

Diseño y Programación Orientada a Objetos – Sección 6

Jerónimo Vargas Rendon - 202113305

Nicolas Camargo Prieto – 202020782

Documento de Diseño

Análisis

→ **Iteración Antes de los Nuevos Requerimientos.**

Descripción de Requerimientos Funcionales

HISTORIAS DE USUARIO

1. Como participante de un proyecto, quiero poder tener un sistema que registre todo el proceso de desarrollo de dicho proyecto, con el fin de conocer los miembros que participaron e información de las diferentes tareas que se efectuaron los miembros del grupo.

- Registrar en un sistema información del proyecto.
- Ver los miembros del proyecto.
- Ver registro de actividades.
- Ver información calculada de los datos.

2. Como dueño del proyecto creado, quiero poder registrar nuevos participantes ya que inicialmente soy el único, mientras que es muy probable que un proyecto se conforme de varios participantes que aportan al desarrollo de este mismo.

- Registrar los miembros del proyecto (Crear un participante).
- Registrar nombre y correo del participante.

3. Como participante del proyecto, deseo poder registrar en el sistema las actividades que voy realizando en el proyecto, para dejar soporte de la tarea que realicé, con la fecha en la que se realizó, información que la caracteriza y tiempos en el cual se desarrolló.

4. Como participante del proyecto, quiero poder hacer varios registros o commits en una misma tarea, debido a que la actividad a realizar puede que tarde días en realizarse.

5. Como participante del proyecto, quiero poder cronometrar el tiempo que tardo en realizar una actividad del proyecto, además deseo poder detener el cronómetro en caso de que deba interrumpir el desarrollo de la actividad.

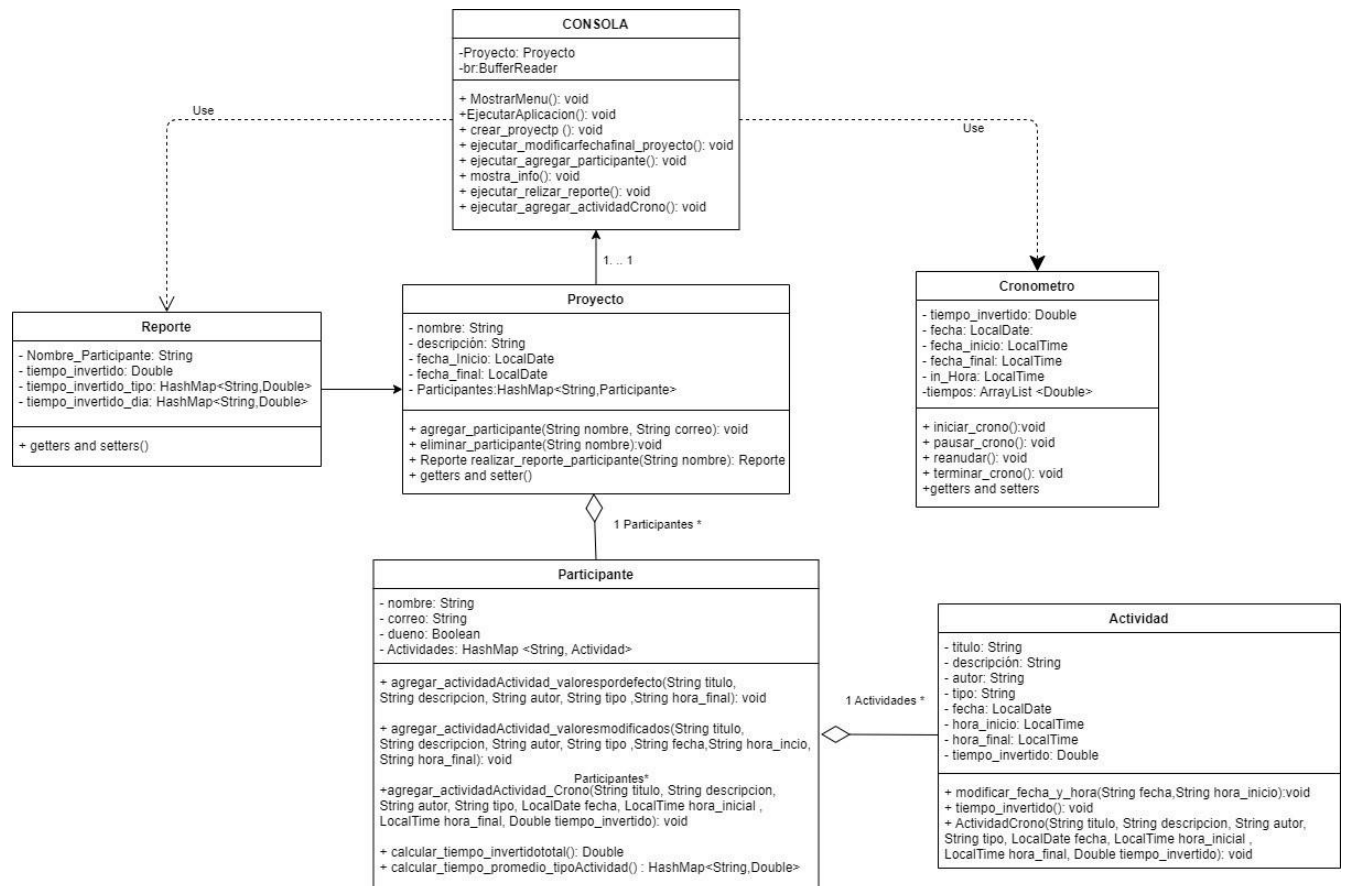
6. Como dueño del proyecto, quiero poder recibir un reporte que informa de las actividades realizadas por cada miembro y diferentes características calculadas.

