo Hora, limitando los datos a tipo numérico con formato HH:MM (Hora:minuto). Si los valores son ingresado de manera incorrecta, se retornara una opción de ingreso nuevamente de datos con la restricción solicitada.

```
def check_input_hora():
    while True:
        data = input("Ingrese hora en formato HH:MM")
        try:
            time.strptime(data, "%H:%M")
            hour=time.strptime(data, "%H:%M")
            hour_min=time.strptime('%H:%M', hour)
            print (hour_min)
            return hour_min
        except ValueError:
            print ("Debe ingresar formato valido HH:MM ")
```

Figura 21 - Módulo Check\_input\_hora.

Para los Módulos de Controladores (Controllers) se han desarrollado a la fecha:

- Módulo **dose\_inicio** : Encargado de solicitar los datos iniciales de Dosis, tiempo y cantidad de ml.

```
def dose_inicio():
    dose=check_input_dose()
    hour=check_input_hora()
    ml=check_input_ml()
    tupla=(dose,hour,ml)
    lista.append(tupla)
    print ("Datos ingresados con exito")
```

Figura 22 - Módulo dose\_inicio.

- Modulo **dose\_now** : Encargado de entregar la dosis en tiempo Real.

```
def dose_now(lista):
    act_ini=dose_last(lista)
    time=time_last(lista)
    minutos=dif_min(time)
    dose_now=cal_decay(act_ini,minutos)
    dose_now=round(dose_now,3)
    return dose_now
```

Figura 23 - Módulo dose\_now.

- Modulo **dose\_proy**: Encargado de entregar la dosis en un tiempo proyectado.

```
def dose_proy(lista):
    hour_x=check_input_hora()
    act_ini=dose_last(lista)
    time=time_last(lista)
    minutos=dif_min_proy(time,hour_x)
    dose=cal_decay(act_ini,minutos)
    dose=round(dose,3)
    print ("la dosis proyectada para las",hour_x," es", dose, "mCi")
    return dose
```

Figura 24 - Módulo dose\_proy.

Por Último, se implementó provisoriamente un menú y un selector de opciones para la realización de pruebas del código expuesto anteriormente, con el fin de comprobar que los módulos se han conectado de manera correcta entre sí.

- Módulo **menu** y **opciones** : Permiten la navegabilidad dentro del código y actúa como herramienta provisoria antes de la interfaz gráfica a realizar.

```
def menu():
    #MENU DE OPCIONES
    print("** :::::::::::::::::::::::::::::: **")
    print(":: Seleccione una Opcion ::")
    print("** :::::::::::::::::::::::::::::: **")
    print("-----")
    print("| Ingresar Dosis Inicial   :-> 1   |")
    print("| Calcular Dosis Proyectada:-> 2   |")
    print("| Calcular Dosis Real-Time: -> 3   |")
    print("| Calcular mL para X dosis: -> 4   |")
    print("| Ingreso dato actualizado: -> 5   |")
    print("| Generar Reporte :         -> 6   |")
    print("| Salir:                    -> s   |")
    print("-----")
    print("** :::::::::::::::::::::::::::::: **")
    print("\n")
    #MENU DE OPCIONES
#OPCIÓN DEL MENU

def opciones(opc=0):
    opcion =input("Seleccione una Opción... ")
    return opcion

opc=0
while opc != "s":
    menuPrincipal = menu()
    opc = opciones()
    #OPCIÓN DEL MENU
```

Figura 25 - Módulo menu y opciones..

Los resultados generados en la actualidad permiten:

- Menu y selector de opciones ingresadas.

```
** :::::::::::::::::::::::::: **
:: Seleccione una Opcion ::
** :::::::::::::::::::::::::: **

-----
| Ingresar Dosis Inicial   :-> 1   |
| Calcular Dosis Proyectada:-> 2   |
| Calcular Dosis Real-Time:-> 3   |
| Calcular mL para X dosis:-> 4   |
| Ingreso dato actualizado:-> 5   |
| Generar Reporte :       -> 6   |
| Salir:                  -> s   |
-----

** :::::::::::::::::::::::::: **

Seleccione una Opción... |
```

Figura 26 - Ejecución del menu y selector de opciones.

- Ingreso de Dosis inicial en conjunto con chequeo de inputs.

```
Seleccione una Opción... 1
Ingrese Dosis con punto y solo 2 decimalesa
Debe ingresar valores separador por PUNTO y solo dos decimales
Ingrese Dosis con punto y solo 2 decimales120.00
120.00
Ingrese hora en formato HH:MMds
Debe ingresar formato valido HH:MM
Ingrese hora en formato HH:MM10:20
10:20
Ingrese cantidad de ml con punto y solo 2 decimalesdf
Debe ingresar valores separador por PUNTO y solo dos decimales si tiene.
Ingrese cantidad de ml con punto y solo 2 decimales20.1
20.10
Datos ingresados con exito
Regresando al Menu
```

Figura 27 - Ejecución de cheque de input e ingreso de dosis inicial.



- Conocer la dosis en tiempo real.

```

Seleccione una Opción... 3
La Dosis para este momento es; 15.227 mCi
Regresando al Menu

```

Figura 28 - Ejecución de Dosis en tiempo real.

- Conocer una dosis estimada en el tiempo.

```

Seleccione una Opción... 2
Ingrese hora en formato HH:MM18:00
18:00
460
La dosis proyectada para las 18:00 es 6.576 mCi
Regresando al Menu

```

Figura 29 - Ejecución de Dosis proyectada en un tiempo.

Dentro de los principales problemas que se fueron dilucidando a la fecha están:

- falta de conocimientos para programar en lenguajes modernos y avanzados.
  - Se realizó curso en python 3 para proceder con la programación.
- El ingreso de datos pueden generar errores con ciertos caracteres, que no han sido controlables hasta la fecha.
  - Se realizaron los módulos de chequeo de dosis dedicados para cada tipo de dato ingresados.
- Tener los codigos en archivos separados generaba desorden y poca efectividad para las pruebas
  - Se creó un archivo con el código principal que generó relaciones entre los módulos.
- Las posibles alternativas de interfaz gráfica en bibliotecas de python no son dominadas por el grupo de desarrollo
  - Se están realizando cursos para dominar la implementación de interfaz gráfica.

Cabe destacar que los resultados han sido expuestos al cliente el día viernes 7 de diciembre 2018, en una reunión de avance que solicito de manera extraordinaria.

Los avances han producido un impacto positivo, por lo cual tenemos la aprobación de los módulos implementados por parte del cliente en fase teórica hasta la actualidad.