- Ver total actividades.
- Ver Tiempo total invertido.
- Ver Tiempo promedio por actividad.

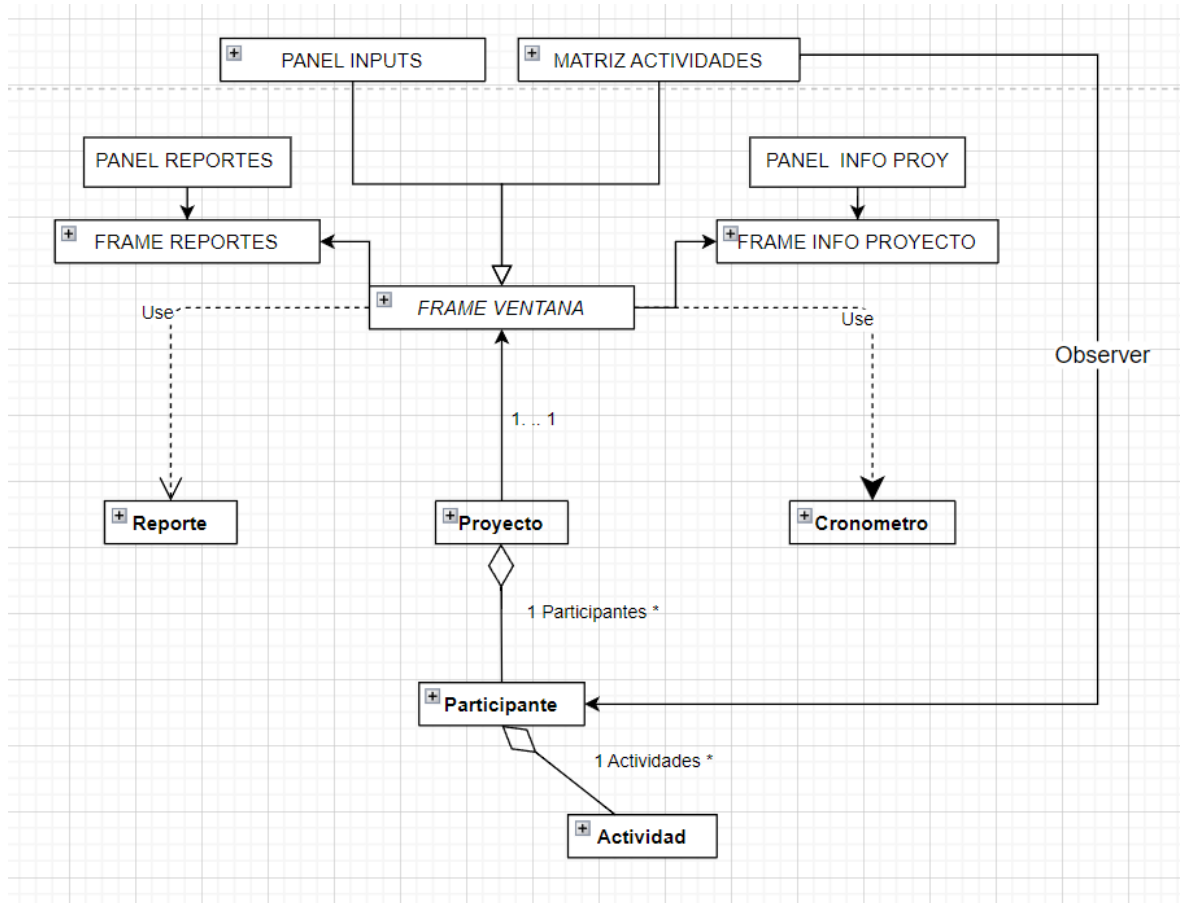
Modelos de dominio (UML) Actuales.