## 4.2. Diseño detallado.

### Diagrama Módulo ingreso de dosis inicial.

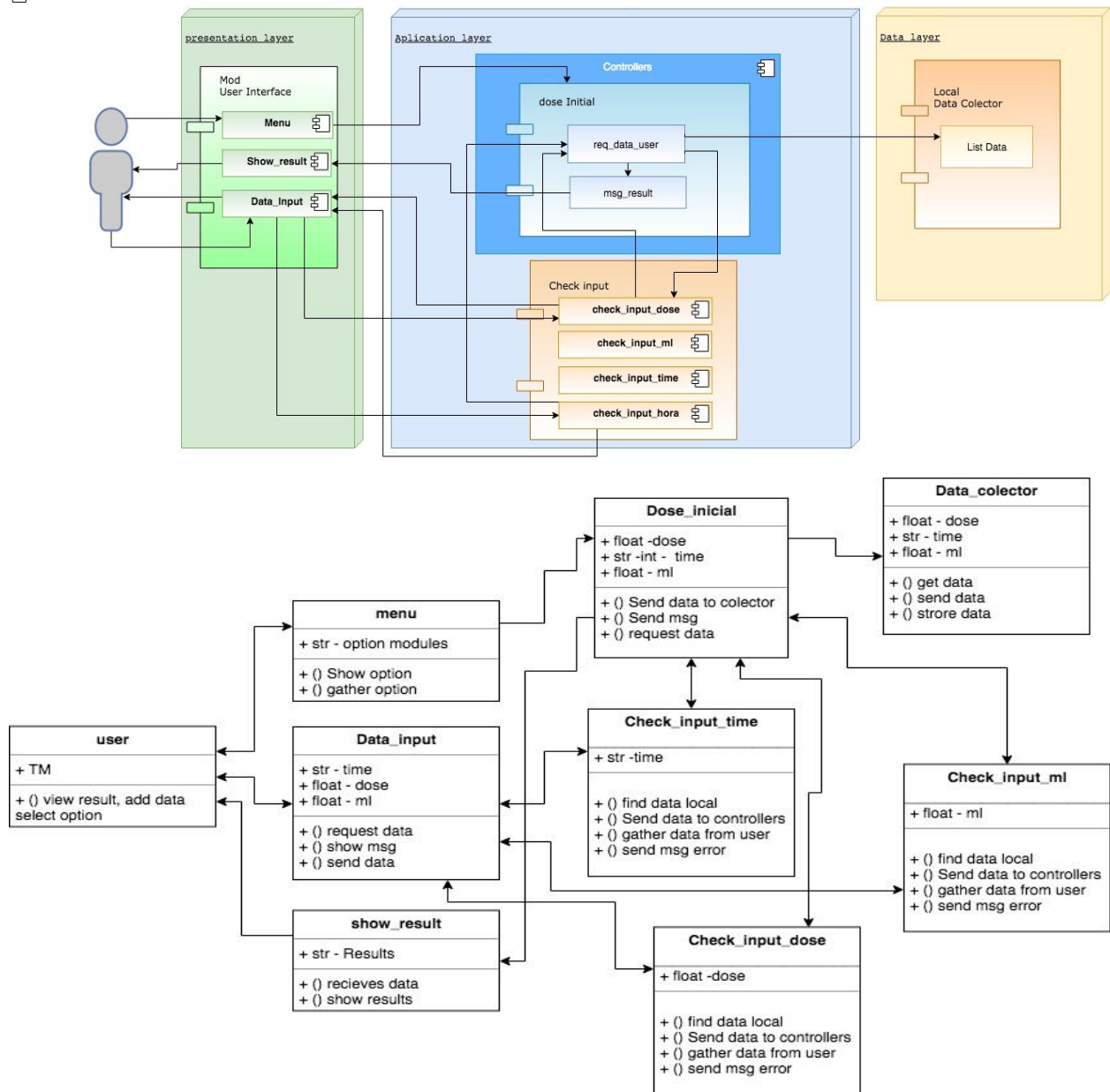


Figura 30. Diseño detallado para módulo ingreso de dosis Inicial.