A continuación, se presentará el modelo (UML) del producto que se lleva hasta el momento.

- UML Lógica de Negocio



- UML con la integración de la interfaz gráfica.



Los diagramas anteriores muestran las relaciones que podemos encontrar en el proyecto dentro de la lógica y la interfaz, donde se decidió construir con esta estructura ya que de este modo la complejidad era menor y se reducía el acoplamiento entre componentes. Además, se implementó un patrón de observador para poder hacer dinámica la interfaz en el momento de operar la matriz de actividades.

→ **Iteración con los Nuevos Requerimientos.**

Proceso

Iteración

Descripción de requerimientos funcionales

Historias de Usuario:

1. Como participante de un proyecto, quiero poder tener un sistema que registre todo el proceso de desarrollo de dicho proyecto, con el fin de conocer los miembros que participaron e información de las diferentes tareas que se efectuaron los miembros del grupo.

- Registrar en un sistema información del proyecto.
- Ver los miembros del proyecto.
- Ver registro de actividades.
- Ver información calculada de los datos.

2. Como dueño del proyecto creado, quiero poder registrar nuevos participantes ya que inicialmente soy el único, mientras que es muy probable que un proyecto se conforme de varios participantes que aportan al desarrollo de este mismo.

- Registrar los miembros del proyecto (Crear un participante).
- Registrar nombre y correo del participante.

3. Como dueño del proyecto, quiero poder establecer una WBS compuesta por sus paquetes de trabajo y tareas organizadas de forma jerárquica.

4. Como participante del proyecto, deseo poder registrar en el sistema las actividades que voy realizando en el proyecto, para dejar soporte de la tarea que realicé, con la fecha en la que se realizó, información que la caracteriza y tiempos en el cual se desarrolló.

5. Como participante del proyecto, quiero poder hacer varios registros o commits en una misma tarea, debido a que la actividad a realizar puede que tarde días en realizarse.

6. Como participante del proyecto, quiero poder cronometrar el tiempo que tardo en realizar una actividad del proyecto, además deseo poder detener el cronómetro en caso de que deba interrumpir el desarrollo de la actividad.

7. Como participante del proyecto, quiero poder definir la o las tareas dentro de la WBS a la que una actividad quedara asociada y establecer si con esa actividad se completa la tarea.

8. Como dueño del proyecto, quiero poder recibir un reporte que informa de las actividades realizadas por cada miembro y diferentes características calculadas.