## Diagrama Módulo Dosis en Tiempo Real.

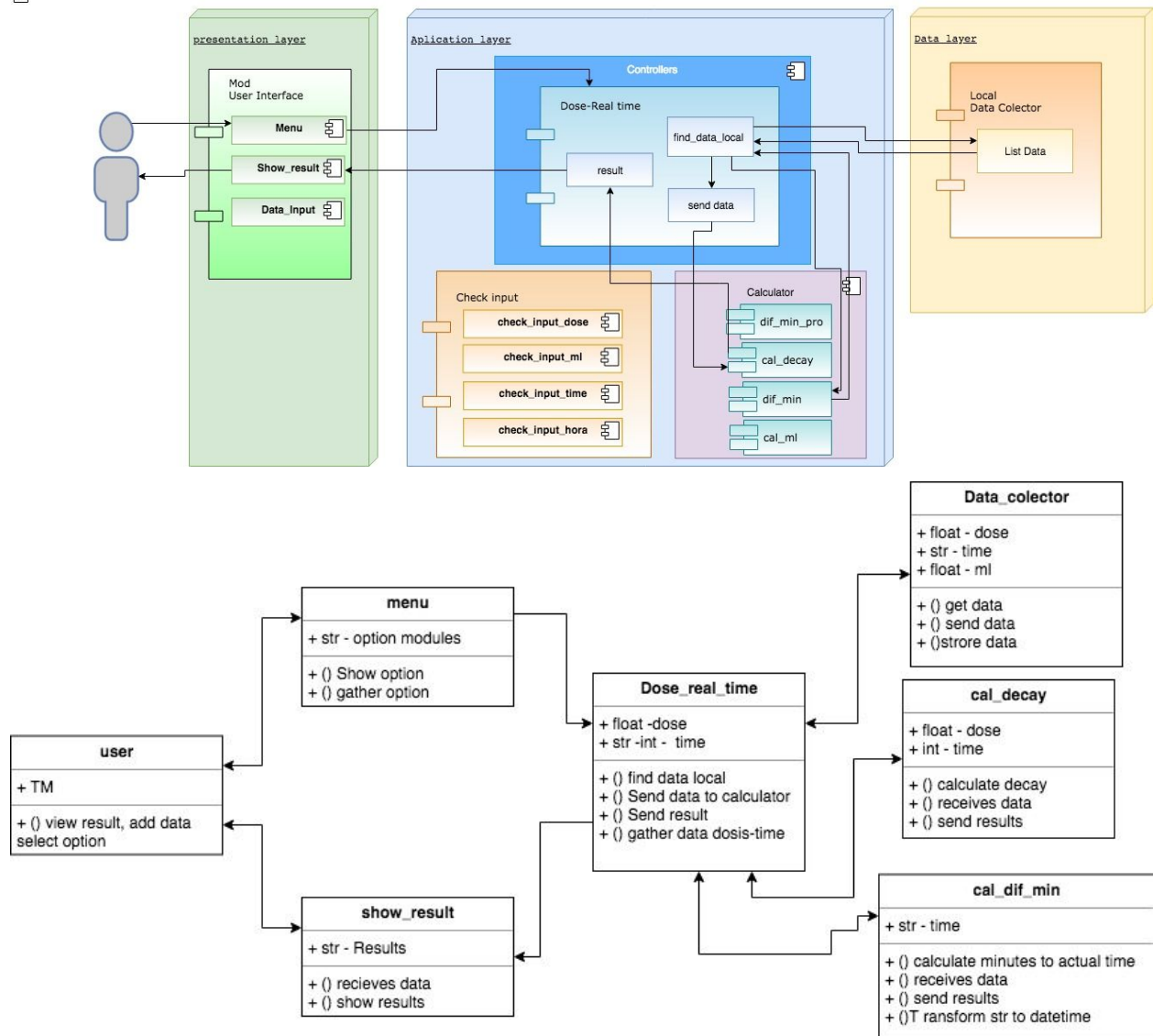


Figura 31. Diseño detallado para módulo Dosis en tiempo real.