- Ver total actividades.
- Ver el total de paquetes de trabajo y actividades
- Ver Tiempo total invertido.
- Ver Tiempo promedio por actividad.
- Ver Tiempo total de una tarea

9. Como dueño del proyecto, quiero poder generar y ver un reporte del avance del proyecto y del equipo de trabajo, que indique los siguientes aspectos:

- Avance del proyecto: sobre los tiempos de entrega y realización del proyecto, cantidad de tareas completadas, tiempo en las tareas, y un avance sobre alguna tarea.
- Calidad de planeación: conocer el tiempo total invertido sobre el tiempo que se estimó y encontrar la tasa de error que se encontró.
- Equipo: conocer el tiempo que cada miembro ha invertido en una tarea, las tareas pendientes y el tiempo planeado para esas tareas.
- Resumen del proyecto: resumen general que incluye tiempos totales, tiempos estimados, avances, etc.

Restricciones:

Aunque el diseño del proyecto incluye la mayoría de los aspectos esenciales y requerimientos funcionales, también este cuenta con unas cuentas restricciones. Por un lado, el proyecto solo

puede ser realizado por dos personas, quienes deben tener en cuenta los cambios realizados por el otro. Se debe usar eclipse (Java) para su realización, y de la misma forma, todos los cambios deben ser subidos a GitHub. Por otro lado, la modificación de datos en el registro puede generar conflictos con la fecha y hora en la que se realizaron las actividades. A sí mismo, establecer en el diseño varias tareas con el mismo nombre, puede suponer conflictos a la hora de guardar la información. De la misma forma, el tipo de actividad que debe ser definida (Documentación, Implementación, Pruebas, etc.) deben ser seleccionadas previamente, pues hacen parte de un arraylist dentro de la clase proyecto, que es importante para interactuar con el resto de las clases. Por otro lado, cuando se registre una actividad se debe indicar el tipo de tarea de WBS a la que la actividad quedara asociada. Por último, no pueden eliminarse tareas WBS que ya tengan actividades asociadas.

Diseño Nuevos Requerimientos.

La implementación que se va a realizar para poder cumplir con todos requerimientos funcionalidades propuestas se va a dividir de la siguiente manera.

- Requerimiento de Lógica

Dentro de la lógica implementar algunos métodos que puedan sacar cálculos de estadística con respecto a toda la información almacenada en el proyecto. Es por esto que dentro de la clase de proyecto van a ver varios métodos que retorne cifras esenciales para reportarle al usuario.

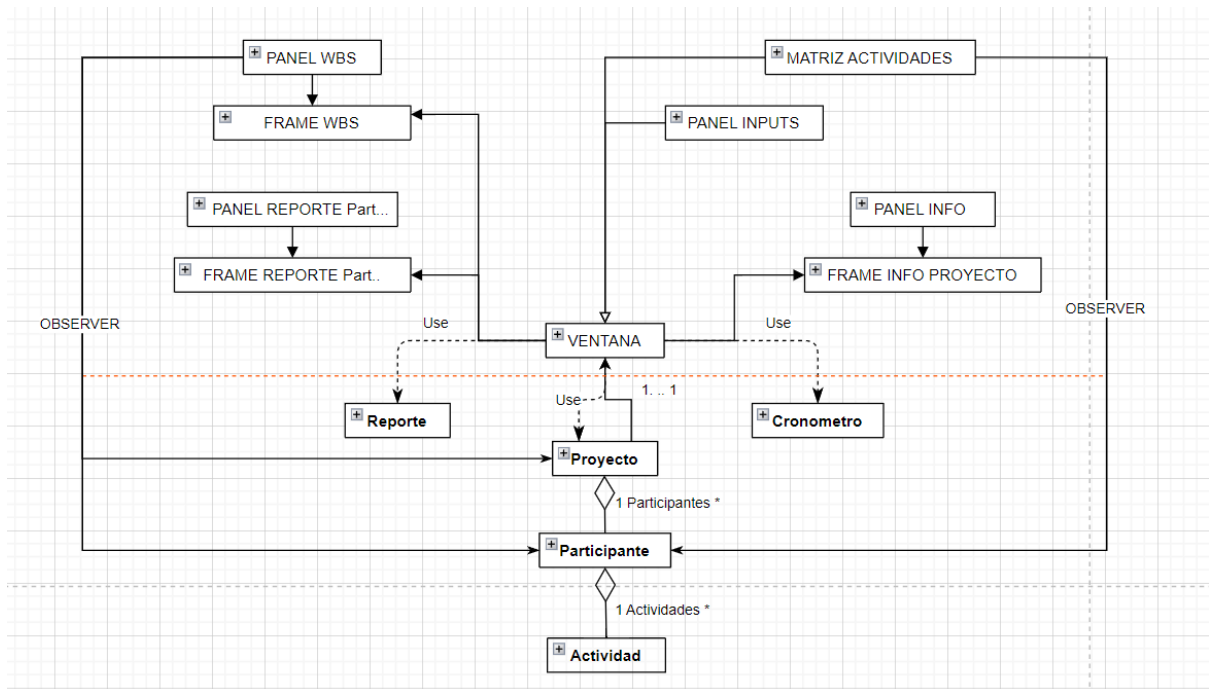
- Requerimientos de Interfaz grafica

Con los nuevos Datos generado por la lógica se tendrá que desarrollar nuevos mecanismos de interacción con el usuario para mostrarle la información calculada en estos nuevos requerimientos.

Además, En la interfaz gráfica se desarrollará otro frame que permita mostrar el comportamiento de la herramienta WBS