## Diagrama Módulo Dosis Proyectada.

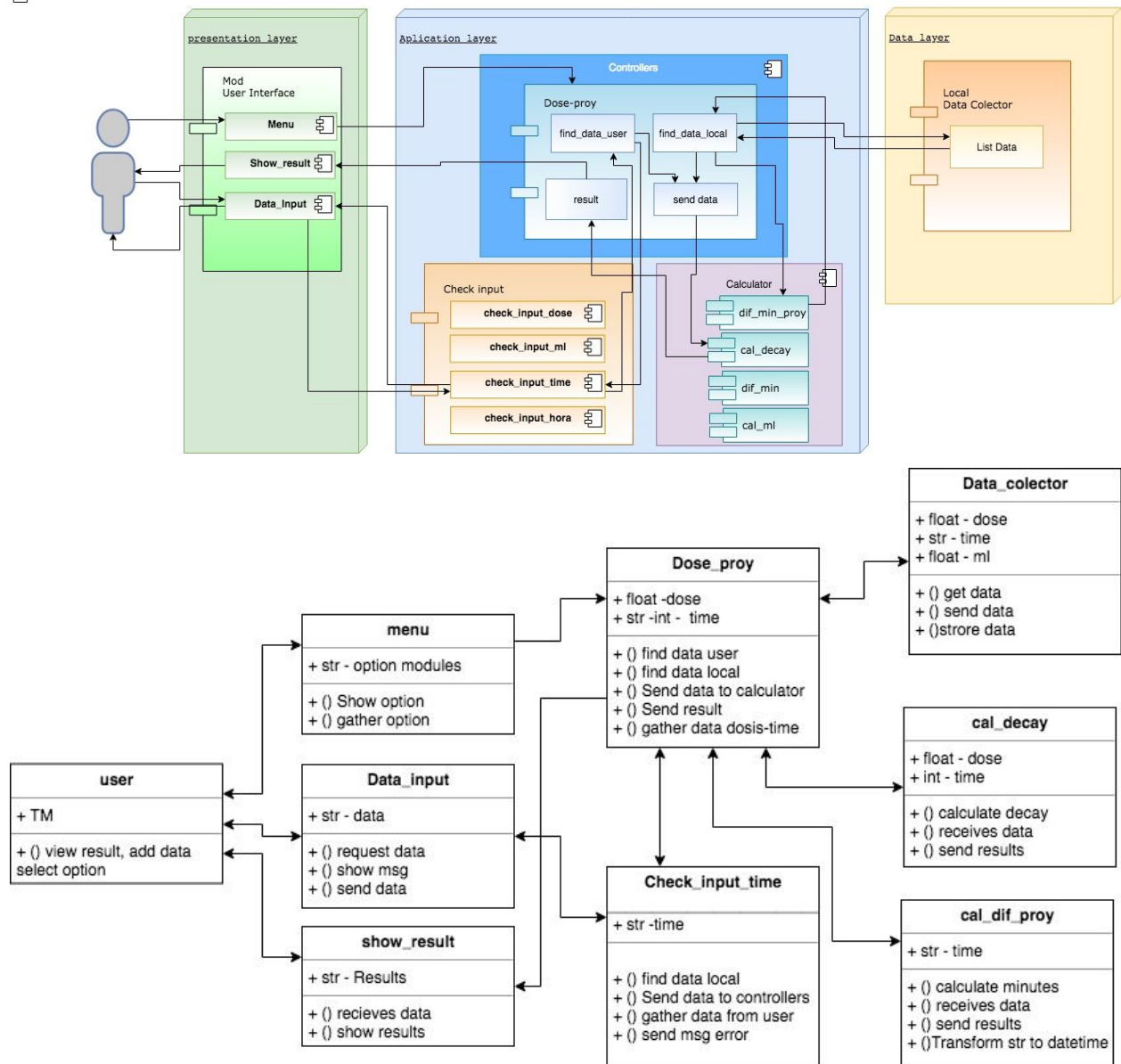


Figura 32. Diseño detallado para módulo Dosis proyectada en el tiempo.

### 4.3. Situación futura.

- El Desarrollo de la aplicación ha sido verificada por los stakeholders, en la última reunión (sprint 1) realizada en Clínica Reñaca, con un resultado positivo ante el prototipo funcional propuesto.
- El primer prototipo entregado a los stakeholders permite la realización del ingreso de datos iniciales, realizar calculo de dosis en tiempo real y cálculo de dosis proyectada. Avalado por la teoría de las fórmulas de decaimiento, ha tenido resultados positivos, los cuales han logrado mitigar en parte la probabilidad de no lograr el funcionamiento básico de la aplicación.
- Los pasos a seguir son:
  - Desarrollar los módulos restantes de la aplicación.
  - Realizar las pruebas de cumplimiento de requerimientos específicos N° 1 , 4, 5 y 6
  - Lograr la funcionalidad completa de la aplicación.
  - Definir las herramientas para la interfaz acorde a los mockup propuestos.
  - Comenzar el trabajo de implementación de interfaz a la aplicación.
  - Realizar las pruebas en conjunto con stakeholders.
  - Entrega del producto a cliente.

### 4.3.1. Resultados esperados.

**Clínica Refiaca**

**PET-Manager V1.0**

**Ingreso Paciente**

Actividad  mCi

Hora  HH:mm Am

Cantidad  mL

Ok

**Opciones**

Ingreso Paciente

Actividad Tiempo Real

Actividad Proyectada

Dosificación

Generar Reporte

**Ingreso Datos Iniciales**

Actividad  mCi

Hora  HH:mm Am

Cantidad  mL

Ok

**Actividad Proyectada**

83.3 mCi

Hora Actual : 12:10 Pm

Hora Futura:  14:30 Pm HH:mm am/pm

**Actividad Actual**

120.2 mCi

Hora actual : 12:10 PM

Cantidad Actual : 5 mL

**Dosificación**

2,5 mL

Hora Actual : 12:10 Pm

Actividad necesaria  8.00 mCi

**Reporte Jornada**

Paciente 1: Actividad: XX.X mCi  
Hora: HH:mm Am/Pm  
Cantidad: XX.X mL

Paciente 2: Actividad: XX.X mCi  
Hora: HH:mm Am/Pm  
Cantidad: XX.X mL

Paciente 3: Actividad: XX.X mCi  
Hora: HH:mm Am/Pm  
Cantidad: XX.X mL

Figura 33 - Cuadro Mockups esperados

## Capítulo 5: Conclusiones

## **5. Conclusiones.**

### **5.1. Conclusión.**

Por ahora solo podemos concluir en que el avance de proyecto está bajo las fechas establecidas, los primeros desarrollos han sido aceptados por el cliente y el proyecto sigue su ciclo de vida.

### **5.2. Problemas abiertos.**

Uno de los principales problemas actuales es el tiempo disponible para la entrega del mockup básico para la finalización del sprint número 3.

Otro es la falta de conocimientos sobre interfaz gráfica acorde al proyecto que posee el team.

Ante esto se están ocupando las técnicas de mitigación propuestas para ambos riesgos que pueden afectar el proyecto.

### **5.3. Trabajo futuro.**

Finalizar interfaz Gráfica y realizar entrega del producto.