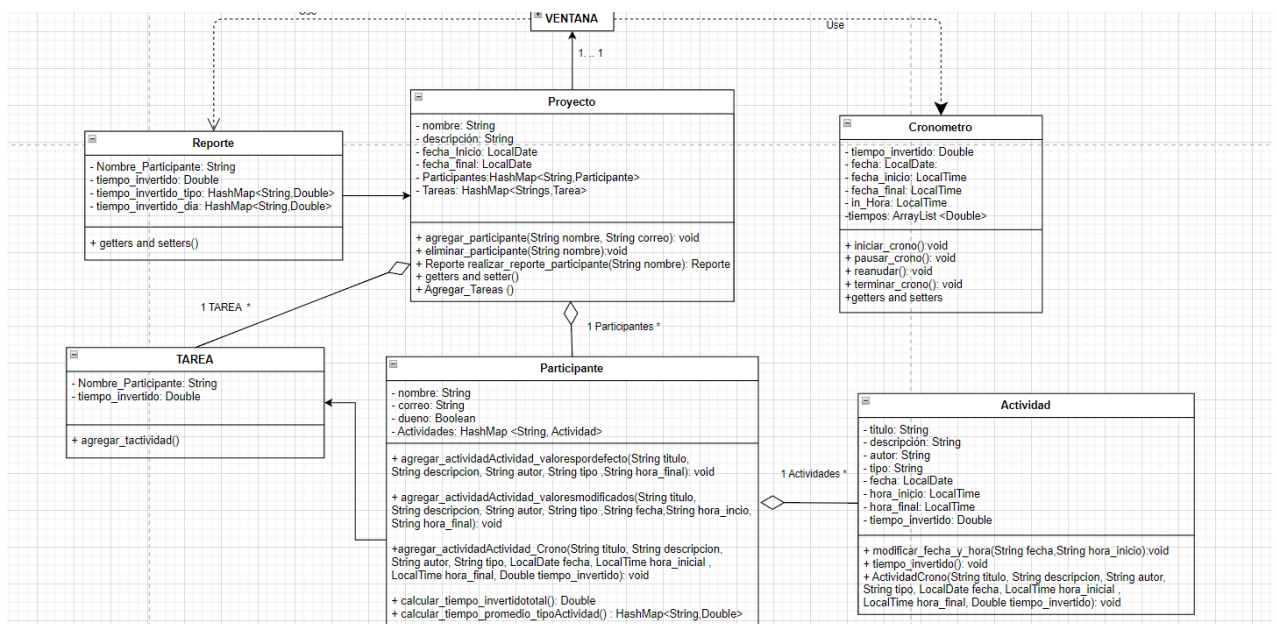
Diseño UML

INTERFAZ



En la interfaz vemos que se agregan dos componentes relacionadas la estructura WBS, que se encaran con toda la interacción grafica con el usuario y poderle suministrar el requerimiento de llevar un control del proyecto.

LOGICA



En el diseño anterior vemos que en la lógica se propone crear un objeto “TAREA” con el propósito de que este pueda almacenar información del proyecto con la estructura para poder suminístrale la información a mostrar al WBS.

Modelo WBS:



Estereotipos (roles):

- **Proyecto:** Structurer, pues es responsable de mantener la estructura, información y ciertos comandos que se deben hacer
- **Participante:** Controller, ya que es el principal responsable del proyecto, así como, el que se encarga de las tareas como hacer commits o generar un reporte en base a una actividad.
- **TAREA:** Information holder.
- **Reporte:** Information holder, tiene que almacenar los reportes.
- **Actividad:** Structurer, mantiene información sobre las actividades, commits y genera los reportes.
- **Commits:** Service provider, permite al miembro o participante guardar información sobre una actividad.
- **Interfaz:** Interfacer, permite al usuario introducir información y recibir resultados visuales.
- **Cronometro:** Service provider, realiza el cálculo del tiempo de una actividad y de una tarea (dada por la suma del tiempo de sus actividades), almacena y despliega esta información.
- **Ventana:** Service provider, ya que es el Front End del sistema, es decir el componente que se encarga de toda la interacción con el usuario con respecto en temas de pedirle datos y presentarle información.

Reflexión

Finalmente, se evidencia que el proceso de diseño que se llevó a cabo abarca de manera eficiente lo que se desarrolla en el programa, pues de manera significativa se denota la separación entre la interfaz visual y la lógica del programa, la cual ahora abarca la implementación de WBS y el reporte de tareas y graficas. De tal manera, el modelo planteado ayuda en el encapsulamiento de las actividades y la lógica de WBS implementada, así como se complementa con lo que debe mostrar la interfaz.

Sin embargo, la principal desventaja que se identifica dadas las nuevas implementaciones son sobre la lógica del programa, ya que la organización jerárquica de WBS cambia parcialmente el modelo usado anteriormente, generando posiblemente una complejidad más alta.

Por otro lado, en cuanto a los trade-offs, la nueva adición del sistema WBS, así como los reportes con gráficas, pueden resultar sumamente beneficiosos para la manipulación y el entendimiento de la información del proyecto, sus actividades y tareas. También, se puede

evidenciar que, pese a que el modelo anterior era más sencillo de implementar, este nuevo modelo jerarquizado (WBS) resulta, aunque más complejo, mejor organizado en cuanto a la categorización de las tareas y actividades, que generan un programa más ordenado y limpio.

Posibles Errores y pruebas error

Ahora bien, una parte importante a tratar de este nuevo diseño es la contemplación de posibles errores que se presenten una vez implementado y la forma en la que se deberían tratar de solucionar.

Este manejo de posibles errores se tratará con el servicio que ofrece JUnit, una herramienta que permite construir un conjunto de pruebas unitarias en este caso para poder identificar que las aserciones de los métodos implementados en la lógica sean los esperados. A la hora de evaluar los métodos, claramente se tendrá en cuenta la cobertura acertada ya que del 100% es innecesario, por lo tanto, la cobertura solo será mediante pruebas unitarias que evaluarán los métodos enfocados en cálculos de información a reportar al usuario. En esta altura del proyecto se puede evidenciar un buen funcionamiento de integración y creación de estructuras de datos dentro del sistema.